Digi IX15

Gateway

User Guide

Firmware version 22.2

Revision history—90002400

Revision	Date	Description
	September 2021	Release of Digi IX15 firmware version 21.8: ■ Added LXC container support for running localized containers on the device. ■ Added support for maintenance windows triggers to control when a device is available for Digi Remote Manager maintenance activity. ■ VPN enhancements: • Added support for L2TPv3 tunneling. • New option to enable, disable, or force IPsec IKE fragmentation. ■ Improved options for creating a custom default configuration: • **system backup** CLI commands for generating a custom default config file based on the active config settings on the device. • New section on the **File System** page of the Web UI for loading a configuration backup file as the custom default config • New persistent files folder accessible through Digi Remote Manager where users can upload a configuration backup. • Added ability to clear a custom default configuration by performing by pressing the ERASE button, waiting for the device to reboot, then pressing the ERASE again. ■ Added ability to override or edit SSH server options. ■ Added options for filtering Wi-Fi scanner results based off of MAC addresses or RSSI signal strength. ■ Added options for sending local device event logs to Digi Remote Manager. ■ New **system time** CLI command for manually setting the local date and time. ■ New firewall packet filter options to allow/deny packets based off of a range of source or destination MAC addresses. ■ New **monitoring metrics upload** CLI command to send on-demand health metrics to Digi Remote Manager. ■ Added support for the configuration of custom scripts that will be run manually, and a new **system script start** CLI command to run manual scripts. ■ New "Find me" feature that flashes cellular-related LEDs to help locate the device onsite and a new **system find-me** command.

Revision	Date	Description
		■ Added datapoint.upload_multiple function to digidevice python module for uploading multiple datapoints to DigiRM at once. ■ Added **clear dhcp-lease** command to remove all dynamic DHCP leases or certain DHCP leases based on MAC address or IP address. ■ Added **speedtest** command for performing on-demand iPerf or nuttcp speedtests. ■ Local users are now required to be assigned to an authentication group. ■ New Network → Advanced → Sequential DHCP address allocation configuration setting for controlling if DHCP addresses are assigned sequentially or randomly (disabled by default). ■ Added ability to control if DHCP addresses are assigned sequentially or randomly (disabled by default). ■ Added 802.1x port-based network access control, configurable per network interface.
	November 2021	Added the **pyinstall** command. Added **Install Python packages** and **IX15 BLE configurator** sections. Added the IX15 BLE Configuration demo.
	December 2021	Release of Digi IX15 firmware version 21.11: ■ Configuration option to allow for automatic update of new firmware (disabled by default). ■ User authentication enhancements: • Added TACACS+ command authorization and accounting for Admin CLI commands. • Added a new username alias to provide an alternate username that allows for special characters in the username. ■ VPN enhancements: • Added support for PPP-over-L2TP tunnels. • Changed default IPsec IKE DH group to 14 for enhanced compatibility with industry standard settings. ■ Location services enhancements: • New option for Location service to specify the talker ID used in NMEA mesage strings. • Include the mode indicator field in NMEA messages constructed when a GPS fix isn't obtained. ■ SureLink enhancments: • Added new settings under cellular Surelink options to have the device reset the cellular modem if a specified number of Surelink tests fail • Added **show surelink** Admin CLI command.

Revision	Date	Description
		■ Serial port enhancements: • New option to add and configure an external USB-to-serial adapter. • Disable serial history in remote access mode by default. ■ Support for sending analog and digial I/O health metrics to Digi Remote Manager. ■ Added **show containers** Admin CLI command.
	March 2022	Release of Digi IX15 firmware version 22.2: ■ VPN enhancements: • Renamed **VPN** > **IPsec** > **Tunnels** > **Policies** > **Local network setting** to **Local traffic selector** and added **Remote traffic selector**. • Added a **Dynamic** option to the **Local traffic selector** to configuration of a local network by protocol and/or port instead of a network address range. • Added Protocol and Port match criteria for **Local traffic selector** and **Remote traffic selector**. • Added **VPN** > **IPsec** > **Advanced** > **Debug level** to specify the logging verbosity of IPsec messages in the device system logs. ■ Enhancements to communications with Digi Remote Manager: • Enhanced security for communications with Digi Remote Manager by using client-side certificates. • The default URL for the device's Remote Manager connection is now edp12.devicecloud.com. This URL is required to utilize the client-side certificate support. ■ New **Switch SIM** SureLink action for WWAN interfaces, which allows SureLink to be configured to switch to the alternate SIM if the modem is connected but SureLink test are failing. • New **Switch SIM fail count** option to determine how many SureLink failures are required prior to switching to the alternate SIM. ■ New **Socket ID string** option to send the configured text to the remote server(s) when a TCP socket connection is opened to the serial port. ■ New **cat** Admin CLI command for displaying file contents.
C	March 2022	Added translated safety instructions.

Trademarks and copyright

Digi, Digi International, and the Digi logo are trademarks or registered trademarks in the United States and other countries worldwide. All other trademarks mentioned in this document are the property of their respective owners.

© 2022 Digi International Inc. All rights reserved.

Disclaimers

Information in this document is subject to change without notice and does not represent a commitment on the part of Digi International. Digi provides this document “as is,” without warranty of any kind, expressed or implied, including, but not limited to, the implied warranties of fitness or merchantability for a particular purpose. Digi may make improvements and/or changes in this manual or in the product(s) and/or the program(s) described in this manual at any time.

Warranty

To view product warranty information, go to the following website:

www.digi.com/howtobuy/terms

Customer support

Gather support information: Before contacting Digi technical support for help, gather the following information:

- Product name and model
- Product serial number (s)
- Firmware version
- Operating system/browser (if applicable)
- Logs (from time of reported issue)
- Trace (if possible)
- Description of issue
- Steps to reproduce

Contact Digi technical support: Digi offers multiple technical support plans and service packages. Contact us at +1 952.912.3444 or visit us at www.digi.com/support.

Feedback

To provide feedback on this document, email your comments to

techcomm@digi.com

Include the document title and part number (Digi IX15 Gateway User Guide, 90002400 C) in the subject line of your email.

Contents

Revision history—90002400 2

Digi IX15 Gateway User Guide

Overview 20
What's new in Digi IX15 version 22.2 20
IX15 compatibility with S2C XBee devices 21
    Zigbee 21
    DigiMesh 23
    802.15.4 25
Related documents 26

Safety instructions

Safety instructions 28
    XBee adapter, gateways, and routers 28
Инструкции за безопасност 28
Sigurnosne upute 29
Bezpečnostní instrukce 29
Sikkerhedsinstruktioner 30
Veiligheidsinstructies 31
Ohutusjuhised 31
Turvallisuusohjeet 32
Consignes de sécurité 32
Sicherheitshinweise 33
Οδηγίες ασφαλείας 34
Biztonsági utasítások 34
Istruzioni di sicurezza 35
Drošības instrukcijas 36
Saugos instrukcijos 36
Sikkerhetsinstruksjoner 37
Instrukcje bezpieczeństwa 37
Instruções de segurança 38
Instructiuni de siguranta 39
Bezpečnostné inštrukcie 39
Varnostna navodila 40
Adaptador XBee, puertas de enlace y enrutadores 40
Säkerhets instruktioner 41

Digi IX15 Quick start

Step 1: What's in the box 43
Step 2: Gather accessories 43
Step 3: Connect 44
Step 4: Configure 49
Step 5: Next steps 49

Get started

Step 1: Requirements 51
Step 2: Setup the hardware 51
Step 3: Program an XBee profile 51
Step 4: Join nodes to the IX15 network 53
Step 5: Review your XBee network 54
Next steps 54

Digi IX15 hardware reference

Digi IX15 features and specifications 57
IX15 accessories 57
IX15 front and side views 57
IX15 LEDs 59
- Power (PWR) 59
- SIM 59
- LTE 59
- XBee 60
- Signal quality indicators 61
- Ethernet Link and Activity 61
- Signal quality bars explained 61

IX15 power supply requirements 62
Power consumption 63
- Power consumption use cases 63
- Results 64

Digi IX15 serial connector pinout 65
- 10-pin serial cabling options 65

Antenna specifications for the cellular modem 65
Antenna specifications for the XBee RF Module 66
QR code definition 66

Hardware setup

Install SIM cards 69
- Apply Dielectric Grease over SIM Contacts 69
- SIM removal 70
- Tips for improving cellular signal strength 70

Connect data cables 70
Mount the IX15 device 70
- Attach to a mounting surface by using the mounting tabs 71
- Attach to DIN rail with clip 71

Configuration and management

Review IX15 default settings 75
Local WebUI 75
Digi Remote Manager 75
Default interface configuration 75
Other default configuration settings 76
Change the default password for the admin user 76
Configuration methods 78
Using Digi Remote Manager 80
Access Digi Remote Manager 80
Using the web interface 80
Log out of the web interface 81
Use the local REST API to configure the IX15 device 81
Use the GET method to return device configuration information 81
Use the POST method to modify device configuration parameters and list arrays 83
Use the DELETE method to remove items from a list array 84
Using the command line 86
Access the command line interface 86
Log in to the command line interface 86
Exit the command line interface 87

Manage an XBee network

Review the current XBee network state 89
Discover the XBee network 89
Configure active discovery 89
One-shot discovery 93
Configure and update an XBee network 93
What is an XBee profile? 94
Manage XBee profiles 94
Upload the XBee profile 95
Apply XBee profiles 95
Configure a sleeping network to work with the IX15 96
Export your network 98

Bluetooth Low Energy

Configure Bluetooth Low Energy 99
IX15 BLE configurator 102
Verify BLE connectivity 102

Power management

Configure a power profile 105
Suspend mode 109
Configure wake up sources 109
Enter suspend mode 111
Disable interfaces on suspend 112

Interfaces

Wireless Wide Area Networks (WWANs) 116
- Configure SureLink active recovery to detect modem failures 116
- Configure the device to reboot when a failure is detected 126
- Disable SureLink 133
- Using cellular modems in a Wireless WAN (WWAN) 137
- Configure a Wireless Wide Area Network (WWAN) 156
- Show WWAN status and statistics 165
- Delete a WWAN. 167
- Configure a Wide Area Network (WAN) 168

Local Area Networks (LANs) 177
- About Local Area Networks (LANs) 178
- Configure a LAN 178
- Show LAN status and statistics 185
- Delete a LAN 187
- DHCP servers 189
- Create a Virtual LAN (VLAN) route 205

Show Surelink status and statistics 208
- Show Surelink status for all interfaces 208
- Show Surelink status for a specific interface 208
- Show Surelink status for all IPsec tunnels 209
- Show Surelink status for a specific IPsec tunnel 209
- Show Surelink status for all OpenVPN clients 210
- Show Surelink status for a specific OpenVPN client 210

Serial port

Configure Login mode 212
Configure Remote Access mode 215
Configure Application mode 219
Configure UDP serial mode 221
Configure Modbus mode 226
Show serial status and statistics 230
Log serial port messages 230

Routing

IP routing 233
- Configure a static route 234
- Delete a static route 237
- Policy-based routing 238
- Configure a routing policy 239
- Routing services 247
- Configure routing services 247

Show the routing table 250
Dynamic DNS 251
- Configure dynamic DNS 251

Virtual Router Redundancy Protocol (VRRP) 257
- VRRP+ 257
- Configure VRRP 257
- Configure VRRP+ 261
- Example: VRRP/VRRP+ configuration 268
- Configure device one (master device) 269

Configure device two (backup device) ... 273
Show VRRP status and statistics ... 279

Virtual Private Networks (VPN)

IPsec ... 283
IPsec data protection ... 283
IPsec mode ... 283
IPsec modes ... 283
Internet Key Exchange (IKE) settings ... 283
Authentication ... 284
Configure an IPsec tunnel ... 284
Configure IPsec failover ... 311
Configure SureLink active recovery for IPsec ... 314
Show IPsec status and statistics ... 321
Debug an IPsec configuration ... 322
Configure a Simple Certificate Enrollment Protocol client ... 324
Example: SCEP client configuration with Fortinet SCEP server ... 328
Disable hardware cryptographic acceleration ... 333
OpenVPN ... 336
Configure an OpenVPN server ... 337
Configure an OpenVPN Authentication Group and User ... 346
Configure an OpenVPN client by using an .ovpn file ... 350
Configure an OpenVPN client without using an .ovpn file ... 353
Configure SureLink active recovery for OpenVPN ... 357
Show OpenVPN server status and statistics ... 365
Show OpenVPN client status and statistics ... 366
Generic Routing Encapsulation (GRE) ... 368
Configuring a GRE tunnel ... 368
Show GRE tunnels ... 373
Example: GRE tunnel over an IPSec tunnel ... 374
L2TP ... 389
Configure a PPP-over-L2TP tunnel ... 389
Configure SureLink active recovery for PPP-over-L2TP ... 399
L2TP with IPsec ... 406
Show L2TP tunnel status ... 407
L2TPv3 Ethernet ... 408
Configure an L2TPv3 tunnel ... 409
Show L2TPV3 tunnel status ... 413
NEMO ... 414
Configure a NEMO tunnel ... 415
Show NEMO status ... 420

Services

Allow remote access for web administration and SSH ... 423
Configure the web administration service ... 427
Configure SSH access ... 437
Use SSH with key authentication ... 445
Generating SSH key pairs ... 445
Configure telnet access ... 448
Configure DNS ... 453
Show DNS server ... 459
Simple Network Management Protocol (SNMP) ... 460

SNMP Security ... 460
Configure Simple Network Management Protocol (SNMP) ... 460
Download MIBs ... 465
Location information ... 466
Configure the location service ... 467
Configure the device to use a user-defined static location ... 469
Configure the device to accept location messages from external sources ... 471
Forward location information to a remote host ... 475
Configure geofencing ... 482
Show location information ... 494
Modbus gateway ... 495
Configure the Modbus gateway ... 496
Show Modbus gateway status and statistics ... 509
System time ... 512
Configure the system time ... 512
Manually set the system date and time ... 516
Network Time Protocol ... 516
Configure the device as an NTP server ... 517
Show status and statistics of the NTP server ... 522
Configure a multicast route ... 523
Enable service discovery (mDNS) ... 526
Use the iPerf service ... 529
Example performance test using iPerf3 ... 534
Configure the ping responder service ... 535
Example performance test using iPerf3 ... 538

Applications

Develop Python applications ... 541
Set up the IX15 for Python development ... 541
Create and test a Python application ... 542
End-to-end demos ... 543
Python modules ... 547
Set up the IX15 to automatically run your applications ... 580
Configure scripts to run automatically ... 581
Show script information ... 588
Stop a script that is currently running ... 588
Start an interactive Python session ... 589
Run a Python application at the shell prompt ... 590
Install third party Python modules ... 592
Python migration guide ... 592
Programming IDE ... 592
Python version ... 593
Deployment ... 617

User authentication

IX15 user authentication ... 620
User authentication methods ... 620
Add a new authentication method ... 622
Delete an authentication method ... 624
Rearrange the position of authentication methods ... 626
Authentication groups ... 628
Change the access rights for a predefined group ... 629

Add an authentication group ... 631
Delete an authentication group ... 635
Local users ... 637
Change a local user's password ... 638
Configure a local user ... 640
Delete a local user ... 648
Terminal Access Controller Access-Control System Plus (TACACS+) ... 651
TACACS+ user configuration ... 652
TACACS+ server failover and fallback to local authentication ... 653
Configure your IX15 device to use a TACACS+ server ... 653
Remote Authentication Dial-In User Service (RADIUS) ... 658
RADIUS user configuration ... 659
RADIUS server failover and fallback to local configuration ... 659
Configure your IX15 device to use a RADIUS server ... 660
LDAP ... 664
LDAP user configuration ... 665
LDAP server failover and fallback to local configuration ... 666
Configure your IX15 device to use an LDAP server ... 666
Configure serial authentication ... 671
Disable shell access ... 673
Set the idle timeout for IX15 users ... 675
Example user configuration ... 678
Example 1: Administrator user with local authentication ... 678
Example 2: RADIUS, TACACS+, and local authentication for one user ... 680

Firewall

Firewall configuration ... 688
Create a custom firewall zone ... 688
Configure the firewall zone for a network interface ... 690
Delete a custom firewall zone ... 692
Port forwarding rules ... 693
Configure port forwarding ... 693
Delete a port forwarding rule ... 698
Packet filtering ... 701
Configure packet filtering ... 701
Enable or disable a packet filtering rule ... 705
Delete a packet filtering rule ... 707
Configure custom firewall rules ... 709
Configure Quality of Service options ... 711

Containers

Upload a new LXC container ... 722
Configure a container ... 723
Starting and stopping the container ... 726
Starting the container ... 726
Stopping the container ... 727
View the status of containers ... 727
Show status of all containers ... 728
Show status of a specific container ... 728
Schedule a script to run in the container ... 729
Create a custom container ... 731
Create the custom container file ... 731

Test the custom container file ... 732

System administration

Review device status ... 735
Configure system information ... 736
Update system firmware ... 738
Manage firmware updates using Digi Remote Manager ... 738
Certificate management for firmware images ... 739
Downgrading ... 739
Dual boot behavior ... 743
Update cellular module firmware ... 744
Update modem firmware over the air (OTA) ... 745
Update modem firmware by using a local firmware file ... 746
Reboot your IX15 device ... 747
Reboot your device immediately ... 747
Schedule reboots of your device ... 748
Erase device configuration and reset to factory defaults ... 750
Configure the IX15 device to use custom factory default settings ... 753
Locate the device by using the Find Me feature ... 755
Configuration files ... 757
Save configuration changes ... 757
Save configuration to a file ... 758
Restore the device configuration ... 759
Schedule system maintenance tasks ... 762
Disable device encryption ... 767
Re-enable cryptography after it has been disabled. ... 767
Configure the speed of your Ethernet port ... 769

Monitoring

intelliFlow ... 773
Enable intelliFlow ... 773
Use intelliFlow to display average CPU and RAM usage ... 776
Use intelliFlow to display top data usage information ... 777
Use intelliFlow to display data usage by host over time ... 779
Configure NetFlow Probe ... 780

Central management

Digi Remote Manager support ... 786
Certificate-based enhanced security ... 786
Configure Digi Remote Manager ... 786
Collect device health data and set the sample interval ... 793
Enable event log upload to Digi Remote Manager ... 796
Log into Digi Remote Manager ... 798
Use Digi Remote Manager to view and manage your device ... 799
Add a device to Digi Remote Manager ... 800
View Digi Remote Manager connection status ... 800
Configure multiple devices using profiles ... 801
Amazon AWS IoT ... 802
Microsoft Azure ... 802
Learn more ... 802

File system

The IX15 local file system 804
Display directory contents 804
Create a directory 805
Display file contents 806
Copy a file or directory 806
Move or rename a file or directory 807
Delete a file or directory 808
Upload and download files 809
- Upload and download files by using the WebUI 809
- Upload and download files by using the Secure Copy command 810
- Upload and download files using SFTP 811

Command line interface

Access the command line interface 814
Log in to the command line interface 814
Exit the command line interface 815
Execute a command from the web interface 815
Display help for commands and parameters 816
- The help command 816
- The question mark (?) command 816
- Display help for individual commands 817
- Use the Tab key or the space bar to display abbreviated help 818

Auto-complete commands and parameters 818
Available commands 819
XBee-specific commands 820
- Manage an XBee network 821
- Configure individual XBee parameters 823
- Apply XBee profiles 826

Use the scp command 827
Display status and statistics using the show command 829
- show config 829
- show system 829
- show network 830

Device configuration using the command line interface 830
Execute configuration commands at the root Admin CLI prompt 830
- Display help for the config command from the root Admin CLI prompt 831

Configuration mode 832
- Enable configuration mode 832
- Enter configuration commands in configuration mode 833
- Save changes and exit configuration mode 833
- Exit configuration mode without saving changes 834
- Configuration actions 834
- Display command line help in configuration mode 834
- Move within the configuration schema 837
- Manage elements in lists 838
- The revert command 840
- Enter strings in configuration commands 842
- Example: Create a new user by using the command line 842

Install Python packages 844
Command line reference 846
- analyzer clear 847

analyzer save ... 848
analyzer start ... 848
analyzer stop ... 848
clear dhcp-lease ip-address ... 848
clear dhcp-lease mac ... 849
container create ... 849
container delete ... 849
cp ... 849
help ... 850
ls ... 851
mkdir ... 852
modem at ... 852
modem at-interactive ... 852
modem firmware check ... 852
modem firmware list ... 852
modem firmware ota check ... 853
modem firmware ota list ... 853
modem firmware ota update ... 853
modem firmware update ... 853
modem pin change ... 854
modem pin disable ... 854
modem pin enable ... 854
modem pin status ... 855
modem pin unlock ... 855
modem puk status ... 855
modem puk unlock ... 855
modem reset ... 856
modem scan ... 856
modem sim-slot ... 856
monitoring ... 856
monitoring metrics upload ... 857
more ... 857
mv ... 857
ping ... 857
powerctrl ... 858
pyinstall ... 858
reboot ... 859
rm ... 860
scp ... 861
show analyzer ... 861
show arp ... 861
show cloud ... 861
show config ... 862
show containers ... 862
show dhcp-lease ... 862
show dns ... 862
show event ... 862
show hotspot ... 863
show ipsec ... 863
show l2tp lac ... 863
show l2tp lns ... 863
show l2tpeth ... 864
show location ... 864
show log ... 864
show manufacture ... 864

show modbus-gateway 864
show modem 865
show nemo 865
show network 865
show ntp 865
show openvpn client 866
show openvpn server 866
show route 866
show serial 866
show scripts 867
show surelink interface 867
show surelink ipsec 867
show surelink openvpn 867
show system 868
show usb 868
show version 868
show vrrp 868
show web-filter 868
speedtest 869
ssh 869
system backup 869
system disable-cryptography 870
system duplicate-firmware 870
system factory-erase 870
system find-me 870
system firmware ota check 871
system firmware ota list 871
system firmware ota update 871
system firmware update 871
system power ignition off_delay 871
system restore 872
system script start 872
system script stop 872
system serial clear 872
system serial save 873
system serial show 873
system serial start 873
system serial stop 873
system support-report 873
system time set 874
system time sync 874
system time test 874
telnet 874
traceroute 875
xbee 875

Diagnostics

Perform a speedtest 880
Generate a support report 880
View system and event logs 882
View System Logs 882
View Event Logs 884
Configure syslog servers 886
Configure options for the event and system logs 889

Analyze network traffic ... 894
Configure packet capture for the network analyzer ... 895
Example filters for capturing data traffic ... 904
Capture packets from the command line ... 905
Stop capturing packets ... 906
Show captured traffic data ... 907
Save captured data traffic to a file ... 908
Download captured data to your PC ... 909
Clear captured data ... 910
Use the ping command to troubleshoot network connections ... 912
Ping to check internet connection ... 912
Stop ping commands ... 912
Use the traceroute command to diagnose IP routing problems ... 912

Server Command Interface (SCI)

Discover and retrieve the XBee network ... 915
Retrieve XBee device settings ... 916
Configure XBee device settings ... 918
Retrieve XBee device settings ... 918
Execute arbitrary commands ... 919
Read the value of an XBee setting ... 919
Set the value of an XBee setting ... 920
Execute a special XBee command ... 921
Update an XBee device profile ... 922
Reset an XBee device to factory defaults ... 923

Troubleshooting

System log ... 926
Configure XBee log level ... 926
Display the system log ... 926
Recover the local XBee ... 927
xbeemgmt tool ... 927

FAQ

Get the IX15 IP ... 930
A remote XBee is not listed in the IX15 network ... 930
PyCharm: My IX15 is not listed in Digi Device Selector ... 930

Digi IX15 regulatory and safety statements

RF exposure statement ... 933
FCC (USA) exposure notice ... 933
FCC Part 15 Class A ... 933
Radio Frequency Interference (RFI) (FCC 15.105) ... 933
European Community - CE Mark Declaration of Conformity (DoC) ... 933
Maximum transmit power for radio frequencies ... 934
RoHS compliance statement ... 935
ISED (Innovation, Science and Economic Development Canada) ... 935
RF Exposure ... 935
Antennas ... 935

Japan (TELEC) 935
Safety statements 936
Digi IX15 Gateway Hazardous Locations information 937
    Special conditions for safe use 937
    Class I Division 2, Groups A,B,C,D Temperature Code: T4 937
Special safety notes for wireless routers 938
Product disposal instructions 938

Safety warnings

English 941
Bulgarian--български 942
Croatian--Hrvatski 943
French--Français 944
Greek--Ε λληνικά 945
Hungarian--Magyar 946
Italian--Italiano 947
Latvian--Latvietis 948
Lithuanian--Lietuvis 949
Polish--Polskie 950
Portuguese--Português 951
Slovak--Slovák 952
Slovenian--Esloveno 953
Spanish--Español 954

Digi IX15 certifications

International EMC (Electromagnetic Compatibility) and safety standards 955

Digi IX15 Gateway User Guide

The Digi IX15 is a rugged, secure and reliable LTE industrial router powered by an enhanced operating system that supports any utility or industrial application.

This online guide helps site administrators configure and manage Digi IX15 devices. This guide assumes administrators are familiar with network basics, such as network terminology, architecture, interfaces, and related concepts.

Overview 20
What's new in Digi IX15 version 22.2 20
IX15 compatibility with S2C XBee devices 21
Related documents 26

Overview

The Digi IX15 Gateway allows for the provisioning and management of an XBee network and other industrial devices connected to it through the WebUI and CLI.

The IX15 provides a programmable solution to connect networks of XBee-enabled devices to IP networks. With a simple, open-source Python development environment, this gateway enables custom applications to run locally while interfacing across existing Ethernet/Wi-Fi/cellular networks for WAN connectivity to cloud-based software applications.

The IX15 supports 2.4 GHz XBee networks:

- Zigbee
- DigiMesh
- 802.15.4

And is compatible with the following XBee hardware:

- XBee 3
- XBee S2C. See IX15 compatibility with S2C XBee devices.

This documentation helps you configure and manage Digi IX15 devices to

- Manage your XBee network
- Program XBee profiles on local or remote devices
- Develop Python applications with PyCharm and deploy them in the gateway
- Use a command line interface to interact with the settings of your XBee devices

What's new in Digi IX15 version 22.2

Initial release of the *Digi IX15 Gateway User Guide*.

Release of Digi IX15 firmware version 22.2:

- VPN enhancements:
  - Renamed **VPN** > **IPsec** > **Tunnels** > **Policies** > **Local network setting** to **Local traffic selector** and added **Remote traffic selector**.
  - Added a **Dynamic** option to the **Local traffic selector** to configuration of a local network by protocol and/or port instead of a network address range.
  - Added Protocol and Port match criteria for **Local traffic selector** and **Remote traffic selector**.
  - Added **VPN** > **IPsec** > **Advanced** > **Debug level** to specify the logging verbosity of IPsec messages in the device system logs.
- Enhancements to communications with Digi Remote Manager:
  - Enhanced security for communications with Digi Remote Manager by using client-side certificates.
  - The default URL for the device's Remote Manager connection is now edp12.devicecloud.com. This URL is required to utilize the client-side certificate support.
- New **Switch SIM** SureLink action for WWAN interfaces, which allows SureLink to be configured to switch to the alternate SIM if the modem is connected but SureLink test are failing.

    - New **Switch SIM fail count** option to determine how many SureLink failures are required prior to switching to the alternate SIM.
- New **Socket ID string** option to send the configured text to the remote server(s) when a TCP socket connection is opened to the serial port.
- New **cat** Admin CLI command for displaying file contents.

IX15 compatibility with S2C XBee devices

Digi has updated and ported our XBee firmware from the XBee/XBee-PRO (S2C) hardware based on the SiLabs EM357 SoC, to the Digi XBee 3 hardware based on the SiLabs EFR32 SoC.

The Digi IX15 Gateway includes an XBee 3 device that can be configured to work with Zigbee, DigiMesh, or 802.15.4 networks.

This section assists you with configuring your Digi IX15 Gateway to work within a network of XBee/XBee-PRO S2C devices. While the basic functionality and communication are similar and compatible, there are some limitations to consider. The main limitations are:

1. Zigbee
    a. The Digi IX15 Gateway should enable LQI compatibility mode—**C8**—so that all nodes in the network have the same priority when determining route costs.
    b. The Digi IX15 Gateway does not open the join window persistently.
    c. An XBee S2C OTA firmware update from an IX15 is possible when S2C devices are within range of another S2C device:
        - S2C end devices must have an S2C node as their parent.
        - S2C routers must have at least another S2C router node in range.
2. DigiMesh
    a. The channel—**CH**—on the IX15 should be within 0x0C and 0x17 to guarantee communication with XBee S2C devices.
    b. If using encryption, the IX15 should enable 128-bit key for AES Encryption—**C8**—to be over-the-air compatible with S2C devices.
    c. Synchronous sleep mode is not supported in XBee S2C.
    d. XBee S2C OTA firmware update from an IX15 is possible when S2C devices are within range of another S2C device.
3. 802.15.4
    a. The channel—**CH**—on the IX15 should be within 0x0C and 0x17 to guarantee communication with XBee S2C devices.
    b. An IX15 configured as the indirect messaging coordinator will not hold messages for longer than 65 seconds.
    c. XBee S2C OTA firmware update from an IX15 is not supported.

Zigbee

1. Power cap on channel 26

    On the XBee S2C, output power on channel 26 is capped to +3 dBm for all device roles. Additionally, including channel 26 in the scan channel mask—**SC**—of the coordinator would reduce the output power on all channels to +3 dBm.

The power cap on IEEE 802.15.4 channel 26 has been increased on XBee 3 and restrictions on the coordinator have been alleviated.

The power cap on channel 26 of the XBee 3 is increased to +8 dBm. Including channel 26 in the scan channel mask has no impact to other channels. The default value for **SC** is **0x7FFF**—Channels 11 through 25.

2. LQI calculations

   On the S2C, the LQI curve is more flat and trends towards the top of the scale; where the LQI curve on the XBee 3 is linear and lower scale. As a result, if operating in a network with a mixture of XBee 3 and S2C modules, the S2C devices will have higher priority when determining route costs.

   An LQI compatibility mode was introduced in XBee 3 Zigbee version 1009, which can optionally have the XBee 3 operate with a similar LQI curve as the S2C. **C8** bit 4—**C8** | 0x10—on XBee 3 enables the LQI compatibility mode. The Digi IX15 Gateway should enable LQI compatibility mode.

3. End Device timeout

   On the S2C, the parent node determines how long an end device exists in the child timeout table. The timeout value was based on the **SP** and **SN** parameters on the parent node.

   With XBee 3, the end device controls this timeout value. A new **ET** command—which is only set on an End Device—sets the child table timeout when it joins its parent. The timeout is a minimum of two minutes, but can be set as long as 273 hours.

4. Join window is not persistently open

   Zigbee 3.0 does not support an open joining model. To meet this requirement, the **NJ** command has a new default setting of **NJ** = **0xFE**. This generous joining window still allows other devices to join to the network, but the join window will close after 254s. The join window can be reopened again at any time by pressing the commissioning button twice, or issuing a **CB2** AT command.

   To keep the join window open indefinitely, you must explicitly set **NJ** = **FF**.

5. Join window indicators

   If the device is operating in API mode—**AP** = **1** or **2**—Modem Statuses are emitted when the join window opens and closes:

   - 0x43 - Open Join Window
   - 0x44 - Closed Join Window

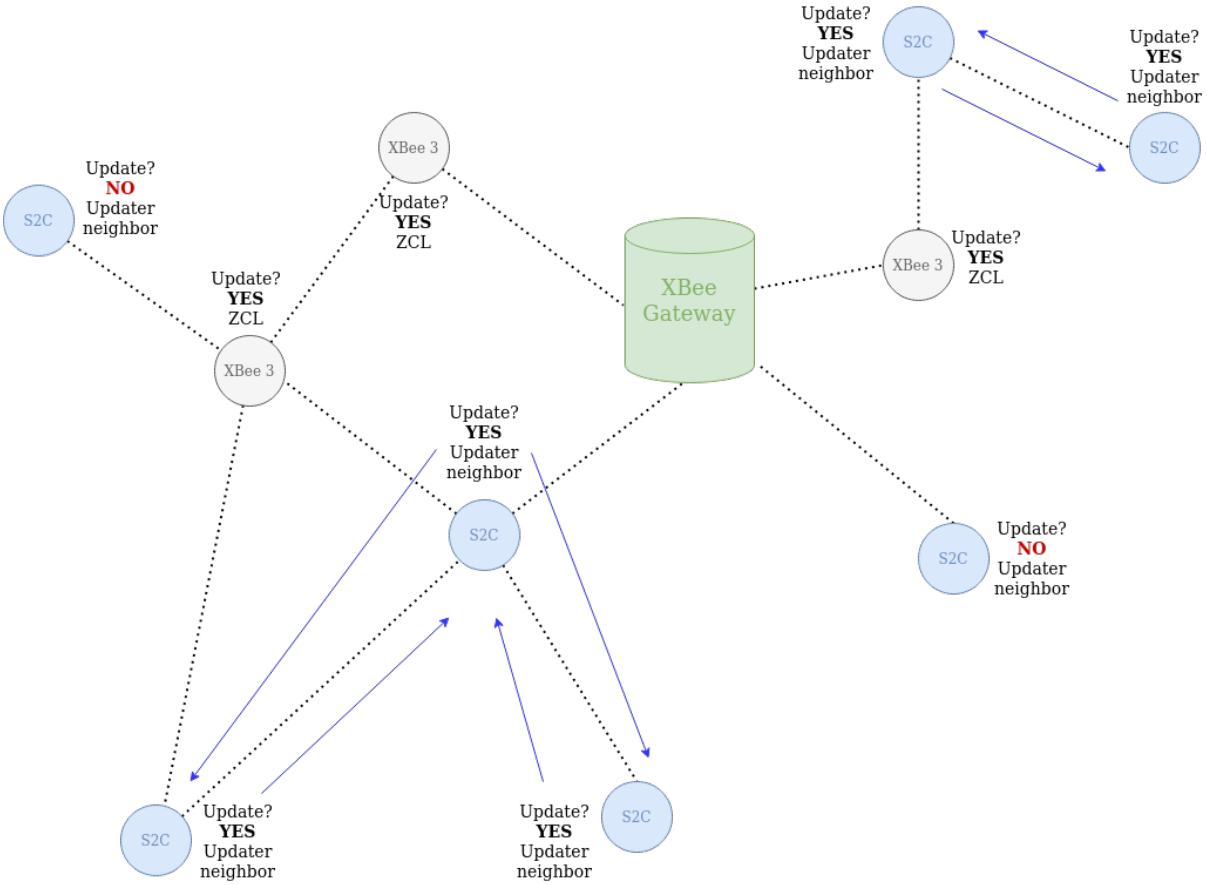
6. OTA firmware update process

   The radio serving up the firmware image can be either a Digi XBee 3 Zigbee 3.0 or XBee/XBee-PRO ZB (S2C). However, since the XBee/XBee-PRO ZB (S2C) requires that another XBee/XBee-PRO ZB (S2C) be a nearest neighbor to act as the updater node, there is some limitation as to how nodes can be updated in a mixed network.

   The image below describes how it is determined whether nodes in a mixed network arrangement can be updated from a single firmware image server. The image server can be either Digi XBee 3 Zigbee 3.0 or XBee/XBee-PRO ZB (S2C). S2C devices must be within range of

another S2C device to be updated:

- S2C end devices must have an S2C node as their parent.
- S2C routers must have at least another S2C router node in range.



Note For more information, see the *Digi XBee 3 Zigbee Migration Guide*.

DigiMesh

1. Expanded channel selection

   The IEEE 802.15.4 standard allows for 16 channels to be used for communication: from 2.405 GHz—Channel 11—through 2.480 GHz—Channel 26. On the previous XBee-PRO S2C DigiMesh modules, the range of available channels—via the **CH** command—was restricted on the PRO variant. This restriction is alleviated on the XBee 3 and all 16 channels are available for use regardless of the variant.

   The **CH** parameter on the IX15 should be within 0x0C and 0x17 to guarantee communication.

2. Synchronized cyclic sleep

XBee S2C DigiMesh devices lacked the hardware necessary to keep the network in sync over long periods of time, so no support was included. This feature has now been reintroduced on XBee 3.

3. Enhanced security

   On the previous XBee S2C DigiMesh modules, 128-bit AES encryption was used to secure RF traffic. The default security on the XBee 3, when enabled, is 256-bit AES with counter mode to protect against replay attacks. As a result, the **KY** parameter has been increased to a 256-bit value—64 ASCII characters—on the XBee 3.

   In order to provide compatibility with legacy DigiMesh networks, a bit in the **C8** command has been created. Bit 2 of the **C8** bitfield—**C8** | 0x4—when set, reduces the security mode of the XBee 3 to 128-bit AES; this is over-the-air compatible with legacy 2.4 GHz DigiMesh devices.

   When this compatibility mode is enabled, only the last 32 ASCII chars—128 least significant bits—of the 256-bit **KY** parameter will be used. **KY** cannot be read and is write-only.

   Examples:

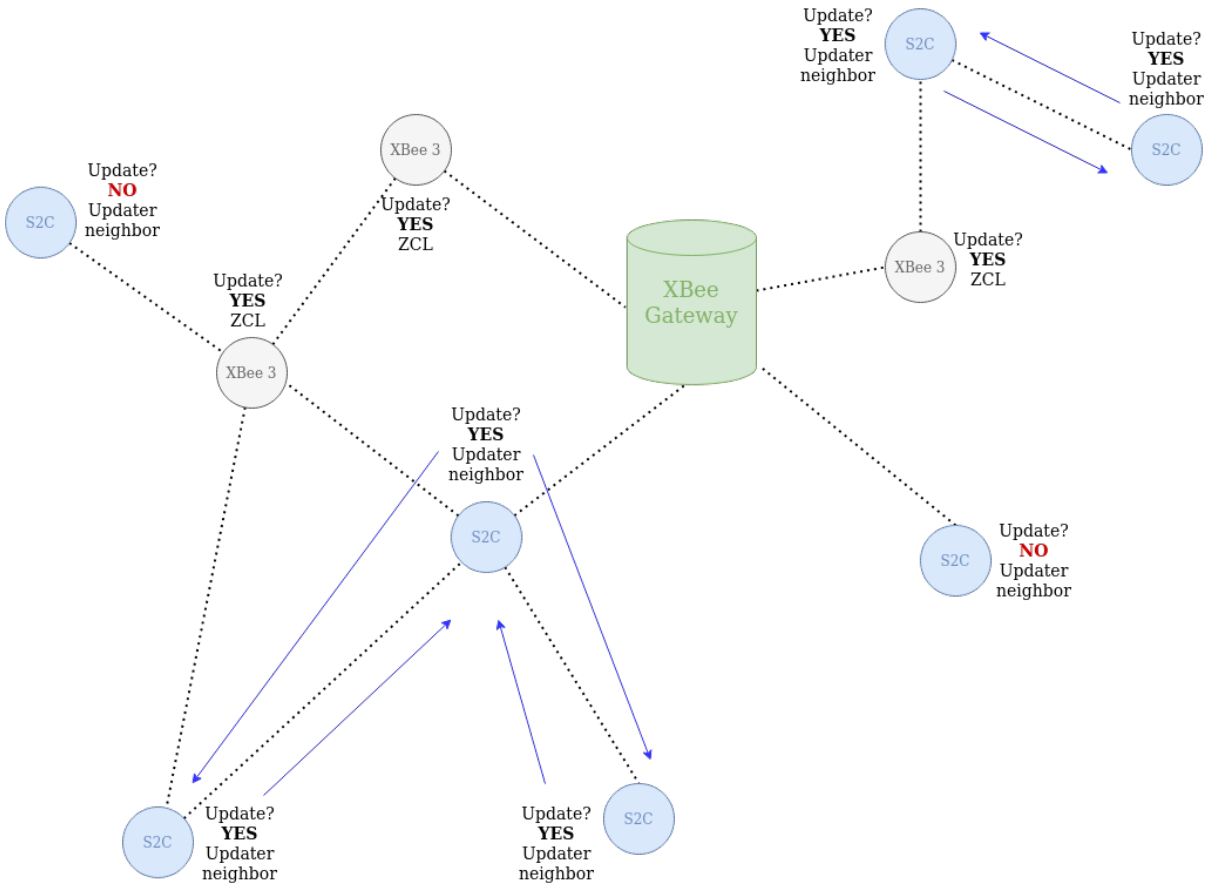
   - If **EE** = **1** and **C8** = **0**, then **KY** is a 256-bit Link key and will only communicate with other XBee 3 DigiMesh devices with the same **C8** and **KY** values.
   - If **EE** = **1** and **C8** = **4**, then K**Y** is a 128-bit Link key that is compatible with S1, S2C, and XB3 DigiMesh using with the same 128-bit **KY** value.

   If **KY** =1111222233334444555566667777888ZZZZYYYYXXXXWWWWVVVVUUUUTTTTSSSS, the underlined portion would be the 128-bit link key.

4. OTA firmware update process

   The radio serving up the firmware image can be either a Digi XBee 3 DigiMesh or XBee/XBee-PRO DM(S2C). However, since the XBee/XBee-PRO DM (S2C) requires that another XBee/XBee-PRO DM(S2C) be a nearest neighbor to act as the updater node, there is some limitation as to how nodes can be updated in a mixed network.

   The image below describes how it is determined whether nodes in a mixed network arrangement can be updated from a single firmware image server. The image server can be either Digi XBee 3 DigiMesh or XBee/XBee-PRO DM(S2C). S2C modules must be within range of another S2C module to be updated.



Note For more information, see the *Digi XBee 3 DigiMesh Migration Guide*.

802.15.4

1. Expanded channel selection

   The IEEE 802.15.4 standard allows for 16 channels to be used for communication: from 2.405 GHz—Channel 11—through 2.480 GHz—Channel 26. On the previous XBee-PRO S2C 802.15.4 modules, the range of available channels—via the **CH** command—was restricted on the PRO variant. This restriction is alleviated on the XBee 3 and all 16 channels are available for use regardless of the variant.

   The **CH** parameter on the IX15 needs to be within **0x0C** and **0x17** to communicate.

2. Indirect messaging limitation

   SP determines the asynchronous cyclic sleep period, the same as S2C 802.15.4. However, on the XBee 3, the indirect messaging coordinator will not hold messages for longer than 65 seconds, even if **SP** * 2.5 > 65 seconds.

3. OTA firmware update process

The OTA firmware update process for the XBee 3 802.15.4 is different than the process on the S2C. When performing an OTA firmware update, both the server and client nodes must be of the same type—XBee 3 to XBee 3, S2C to S2C. This means remote firmware update is not supported from the IX15 to remote XBee S2C 802.15.4 devices.

Note For more information, see the *Digi XBee 3 802.15.4 Migration Guide*.

Related documents

This guide contains the information you need to start working with an IX15. For more detailed information on protocols and related libraries, see:

- *XBee 3 802.15.4 RF Module User Guide*
- *XBee 3 DigiMesh RF Module User Guide*
- *XBee 3 Zigbee RF Module User Guide*
- *Digi XBee PyCharm IDE Plugin User Guide*
- *Digi MicroPython Programming Guide*
- *Digi XBee 802.15.4 Protocol Comparison*
- *Digi XBee 802.15.4 Protocol Comparison*
- *Digi XBee 3 802.15.4 Migration Guide*
- *Digi XBee 3 DigiMesh Migration Guide*
- *Digi XBee 3 Zigbee Migration Guide*
- *PyCharm documentation*
- *XCTU User Guide*
- *XBee Python Library*

Safety instructions

Safety instructions 28
Инструкции за безопасност 28
Sigurnosne upute 29
Bezpečnostní instrukce 29
Sikkerhedsinstruktioner 30
Veiligheidsinstructies 31
Ohutusjuhised 31
Turvallisuusohjeet 32
Consignes de sécurité 32
Sicherheitshinweise 33
Οδηγίες ασφαλείας 34
Biztonsági utasítások 34
Istruzioni di sicurezza 35
Drošības instrukcijas 36
Saugos instrukcijos 36
Sikkerhetsinstruksjoner 37
Instrukcje bezpieczeństwa 37
Instruções de segurança 38
Instructiuni de siguranta 39
Bezpečnostné inštrukcie 39
Varnostna navodila 40
Adaptador XBee, puertas de enlace y enrutadores 40
Säkerhets instruktioner 41

Safety instructions

XBee adapter, gateways, and routers

- The XBee Adapter, Gateway, or Router products cannot be guaranteed operation due to the radio link and so should not be used for interlocks in safety critical devices such as machines or automotive applications.
- The XBee Adapter, Gateway, or Router products have not been approved for use in (this list is not exhaustive):
  - medical devices
  - nuclear applications
  - explosive or flammable atmospheres
- There are no user serviceable components inside the XBee Adapter, Gateway, or Router product. Do not remove the product covers or modify the Gateway or Router in any way. Modifications may exclude the product from any warranty and can cause the gateway or router to operate outside of regulatory compliance for a given country, leading to the possible illegal operation of the product.
- Use industry standard ESD protection when handling the XBee Adapter, Gateway, or Router product.
- Take care while handling to avoid electrical damage to the PCB and components.
- Do not expose the XBee Adapter, Gateway, or Router products to water or moisture.
- Use this product with the antennas specified in the XBee Adapter, Gateway, or Router product user guides.
- The end user must be told how to remove power from the XBee Adapter, Gateway, or Router product or to locate the antennas 20 cm from humans or animals.

Инструкции за безопасност

XBee адаптер, шлюзове и рутери

- Продуктите XBee Adapter, Gateway или Router не могат да бъдат гарантирани за работа поради радиовръзката и затова не трябва да се използват за блокировки в устройства, които са важни за безопасността, като машини или автомобилни приложения.
- Продуктите XBee Adapter, Gateway или Router не са одобрени за използване в (този списък не е изчерпателен):
  - медицински изделия
  - ядрени приложения
  - експлозивна или запалима атмосфера
- В продукта XBee Adapter, Gateway или Router няма компоненти, които могат да бъдат обслужвани от потребителя. Не сваляйте капаците на продукта и не модифицирайте по никакъв начин шлюза или рутера. Модификациите могат да изключат продукта от гаранция и могат да доведат до работа на шлюза или рутера извън регулаторното съответствие за дадена държава, което води

до възможна незаконна работа на продукта.

- Използвайте стандартна ESD защита, когато работите с XBee адаптер, шлюз или рутер.
- Внимавайте, докато боравите, за да избегнете електрически повреди на печатната платка и компонентите.
- Не излагайте продуктите XBee Adapter, Gateway или Router на вода или влага.
- Използвайте този продукт с антените, посочени в ръководствата за потребителя на XBee Adapter, Gateway или Router.
- Крайният потребител трябва да бъде казано как да премахне захранването от XBee Adapter, Gateway или Router продукта или да разположи антените на 20 см от хора или животни.

Sigurnosne upute

XBee adapter, pristupnici i usmjerivači

- Ne može se jamčiti rad proizvoda XBee Adapter, Gateway ili Router zbog radio veze i stoga se ne bi trebali koristiti za blokade u sigurnosnim kritičnim uređajima kao što su strojevi ili automobilske aplikacije.
- XBee Adapter, Gateway ili Router proizvodi nisu odobreni za upotrebu u (ovaj popis nije konačan):
  - medicinskih uređaja
  - nuklearne primjene
  - eksplozivne ili zapaljive atmosfere
- Unutar proizvoda XBee Adapter, Gateway ili Router nema komponenti koje može servisirati korisnik. Nemojte skidati poklopce proizvoda niti na bilo koji način mijenjati pristupnik ili usmjerivač. Izmjene mogu isključiti proizvod iz bilo kakvog jamstva i mogu uzrokovati rad pristupnika ili usmjerivača izvan usklađenosti s propisima za određenu zemlju, što može dovesti do mogućeg nezakonitog rada proizvoda.
- Koristite standardnu ESD zaštitu pri rukovanju proizvodom XBee Adapter, Gateway ili Router.
- Budite oprezni tijekom rukovanja kako biste izbjegli električna oštećenja PCB-a i komponenti.
- Ne izlažite XBee Adapter, Gateway ili Router proizvode vodi ili vlazi.
- Koristite ovaj proizvod s antenama navedenim u korisničkim vodičima proizvoda XBee Adapter, Gateway ili Router.
- Krajnjem korisniku se mora reći kako da isključi napajanje iz XBee adaptera, Gatewaya ili Routera proizvoda ili da locira antene 20 cm od ljudi ili životinja.

Bezpečnostní instrukce

XBee adaptér, brány a routery

- U produktů XBee Adapter, Gateway nebo Router nelze zaručit provoz kvůli rádiovému spojení, a proto by neměly být používány pro blokování v zařízeních kritických z hlediska bezpečnosti, jako jsou stroje nebo automobilové aplikace.

- Produkty XBee Adapter, Gateway nebo Router nebyly schváleny pro použití v (tento seznam není vyčerpávající):
  - zdravotnické prostředky
  - jaderné aplikace
  - výbušné nebo hořlavé atmosféry
- Uvnitř produktu XBee Adapter, Gateway nebo Router nejsou žádné uživatelsky opravitelné součásti. Neodstraňujte kryty produktu ani žádným způsobem neupravujte bránu nebo směrovač. Úpravy mohou vyjmout produkt z jakékoli záruky a mohou způsobit, že brána nebo router bude fungovat mimo zákonnou shodu pro danou zemi, což povede k možnému nezákonnému provozu produktu.
- Při manipulaci s produktem XBee Adapter, Gateway nebo Router používejte standardní ochranu ESD.
- Při manipulaci buďte opatrní, aby nedošlo k elektrickému poškození desky plošných spojů a součástí.
- Nevystavujte produkty XBee Adapter, Gateway nebo Router vodě nebo vlhkosti.
- Používejte tento produkt s anténami uvedenými v uživatelské příručce produktu XBee Adapter, Gateway nebo Router.
- Koncový uživatel musí být informován, jak odpojit napájení XBee adaptéru, brány nebo routeru nebo jak umístit antény 20 cm od lidí nebo zvířat.

Sikkerhedsinstruktioner

XBee-adapter, gateways og routere

- XBee Adapter-, Gateway- eller Router-produkterne kan ikke garanteres drift på grund af radioforbindelsen og bør derfor ikke bruges til aflåsninger i sikkerhedskritiske enheder såsom maskiner eller bilapplikationer.
- XBee Adapter, Gateway eller Router-produkter er ikke blevet godkendt til brug i (denne liste er ikke udtømmende):
  - medicinsk udstyr
  - nukleare applikationer
  - eksplosive eller brandfarlige atmosfærer
- Der er ingen komponenter, der kan repareres af brugeren, inde i XBee Adapter-, Gateway- eller Router-produktet. Fjern ikke produktdækslerne, og modificer ikke gatewayen eller routeren på nogen måde. Ændringer kan udelukke produktet fra enhver garanti og kan få gatewayen eller routeren til at fungere uden for lovgivningsoverholdelse for et givet land, hvilket kan føre til en mulig ulovlig drift af produktet.
- Brug industristandard ESD-beskyttelse, når du håndterer XBee Adapter-, Gateway- eller Router-produktet.
- Vær forsigtig under håndteringen for at undgå elektrisk beskadigelse af printet og komponenterne.
- Udsæt ikke XBee Adapter, Gateway eller Router-produkter for vand eller fugt.
- Brug dette produkt med de antenner, der er specificeret i brugervejledningerne til XBee Adapter, Gateway eller Router.

- Slutbrugeren skal fortælles, hvordan man fjerner strømmen fra XBee Adapter-, Gateway- eller Router-produktet eller placerer antennerne 20 cm fra mennesker eller dyr.

Veiligheidsinstructies

XBee-adapter, gateways en routers

- De werking van de XBee-adapter-, gateway- of routerproducten kan niet worden gegarandeerd vanwege de radioverbinding en mogen daarom niet worden gebruikt voor vergrendelingen in veiligheidskritieke apparaten zoals machines of autotoepassingen.
- De XBee Adapter-, Gateway- of Router-producten zijn niet goedgekeurd voor gebruik in (deze lijst is niet uitputtend):
  - medische apparaten
  - nucleaire toepassingen
  - explosieve of ontvlambare atmosferen
- Er zijn geen door de gebruiker te onderhouden componenten in het XBee Adapter-, Gateway- of Routerproduct. Verwijder de productkappen niet en wijzig de gateway of router op geen enkele manier. Wijzigingen kunnen het product uitsluiten van elke garantie en kunnen ertoe leiden dat de gateway of router buiten de regelgeving voor een bepaald land werkt, wat kan leiden tot de mogelijke illegale werking van het product.
- Gebruik industriestandaard ESD-bescherming bij het hanteren van het XBee Adapter-, Gateway- of Routerproduct.
- Wees voorzichtig bij het hanteren om elektrische schade aan de printplaat en componenten te voorkomen.
- Stel de XBee Adapter-, Gateway- of Routerproducten niet bloot aan water of vocht.
- Gebruik dit product met de antennes die zijn gespecificeerd in de gebruikershandleidingen van de XBee Adapter, Gateway of Router.
- De eindgebruiker moet worden verteld hoe de voeding van de XBee Adapter, Gateway of Router moet worden uitgeschakeld of hoe de antennes op 20 cm van mensen of dieren moeten worden geplaatst.

Ohutusjuhised

XBee adapter, lüüsid ja ruuterid

- XBee adapteri, lüüsi või ruuteri toodete toimimist ei saa raadiolingi tõttu tagada ja seetõttu ei tohiks neid kasutada ohutuse seisukohalt oluliste seadmete (nt masinad või autorakendused) blokeerimiseks.
- XBee adapteri, lüüsi või ruuteri tooteid ei ole heaks kiidetud kasutamiseks järgmistes riikides (see loend ei ole ammendav):
  - meditsiiniseadmed
  - tuumarakendused
  - plahvatusohtlik või tuleohtlik keskkond
- XBee adapteris, lüüsis või ruuteris ei ole kasutaja poolt hooldatavaid komponente. Ärge eemaldage toote katet ega muutke lüüsi ega ruuterit mingil viisil. Muudatused võivad toote

garantiist välja jätta ja põhjustada lüüsi või ruuteri toimimise väljaspool antud riigi regulatiivset vastavust, mis võib viia toote ebaseadusliku kasutamiseni.

- Kasutage XBee adapteri, lüüsi või ruuteri toote käsitsemisel tööstusharu standardset ESD-kaitset.
- Olge käsitsemisel ettevaatlik, et vältida PCB ja komponentide elektrikahjustusi.
- Ärge jätke XBee adapteri, lüüsi või ruuteri tooteid vee või niiskuse kätte.
- Kasutage seda toodet XBee adapteri, lüüsi või ruuteri toote kasutusjuhendis kirjeldatud antennidega.
- Lõppkasutajale tuleb öelda, kuidas eemaldada XBee adapteri, lüüsi või ruuteri toide või leida antennid inimestest või loomadest 20 cm kaugusel.

Turvallisuusohjeet

XBee-sovitin, yhdyskäytävät ja reitittimet

- XBee-sovitin-, yhdyskäytävä- tai reititintuotteiden toimintaa ei voida taata radiolinkin vuoksi, joten niitä ei tule käyttää turvallisuuden kannalta kriittisten laitteiden, kuten koneiden tai autosovelluksien, lukitsemiseen.
- XBee-sovitin-, yhdyskäytävä- tai reititintuotteita ei ole hyväksytty käytettäväksi (tämä luettelo ei ole tyhjentävä):
  - lääketieteelliset laitteet
  - ydinvoimasovellukset
  - räjähdysvaarallisiin tai syttyviin tiloihin
- XBee-sovittimen, yhdyskäytävän tai reitittimen sisällä ei ole käyttäjän huollettavia osia. Älä poista tuotteen kansia tai muokkaa yhdyskäytävää tai reititintä millään tavalla. Muutokset voivat sulkea tuotteen takuun ulkopuolelle ja saada yhdyskäytävän tai reitittimen toimimaan tietyn maan säännöstenmukaisuuden ulkopuolella, mikä voi johtaa tuotteen laittomaan käyttöön.
- Käytä alan standardia ESD-suojausta käsitellessäsi XBee-sovitinta, yhdyskäytävää tai reititintuotetta.
- Ole varovainen käsitellessäsi, jotta vältät piirilevyn ja komponenttien sähkövauriot.
- Älä altista XBee-sovitin-, yhdyskäytävä- tai reititintuotteita vedelle tai kosteudelle.
- Käytä tätä tuotetta XBee-sovittimen, yhdyskäytävän tai reitittimen tuotteen käyttöoppaissa määritettyjen antennien kanssa.
- Loppukäyttäjälle on kerrottava, kuinka XBee-sovittimen, yhdyskäytävän tai reitittimen virta katkaistaan tai antennit sijoitetaan 20 cm:n etäisyydelle ihmisistä tai eläimistä.

Consignes de sécurité

Adaptateur XBee, passerelles et routeurs

- Le fonctionnement des produits XBee Adapter, Gateway ou Router ne peut pas être garanti en raison de la liaison radio et ne doit donc pas être utilisé pour les verrouillages dans des dispositifs critiques pour la sécurité tels que des machines ou des applications automobiles.

- Les produits XBee Adapter, Gateway ou Router n'ont pas été approuvés pour une utilisation dans (cette liste n'est pas exhaustive) :
  - dispositifs médicaux
  - applications nucléaires
  - atmosphères explosives ou inflammables
- Il n'y a aucun composant réparable par l'utilisateur à l'intérieur du produit XBee Adapter, Gateway ou Router. Ne retirez pas les capots du produit et ne modifiez en aucune façon la passerelle ou le routeur. Les modifications peuvent exclure le produit de toute garantie et peuvent entraîner le fonctionnement de la passerelle ou du routeur en dehors de la conformité réglementaire pour un pays donné, conduisant à un éventuel fonctionnement illégal du produit.
- Utilisez la protection ESD standard de l'industrie lors de la manipulation de l'adaptateur, de la passerelle ou du routeur XBee.
- Soyez prudent lors de la manipulation afin d'éviter des dommages électriques au circuit imprimé et aux composants.
- N'exposez pas les produits XBee Adapter, Gateway ou Router à l'eau ou à l'humidité.
- Utilisez ce produit avec les antennes spécifiées dans les guides d'utilisation des adaptateurs, passerelles ou routeurs XBee.
- L'utilisateur final doit savoir comment couper l'alimentation de l'adaptateur, de la passerelle ou du routeur XBee ou comment placer les antennes à 20 cm des humains ou des animaux.

Sicherheitshinweise

XBee-Adapter, Gateways und Router

- Der Betrieb der XBee Adapter-, Gateway- oder Router-Produkte kann aufgrund der Funkverbindung nicht garantiert werden und sollte daher nicht für Verriegelungen in sicherheitskritischen Geräten wie Maschinen oder Automobilanwendungen verwendet werden.
- Die XBee Adapter-, Gateway- oder Router-Produkte wurden nicht für die Verwendung zugelassen in (diese Liste ist nicht vollständig):
  - Medizinprodukte
  - nukleare Anwendungen
  - explosive oder brennbare Atmosphären
- Es gibt keine vom Benutzer zu wartenden Komponenten innerhalb des XBee Adapter-, Gateway- oder Router-Produkts. Entfernen Sie nicht die Produktabdeckungen und modifizieren Sie das Gateway oder den Router nicht in irgendeiner Weise. Modifikationen können das Produkt von jeglicher Garantie ausschließen und dazu führen, dass das Gateway oder der Router außerhalb der gesetzlichen Vorschriften für ein bestimmtes Land betrieben wird, was zu einem möglicherweise illegalen Betrieb des Produkts führen kann.
- Verwenden Sie bei der Handhabung des XBee Adapter-, Gateway- oder Router-Produkts ESD-Schutz nach Industriestandard.
- Seien Sie vorsichtig bei der Handhabung, um elektrische Schäden an der Leiterplatte und den Komponenten zu vermeiden.

- Setzen Sie die Produkte XBee Adapter, Gateway oder Router weder Wasser noch Feuchtigkeit aus.
- Verwenden Sie dieses Produkt mit den Antennen, die in den Produktbenutzerhandbüchern für XBee-Adapter, -Gateways oder -Router angegeben sind.
- Dem Endbenutzer muss mitgeteilt werden, wie er die Stromversorgung des XBee-Adapters, Gateways oder Router-Produkts unterbricht oder die Antennen 20 cm von Menschen oder Tieren entfernt aufstellt.

Οδηγίες ασφαλείας

Προσαρμογέας XBee, πύλες και δρομολογητές

- Τα προϊόντα XBee Adapter, Gateway ή Router δεν είναι εγγυημένα για τη λειτουργία τους λόγω της ραδιοζεύξης και επομένως δεν πρέπει να χρησιμοποιούνται για ασφάλειες σε κρίσιμες για την ασφάλεια συσκευές, όπως μηχανήματα ή εφαρμογές αυτοκινήτου.
- Τα προϊόντα XBee Adapter, Gateway ή Router δεν έχουν εγκριθεί για χρήση σε (αυτή η λίστα δεν είναι εξαντλητική):
  - ιατροτεχνολογικά προϊόντα
  - πυρηνικές εφαρμογές
  - εκρηκτικές ή εύφλεκτες ατμόσφαιρες
- Δεν υπάρχουν εξαρτήματα που να μπορούν να επισκευαστούν από το χρήστη μέσα στο προϊόν XBee Adapter, Gateway ή Router. Μην αφαιρείτε τα καλύμματα του προϊόντος και μην τροποποιείτε την πύλη ή το δρομολογητή με κανέναν τρόπο. Οι τροποποιήσεις ενδέχεται να αποκλείουν το προϊόν από οποιαδήποτε εγγύηση και μπορεί να προκαλέσουν τη λειτουργία της πύλης ή του δρομολογητή εκτός της συμμόρφωσης με τους κανονισμούς για μια δεδομένη χώρα, οδηγώντας σε πιθανή παράνομη λειτουργία του προϊόντος.
- Χρησιμοποιήστε βιομηχανική προστασία ESD όταν χειρίζεστε το προϊόν XBee Adapter, Gateway ή Router.
- Προσέχετε κατά το χειρισμό για να αποφύγετε ηλεκτρική βλάβη στο PCB και στα εξαρτήματα.
- Μην εκθέτετε τα προϊόντα XBee Adapter, Gateway ή Router σε νερό ή υγρασία.
- Χρησιμοποιήστε αυτό το προϊόν με τις κεραίες που καθορίζονται στους οδηγούς χρήσης του XBee Adapter, Gateway ή Router.
- Πρέπει να ενημερωθεί ο τελικός χρήστης πώς να αφαιρέσει το ρεύμα από τον προσαρμογέα XBee, το Gateway ή το προϊόν του δρομολογητή ή να εντοπίσει τις κεραίες σε απόσταση 20 cm από ανθρώπους ή ζώα.

Biztonsági utasítások

XBee adapter, átjárók és útválasztók

- Az XBee Adapter, Gateway vagy Router termékek működése nem garantálható a rádiókapcsolat miatt, ezért nem használhatók biztonsági szempontból kritikus eszközök, például gépek vagy autóipari alkalmazások reteszelésére.

- Az XBee Adapter, Gateway vagy Router termékek nem engedélyezettek a következő országokban való használatra (ez a lista nem teljes):
  - orvosi eszközök
  - nukleáris alkalmazások
  - robbanásveszélyes vagy gyúlékony légkör
- Az XBee Adapter, Gateway vagy Router termékben nincsenek felhasználó által javítható alkatrészek. Ne távolítsa el a termék fedelét, és semmilyen módon ne módosítsa az átjárót vagy az útválasztót. A módosítások kizárhatják a termékre a jótállást, és az átjáró vagy az útválasztó az adott ország szabályozási megfelelőségén kívüli működését okozhatják, ami a termék esetleges illegális működéséhez vezethet.
- Az XBee Adapter, Gateway vagy Router termékek kezelésekor használjon ipari szabvány ESD védelmet.
- A kezelés során ügyeljen arra, hogy elkerülje a PCB és az alkatrészek elektromos károsodását.
- Ne tegye ki az XBee Adapter-, Gateway- vagy Router-termékeket víznek vagy nedvességnek.
- Használja ezt a terméket az XBee Adapter, Gateway vagy Router termék használati útmutatójában meghatározott antennákkal.
- A végfelhasználót tájékoztatni kell arról, hogyan távolítsa el az XBee Adapter, Gateway vagy Router termék áramellátását, vagy hogyan helyezze el az antennákat az emberektől vagy állatoktól 20 cm-re.

Istruzioni di sicurezza

Adattatore, gateway e router XBee

- Non è possibile garantire il funzionamento dei prodotti XBee Adapter, Gateway o Router a causa del collegamento radio e pertanto non devono essere utilizzati per gli interblocchi in dispositivi critici per la sicurezza come macchine o applicazioni automobilistiche.
- I prodotti XBee Adapter, Gateway o Router non sono stati approvati per l'uso in (questo elenco non è esaustivo):
  - dispositivi medici
  - applicazioni nucleari
  - atmosfere esplosive o infiammabili
- Non ci sono componenti riparabili dall'utente all'interno del prodotto XBee Adapter, Gateway o Router. Non rimuovere i coperchi del prodotto né modificare in alcun modo il Gateway o il Router. Le modifiche possono escludere il prodotto da qualsiasi garanzia e possono causare il funzionamento del gateway o del router al di fuori della conformità normativa per un determinato paese, portando al possibile funzionamento illegale del prodotto.
- Utilizzare la protezione ESD standard del settore quando si maneggia l'adattatore XBee, il gateway o il prodotto router.
- Prestare attenzione durante la manipolazione per evitare danni elettrici al PCB e ai componenti.
- Non esporre i prodotti XBee Adapter, Gateway o Router ad acqua o umidità.
- Utilizzare questo prodotto con le antenne specificate nelle guide per l'utente del prodotto XBee Adapter, Gateway o Router.

- L'utente finale deve essere informato su come rimuovere l'alimentazione dal prodotto XBee Adapter, Gateway o Router o come posizionare le antenne a 20 cm da esseri umani o animali.

Drošības instrukcijas

XBee adapteris, vārtejas un maršrutētāji

- XBee adaptera, vārtejas vai maršrutētāja izstrādājumu darbība nevar tikt garantēta radio saites dēļ, tāpēc tos nevajadzētu izmantot bloķēšanai tādās drošībai svarīgās ierīcēs kā mašīnas vai automobiļu lietojumprogrammas.
- XBee adaptera, vārtejas vai maršrutētāja produkti nav apstiprināti lietošanai (šis saraksts nav pilnīgs):
  - medicīniskās ierīces
  - kodolprogrammas
  - sprādzienbīstamā vai uzliesmojošā vidē
- XBee adaptera, vārtejas vai maršrutētāja izstrādājumā nav neviena komponenta, ko lietotājs varētu apkalpot. Nenoņemiet izstrādājuma vākus un nekādā veidā nepārveidojiet vārteju vai maršrutētāju. Izmaiņas var izslēgt uz izstrādājumu no jebkādas garantijas un var izraisīt vārtejas vai maršrutētāja darbību, kas neatbilst noteiktās valsts normatīvajiem aktiem, izraisot iespējamu produkta nelikumīgu darbību.
- Strādājot ar XBee adaptera, vārtejas vai maršrutētāja izstrādājumu, izmantojiet nozares standarta ESD aizsardzību.
- Rīkojoties, rīkojieties uzmanīgi, lai izvairītos no PCB un komponentu elektriskiem bojājumiem.
- Nepakļaujiet XBee adaptera, vārtejas vai maršrutētāja izstrādājumus ūdens vai mitruma iedarbībai.
- Izmantojiet šo izstrādājumu ar antenām, kas norādītas XBee adaptera, vārtejas vai maršrutētāja izstrādājuma lietotāja rokasgrāmatās.
- Galalietotājam ir jāinformē, kā atvienot strāvu no XBee adaptera, vārtejas vai maršrutētāja izstrādājuma vai atrast antenas 20 cm attālumā no cilvēkiem vai dzīvniekiem.

Saugos instrukcijos

XBee adapteris, šliuzai ir maršrutizatoriai

- „XBee“ adapterio, šliuzo ar maršrutizatoriaus gaminių veikimas negali būti garantuotas dėl radijo ryšio, todėl jie neturėtų būti naudojami blokuojant saugai svarbius įrenginius, pvz., mašinas ar automobilius.
- XBee adapterio, šliuzo ar maršrutizatoriaus produktai nebuvo patvirtinti naudoti (šis sąrašas nėra baigtinis):
  - medicinos prietaisai
  - branduolinės programos
  - sprogioje ar degioje aplinkoje
- XBee Adapter, Gateway arba Router gaminyje nėra komponentų, kuriuos vartotojas galėtų taisyti. Jokiu būdu nenuimkite gaminio dangtelių ir nekeiskite šliuzo ar maršrutizatoriaus. Dėl modifikacijų gaminiui gali būti netaikoma jokia garantija, o šliuzas arba maršruto parinktuvas

gali veikti nesilaikant tam tikros šalies teisės aktų reikalavimų, o tai gali sukelti neteisėtą gaminio veikimą.

- Dirbdami su XBee adapteriu, šliuzu ar maršrutizatoriumi naudokite pramonės standartinę ESD apsaugą.
- Dirbdami būkite atsargūs, kad nepažeistumėte PCB ir komponentų.
- Saugokite XBee adapterį, šliuzą ar maršrutizatorių nuo vandens ar drėgmės.
- Naudokite šį gaminį su antenomis, nurodytomis XBee adapterio, šliuzo ar maršrutizatoriaus gaminio vartotojo vadove.
- Galutiniam vartotojui turi būti paaiškinta, kaip atjungti maitinimą iš XBee adapterio, šliuzo ar maršrutizatoriaus gaminio arba nustatyti antenas 20 cm atstumu nuo žmonių ar gyvūnų.

Sikkerhetsinstruksjoner

XBee-adapter, gatewayer og rutere

- XBee Adapter-, Gateway- eller Router-produktene kan ikke garanteres drift på grunn av radiolinken og bør derfor ikke brukes til forriglinger i sikkerhetskritiske enheter som maskiner eller bilapplikasjoner.
- XBee Adapter, Gateway eller Router-produktene er ikke godkjent for bruk i (denne listen er ikke uttømmende):
  - medisinsk utstyr
  - kjernefysiske applikasjoner
  - eksplosive eller brennbare atmosfærer
- Det er ingen komponenter som kan repareres av brukeren inne i XBee Adapter-, Gateway- eller Router-produktet. Ikke fjern produktdekslene eller modifiser gatewayen eller ruteren på noen måte. Endringer kan ekskludere produktet fra enhver garanti og kan føre til at gatewayen eller ruteren fungerer utenfor regelverket for et gitt land, noe som kan føre til ulovlig drift av produktet.
- Bruk industristandard ESD-beskyttelse når du håndterer XBee Adapter-, Gateway- eller Router-produktet.
- Vær forsiktig ved håndtering for å unngå elektrisk skade på PCB og komponenter.
- Ikke utsett XBee Adapter-, Gateway- eller Router-produktene for vann eller fuktighet.
- Bruk dette produktet med antennene som er spesifisert i brukerveiledningene for XBee Adapter, Gateway eller Router.
- Sluttbrukeren må bli fortalt hvordan man fjerner strømmen fra XBee Adapter-, Gateway- eller Router-produktet eller plasserer antennene 20 cm fra mennesker eller dyr.

Instrukcje bezpieczeństwa

Adapter, bramy i routery XBee

- Produkty XBee Adapter, Gateway lub Router nie mogą zagwarantować działania ze względu na łącze radiowe, dlatego nie należy ich używać do blokad w urządzeniach o krytycznym znaczeniu dla bezpieczeństwa, takich jak maszyny lub aplikacje motoryzacyjne.

- Produkty XBee Adapter, Gateway lub Router nie zostały zatwierdzone do użytku w (lista ta nie jest wyczerpująca):
  - wyroby medyczne
  - zastosowania nuklearne
  - atmosferach wybuchowych lub łatwopalnych
- Wewnątrz adaptera, bramy lub routera XBee nie ma żadnych elementów, które mogłyby być serwisowane przez użytkownika. Nie zdejmuj osłon produktu ani nie modyfikuj w żaden sposób bramki lub routera. Modyfikacje mogą wyłączyć produkt z jakiejkolwiek gwarancji i spowodować, że brama lub router będzie działać niezgodnie z przepisami w danym kraju, co może prowadzić do nielegalnego działania produktu.
- Używaj standardowej ochrony ESD podczas obsługi produktów XBee Adapter, Gateway lub Router.
- Podczas obsługi należy zachować ostrożność, aby uniknąć uszkodzeń elektrycznych PCB i komponentów.
- Nie wystawiaj adaptera, bramki lub routera XBee na działanie wody lub wilgoci.
- Używaj tego produktu z antenami określonymi w instrukcji obsługi adaptera, bramki lub routera XBee.
- Użytkownik końcowy musi zostać poinformowany, jak odłączyć zasilanie od adaptera, bramki lub routera XBee lub jak zlokalizować anteny w odległości 20 cm od ludzi lub zwierząt.

Instruções de segurança

Adaptador, gateways e roteadores XBee

- Os produtos XBee Adapter, Gateway ou Router não podem ter operação garantida devido ao link de rádio e, portanto, não devem ser usados para intertravamentos em dispositivos críticos de segurança, como máquinas ou aplicações automotivas.
- Os produtos XBee Adapter, Gateway ou Router não foram aprovados para uso em (esta lista não é exaustiva):
  - dispositivos médicos
  - aplicações nucleares
  - atmosferas explosivas ou inflamáveis
- Não há componentes que possam ser reparados pelo usuário dentro do produto XBee Adapter, Gateway ou Router. Não remova as tampas do produto nem modifique o Gateway ou o Roteador de forma alguma. As modificações podem excluir o produto de qualquer garantia e fazer com que o gateway ou roteador opere fora da conformidade regulatória de um determinado país, levando à possível operação ilegal do produto.
- Use a proteção ESD padrão do setor ao manusear o produto Adaptador, Gateway ou Roteador XBee.
- Tome cuidado ao manusear para evitar danos elétricos à PCB e aos componentes.
- Não exponha os produtos XBee Adapter, Gateway ou Router à água ou umidade.
- Use este produto com as antenas especificadas nos guias do usuário do produto Adaptador, Gateway ou Roteador XBee.

- O usuário final deve ser informado sobre como remover a energia do produto Adaptador, Gateway ou Roteador XBee ou localizar as antenas a 20 cm de humanos ou animais.

Instructiuni de siguranta

Adaptor XBee, gateway-uri şi routere

- Nu se poate garanta funcţionarea produselor XBee Adapter, Gateway sau Router din cauza conexiunii radio şi, prin urmare, nu trebuie utilizate pentru interblocări în dispozitive critice pentru siguranţă, cum ar fi maşinile sau aplicaţiile auto.
- Produsele XBee Adapter, Gateway sau Router nu au fost aprobate pentru utilizare în (aceasta listă nu este exhaustivă):
  - dispozitive medicale
  - aplicaţii nucleare
  - atmosfere explozive sau inflamabile
- Nu există componente care să poată fi reparate de utilizator în interiorul produsului XBee Adapter, Gateway sau Router. Nu îndepărtaţi capacele produsului şi nu modificaţi Gateway-ul sau Routerul în niciun fel. Modificările pot exclude produsul din orice garanţie şi pot face ca gateway-ul sau routerul să funcţioneze în afara conformităţii cu reglementările pentru o anumită ţară, ceea ce duce la o posibilă funcţionare ilegală a produsului.
- Folosiţi protecţia ESD standard în industrie când manipulaţi produsul XBee Adapter, Gateway sau Router.
- Aveţi grijă în timpul manipulării pentru a evita deteriorarea electrică a PCB-ului şi a componentelor.
- Nu expuneţi produsele XBee Adapter, Gateway sau Router la apă sau umiditate.
- Utilizaţi acest produs cu antenele specificate în ghidurile de utilizare ale produsului XBee Adapter, Gateway sau Router.
- Utilizatorului final trebuie să i se spună cum să scoată alimentarea de la adaptorul, gateway-ul sau routerul XBee sau să găsească antenele la 20 cm de oameni sau anima

Bezpečnostné inštrukcie

XBee adaptér, brány a smerovače

- Produkty XBee Adapter, Gateway alebo Router nemožno zaručiť kvôli rádiovému spojeniu, a preto by sa nemali používať na blokovanie v zariadeniach kritických z hľadiska bezpečnosti, ako sú stroje alebo automobilové aplikácie.
- Produkty XBee Adapter, Gateway alebo Router neboli schválené na použitie v (tento zoznam nie je úplný):
  - zdravotnícke pomôcky
  - jadrové aplikácie
  - výbušné alebo horľavé atmosféry
- Vo vnútri produktu XBee Adapter, Gateway alebo Router sa nenachádzajú žiadne užívateľsky opraviteľné komponenty. Neodstraňujte kryty produktu ani žiadnym spôsobom neupravujte bránu ani smerovač. Úpravy môžu vyňať produkt zo záruky a môžu spôsobiť, že brána alebo

smerovač bude fungovať mimo zákonných predpisov pre danú krajinu, čo môže viesť k možnej nezákonnej prevádzke produktu.

- Pri manipulácii s produktom XBee Adapter, Gateway alebo Router používajte štandardnú ochranu ESD.
- Pri manipulácii buďte opatrní, aby ste predišli elektrickému poškodeniu dosky plošných spojov a komponentov.
- Produkty XBee Adapter, Gateway alebo Router nevystavujte vode ani vlhkosti.
- Tento produkt používajte s anténami špecifikovanými v používateľských príručkách produktu XBee Adapter, Gateway alebo Router.
- Koncový používateľ musí byť informovaný o tom, ako odpojiť napájanie XBee adaptéra, brány alebo smerovača alebo ako umiestniť antény 20 cm od ľudí alebo zvierat.

Varnostna navodila

Adapter, prehodi in usmerjevalniki XBee

- Izdelkom XBee Adapter, Gateway ali Router ni mogoče zagotoviti delovanja zaradi radijske povezave in se zato ne smejo uporabljati za zaklepanje v varnostno kritičnih napravah, kot so stroji ali avtomobilske aplikacije.
- Izdelki XBee Adapter, Gateway ali Router niso bili odobreni za uporabo v (ta seznam ni izčrpen):
  - medicinskih pripomočkov
  - jedrske aplikacije
  - eksplozivne ali vnetljive atmosfere
- V izdelku XBee Adapter, Gateway ali Usmerjevalnik ni komponent, ki bi jih lahko popravil uporabnik. Ne odstranjujte pokrovov izdelka in na noben način ne spreminjajte prehoda ali usmerjevalnika. Spremembe lahko izključijo izdelek iz kakršne koli garancije in lahko povzročijo, da prehod ali usmerjevalnik deluje zunaj zakonske skladnosti za dano državo, kar vodi do možnega nezakonitega delovanja izdelka.
- Pri ravnanju z izdelkom XBee Adapter, Gateway ali Usmerjevalnik uporabljajte industrijsko standardno zaščito pred ESD.
- Pri rokovanju pazite, da se izognete električnim poškodbam tiskanega vezja in komponent.
- Izdelkov XBee Adapter, Gateway ali Usmerjevalnik ne izpostavljajte vodi ali vlagi.
- Ta izdelek uporabljajte z antenami, navedenimi v uporabniških priročnikih izdelka XBee Adapter, Gateway ali Router.
- Končnemu uporabniku je treba povedati, kako odstraniti napajanje iz izdelka XBee Adapter, Gateway ali Usmerjevalnik ali naj locira antene 20 cm od ljudi ali živali.

Adaptador XBee, puertas de enlace y enrutadores

- No se puede garantizar el funcionamiento de los productos del adaptador, puerta de enlace o enrutador XBee debido al enlace de radio y, por lo tanto, no deben usarse para enclavamientos en dispositivos críticos para la seguridad, como máquinas o aplicaciones automotrices.

- Los productos XBee Adapter, Gateway o Router no han sido aprobados para su uso en (esta lista no es exhaustiva):
  - dispositivos médicos
  - aplicaciones nucleares
  - atmósferas explosivas o inflamables
- No hay componentes reparables por el usuario dentro del adaptador XBee, la puerta de enlace o el enrutador. No retire las cubiertas del producto ni modifique la puerta de enlace o el enrutador de ninguna manera. Las modificaciones pueden excluir el producto de cualquier garantía y pueden hacer que la puerta de enlace o el enrutador funcionen fuera del cumplimiento normativo de un país determinado, lo que puede conducir a la operación ilegal del producto.
- Utilice la protección ESD estándar de la industria cuando manipule el adaptador, la puerta de enlace o el enrutador XBee.
- Tenga cuidado al manipularlo para evitar daños eléctricos en la PCB y los componentes.
- No exponga los productos XBee Adapter, Gateway o Router al agua o la humedad.
- Utilice este producto con las antenas especificadas en las guías de usuario del producto XBee Adapter, Gateway o Router.
- Se debe indicar al usuario final cómo desconectar la alimentación del adaptador, la puerta de enlace o el enrutador XBee o cómo ubicar las antenas a 20 cm de personas o animales.

Säkerhets instruktioner

XBee-adapter, gateways och routrar

- XBee Adapter, Gateway eller Router-produkter kan inte garanteras funktion på grund av radiolänken och bör därför inte användas för förreglingar i säkerhetskritiska enheter som maskiner eller biltillämpningar.
- XBee Adapter, Gateway eller Router-produkter har inte godkänts för användning i (denna lista är inte uttömmande):
  - medicinsk utrustning
  - kärnkraftstillämpningar
  - explosiv eller brandfarlig atmosfär
- Det finns inga komponenter som användaren kan reparera inuti XBee Adapter-, Gateway- eller Routerprodukten. Ta inte bort produkthöljena eller modifiera gatewayen eller routern på något sätt. Ändringar kan utesluta produkten från alla garantier och kan göra att gatewayen eller routern fungerar utanför regelverket för ett visst land, vilket kan leda till olaglig användning av produkten.
- Använd industristandard ESD-skydd när du hanterar XBee Adapter, Gateway eller Router-produkten.
- Var försiktig vid hanteringen för att undvika elektriska skador på kretskortet och komponenterna.
- Utsätt inte XBee Adapter, Gateway eller Router-produkter för vatten eller fukt.
- Använd den här produkten med de antenner som specificeras i användarhandböckerna för XBee Adapter, Gateway eller Router.

- Slutanvändaren måste informeras om hur man kopplar bort strömmen från XBee Adapter, Gateway eller Router-produkten eller hur man placerar antennerna 20 cm från människor eller djur.

Digi IX15 Quick start

Migrating from a WR-series device? Click here for information and tools to set up your new IX-series router.

The following steps guide you through the setup of your DigiIX15 device.

Step 1: What's in the box

When you open the IX15 package, look for the following:

- Digi IX15 device



  The IX15 has a product label on the bottom of the device. The label includes product identification information and the default password assigned to the device. The IX15 also includes a terminal connector for the power supply installed in the power input.

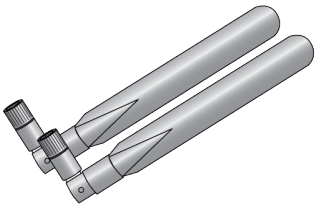
- Insert cards
- Digi IX15 label

  There is a printed copy of the product label on the bottom of your device. You can affix this label to the top or side of the device such that you can access the label after the device is mounted, or store the label in a safe place for future reference.

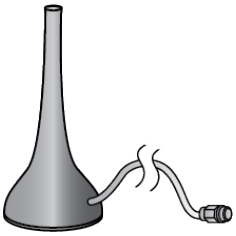
Step 2: Gather accessories

You may purchase accessories with the IX15 device, or you can provide your own.

Here is the list of accessories used in this *Quick start*:



Cellular antennas
Two cellular antennas.
Included in IX15 Accessory kit (76002107)—the kit may be ordered separately.



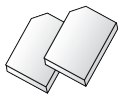
RF antenna
One RF antenna for XBee.
Included in IX15 Accessory kit (76002107)—the kit may be ordered separately.



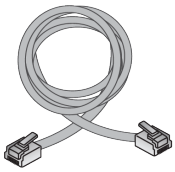
Power supply
9-30 V
Included in IX15 Accessory kit (76002107)—the kit may be ordered separately.



Laptop or personal computer
Use an Ethernet cable to connect your IX15 to a laptop or PC.



SIM card(s)
If you intend to configure cellular WWAN access at this time, acquire SIM cards as needed. Note the carrier, network Access Point Name (APN), and SIM pin—if any—for each card.



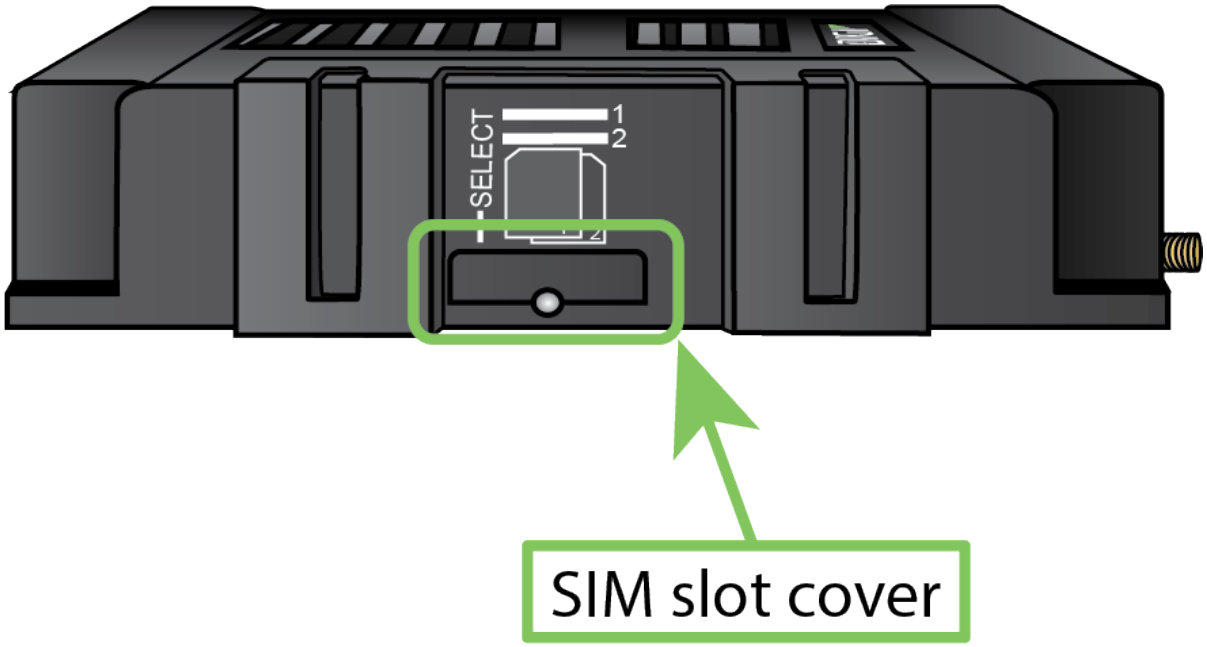
Ethernet cable.
One—Included in IX15 Accessory kit (76002107)—the kit may be ordered separately.

Step 3: Connect

1. Insert SIM card(s)

   Insert your activated SIM (2FF) card(s) provided by your cellular carrier into the device:

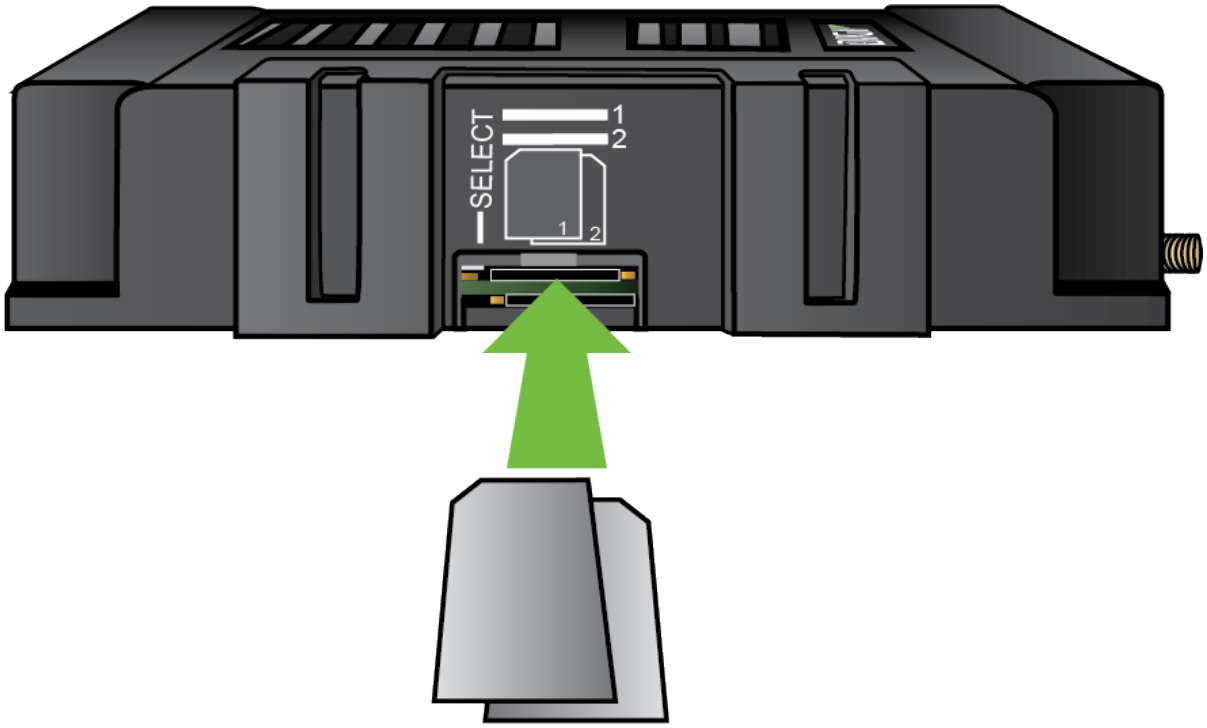
i. Use a small Phillips screw driver to remove the SIM slot cover.



ii. For high-vibration environments, apply a thin layer of dielectric grease to the SIM contacts.

Note If the IX15 device is used in an environment with high vibration levels, SIM card contact fretting may cause unexpected SIM card failures. To protect the SIM cards, Digi strongly recommends that you apply a thin layer of dielectric grease to the SIM contacts prior to installing the SIM cards.

a. Insert the SIM card(s) into the SIM sockets. Insert the end of each SIM card with the chamfered corner positioned as indicated. Push the SIM in until it clicks into place.



b. After SIM cards are installed, replace the SIM slot cover.

2. Attach antennas

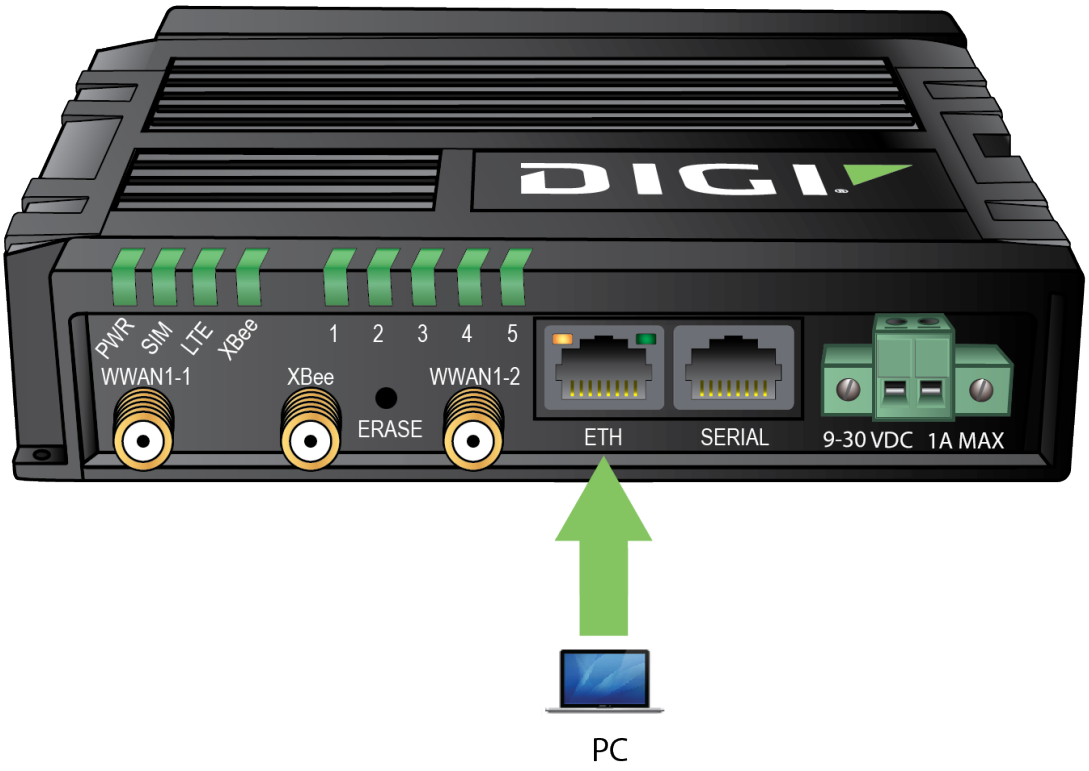
   Connect IX15-compatible cellular antennas to the WWAN1-1 and WWAN1-2 antenna connectors on the back of the device.

   Connect the RF antenna to the XBee antenna connector.

   Position the antennas for the best reception.



3. Use an Ethernet cable to connect the IX15 ETH port to your PC.



4. Connect DC power and power on the IX15 device. The IX15 is intended to be powered by a certified power supply with output rated at either 12 VDC/0.75 A or 24 VDC/0.375 A minimum. See IX15 power supply requirements.



5. Wait until a cellular connection has been established.

- The Power LED is solid green as the device prepares to boot up.
- When the Power LED turns to solid turquoise, the IX15 is ready.

When connecting for the first time, it could take several minutes for the IX15 device to connect to the cellular network while it attempts to determine the APN required for the connection.

- Indicator LEDs blink to show status during startup.
- Verify that the LTE LED on the front of the IX15 shows either green or blue—solid or flashing—for proper operation.
- Verify that the signal strength indicator on the front of the IX15 shows two or more bars.

CAUTION! If your laptop or PC is connected to the ETH port on the IX15 and the IX15 has a cellular internet connection established, the laptop or PC will likely automatically switch from its Wi-Fi internet connection to its Ethernet port. This will result in the IX15 routing all internet data to and from the laptop or PC. This could result in excessive data charges to the cellular data account associated with the SIM card in the IX15. To interface with the IX15 via LAN while maintaining internet through the laptop or PC Wi-Fi, follow the standard procedure for changing the priority order of network adapters in the laptop or PC.
You can also change the configuration of the Ethernet port to WAN, connect the IX15 to the internet by providing an internet connection to the WAN-configured Ethernet port, and configure the device over Digi Remote Manager. For instructions on configuring the ETH port for WAN, see Configure a Wide Area Network (WAN).

Note If your SIM card has an APN that is not recognized by the IX15 device, skip this step and configure the APN following the procedure in Configure cellular modem APNs.

Step 4: Configure

This section describes how to configure the device by using the local Web UI. You can also use Digi Remote Manager to configure the device, including using a Digi Remote Manager device configuration to automatically update the device. See the *Digi Remote Manager User Guide*.

1. On the PC connected to the IX15 , open a browser and go to **192.168.2.1**.

Note The device is also accessible at the default IP address of **192.168.210.1**. However, because this IP address does not use a DHCP server, to connect to this address you must configure your local PC with an appropriate static IP address—for example, **192.168.210.2**.

2. Log into the IX15:

   User name: Use the default user name: **admin**.

   Password: Use the unique password printed on the bottom label of the device—or the printed label included in the package.

3. When you first log into the WebUI or the command line, you must change the password for the **admin** user. See Change the default password for the admin user for the admin user for instructions.

Step 5: Next steps

Once you have setup your IX15, it is time to start working with your device and discover all its features following the *Get started guide*.

Get started

This section guides you through your first steps with the Digi IX15 Gateway.

You will connect your hardware, program the IX15 XBee with a profile, and create an XBee network.

Step 1: Requirements 51
Step 2: Setup the hardware 51
Step 3: Program an XBee profile 51
Step 4: Join nodes to the IX15 network 53
Step 5: Review your XBee network 54
Next steps 54

Step 1: Requirements

To start working with your IX15 and create an XBee network, you need:

1. The accessories listed in Gather accessories.
2. An XBee3 module and its corresponding XBIB-C Development Board and USB-C cable.

Note We recommend using Digi XBIB-C Development Boards to follow this guide, but any other Digi XBee development board is also valid—such as Digi XBIB-U Development Boards or the Digi XBee Grove Development Board.

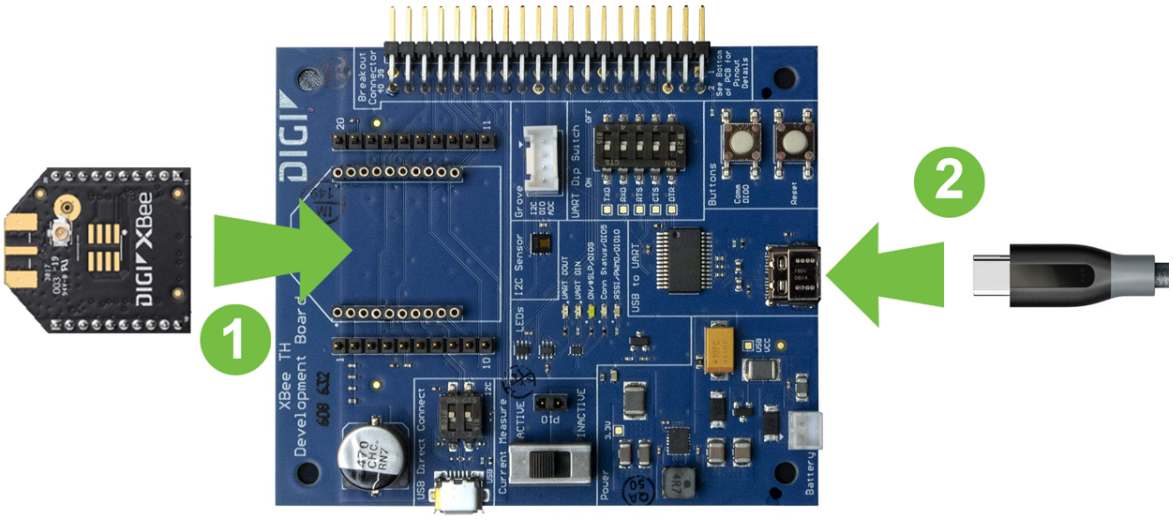
---

3. XCTU application to program XBee profiles. You can download and install from here.

Step 2: Setup the hardware

To connect the hardware:

1. Follow the instructions in Digi IX15 Quick start to connect your IX15.
2. Set up the XBee device for the XBee network:



3. Plug the XBee device into the XBIB-C board. Pay attention to the XBee footprint drawn on the board so you place it with the correct orientation.
4. Connect the USB-C cable to the XBIB-C board in one end and to your local PC on the other end. These devices will be discovered and configured later using XCTU.

Step 3: Program an XBee profile

The IX15 comes with a set of default configuration settings, firmware version, and XBee protocol based on a Zigbee router profile. To update these configuration, you have to program an XBee profile.

The IX15 includes three default profiles—one per protocol—that you can use:
Zigbee: xbee_gateway_zigbee

Parameter	Value
CE	1
ID	1234
NI	GATEWAY

DigiMesh: xbee_gateway_digimesh

Parameter	Value
CH	C
ID	1234
CE	0
NI	GATEWAY

802.15.4: xbee_gateway_802.15.4

Parameter	Value
CH	C
ID	1234
NI	GATEWAY
MY	FFFE

Follow these steps to program one of them:

1. Open the IX15 WebUI and login.
2. On the top menu, click **XBee**.
3. Under **XBee Administration**, click **Update Manager**.
4. On the **XBee Update Manager** page, click **Add Update Tasks**. The **Add new update** tasks window displays.
5. Select your profile from the **Available XBee Profiles** list and click **Next**.
6. Check the XBee device that appears in the list and click **Add** to apply the XBee profile.

Note If more than one XBee device appears in the list, the first one corresponds to the IX15. Select that one.

7. The new task appears in the **Current XBee Update Tasks** panel.
8. Click **Start Update** to begin running the update tasks list.
9. The window refreshes showing the update progress.

Step 4: Join nodes to the IX15 network

Now that you have the IX15 configured, you can add new XBee devices to the network. To do so, you have to configure the XBee devices that you connected to your computer in Setup the hardware.

1. Download the XBee 3 get started profiles from this link and unzip it. The package contains profiles for each XBee 3 protocol and hardware variants, through-hole or SMT:
   a. Zigbee
      gs_sensor_zigbee-th.xpro
      gs_sensor_zigbee-smt.xpro
   b. DigiMesh
      gs_sensor_digimesh-th.xpro
      gs_sensor_digimesh-smt.xpro
   c. 802.15.4
      gs_sensor_802.15.4-th.xpro
      gs_sensor_802.15.4-smt.xpro

   These profiles are ready to work with the IX15 programmed with the corresponding protocol profile—xbee_gateway_zigbee, xbee_gateway_digimesh, xbee_gateway_802.15.4—that you already flashed in Program an XBee profile.

   They also include a MicroPython application that reports data to the Digi IX15 Gateway

2. Program the XBee3 module with the corresponding profile—by protocol and hardware variant—using XCTU.

a. Open XCTU.
b. Click **Discover radio modules** on the XCTU toolbar. The **Discover radio modules** dialog box appears.
c. Select the serial ports you would like to scan for the XBee 3 modules and click **Next**.
d. Select any port parameters you would like to include in the search process—you can leave the default settings—and click **Finish** to initiate the discovery scan. A new dialog opens, displaying devices found and the estimated time remaining.
e. When the discover process finishes, check the XBee 3 modules you want to configure and click **Add selected devices**. The modules appear in the device list.
f. Select a radio module from the list and wait until all its settings are read.
g. Click the **Configuration profiles** drop-down menu on the **Configuration** toolbar and select **Apply configuration profile**. An **Open file** dialog appears, asking for the configuration profile file to load.
h. Locate the **gs_sensor_<protocol>-<hw_variant>.xpro** from the software artifacts packet just downloaded and click **Open**.
i. Wait until XCTU applies the profile to the XBee3 module.
j. Close XCTU.

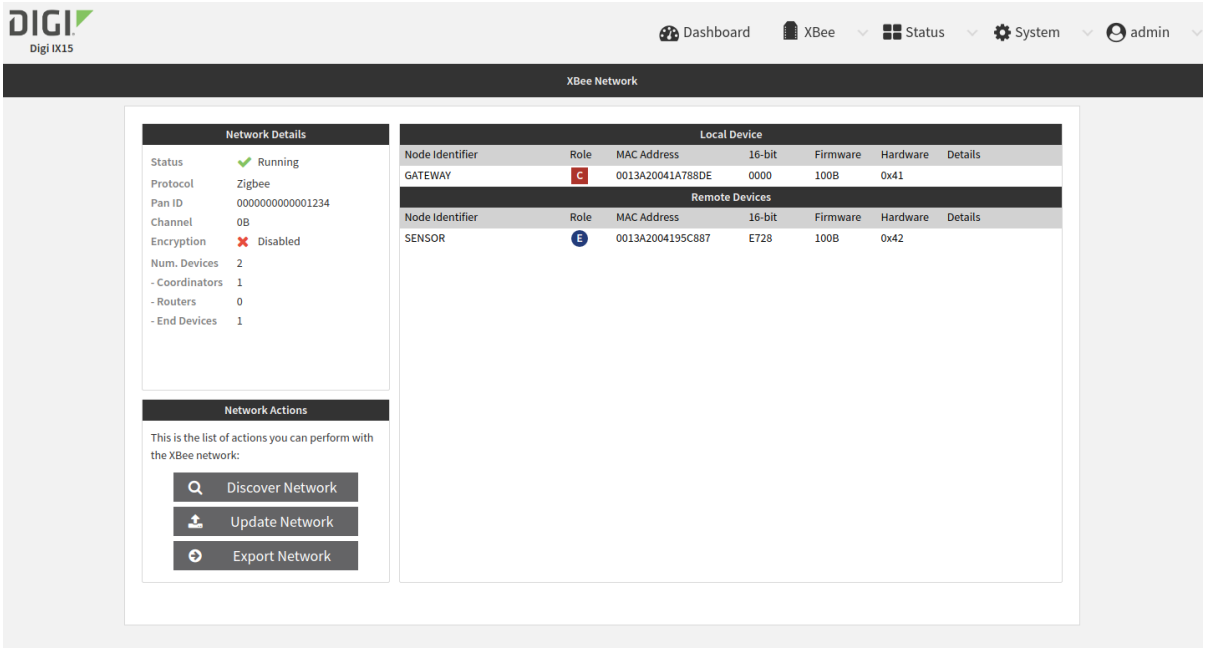
Note See Configure your modules in the *XCTU User Guide* for more information on configuring XBee devices.

3. Push the reset button on the development board. The programmed MicroPython application starts to report data to the Digi IX15 Gateway.

Step 5: Review your XBee network

Once the XBee devices are correctly configured, you can check if they are now part of your XBee network. To do so:

1. Open the IX15 WebUI and login.
2. On the menu, click **XBee**.
3. Under **XBee Administration**, click **Network Manager**.
4. The panel on the right displays all of the network nodes:



If you cannot see the XBee nodes yet, perform a network discovery:

1. Click **Discover Network** on the left.
2. In the popup dialog, check the **Clear devices** list before the discovering option.
3. The discovery process starts.
4. Discovered devices appear in the panel on the right.

Next steps

What else can you do with your Digi IX15 Gateway?

- Python application development: Create a Python application using the Digi XBee PyCharm Plugin, build, and learn how to launch it in your IX15.
- XBee network management: Discover your XBee network, get information from nodes, update their firmware, and configure their settings.

- System configuration: Configure your network interfaces, serial ports, firewall, VPNs, and other services:
  - Configuration and management
  - Bluetooth Low Energy
  - Power management
  - Interfaces
  - Serial port
  - Routing
  - Virtual Private Networks (VPN)
  - Services
  - User authentication
  - Firewall
- Remote device management: Remotely monitor and analyze multiple devices, manage their configuration, or update the entire system via the integrated Remote Manager support. You can also use Amazon AWS IoT and Microsoft Azure.

Digi IX15 hardware reference

This chapter contains the following topics:

Digi IX15 features and specifications 57
IX15 accessories 57
IX15 front and side views 57
IX15 LEDs 59
IX15 power supply requirements 62
Power consumption 63
Digi IX15 serial connector pinout 65
Antenna specifications for the cellular modem 65
Antenna specifications for the XBee RF Module 66
QR code definition 66

Digi IX15 features and specifications

The Digi IX15 key features include:

- Industrial grade components.
- Operating temperatures:

  -40°C to +74°C (-40°F to +165°F)
- LTE CAT 4 modem (Quectel EG25-G) with two SIM slots
- 10/100 BaseT Ethernet port for high-speed connectivity.
- Supports the following cellular bands:

Network standard	Supported bands
LTE Cat 4	B1, B2, B3, B4, B5, B7, B8, B12, B13, B18, B19, B20, B25, B26, B28, B38, B39, B40, B41
3G	B1, B2, B4, B5, B6, B8, B19
2G EDGE / GPRS	850 / 900 / 1800 / 1900 MHz

IX15 accessories

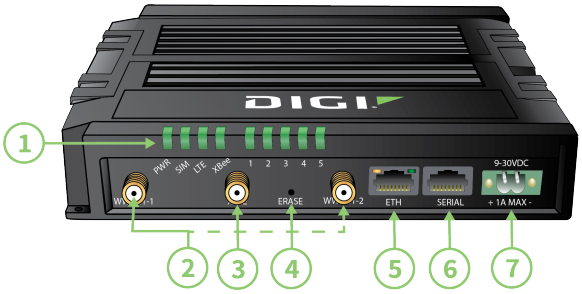
When accessories are purchased with the IX15 device, the following are provided:

- Cellular antennas.
- Power supply.
- Ethernet cable.
- XBee antenna.
- Serial cable.

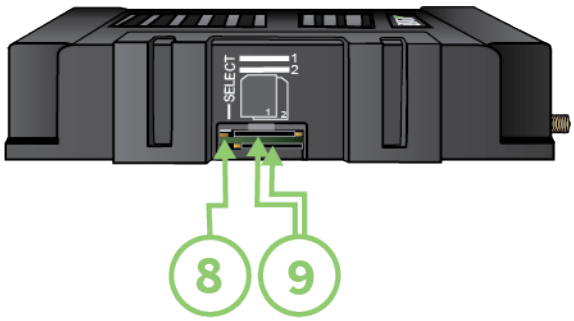
These accessories are included in the IX15 Accessory kit 76002107.

IX15 front and side views

The following figure shows the front view of the IX15.



Item	Description
1. **LEDs**	See IX15 LEDs.
2. **WWAN Antenna connectors**	Main (WWAN1-1) and auxillary (WWAN1-2) cellular antenna connectors.
3. **XBee Antenna connector**	2.4 GHz mag base antenna connector.
4. **ERASE** button	The **ERASE** button is used to perform a device reset, and it has three modes: 1. **Configuration reset**: Pressing the **ERASE** button one time will reset the device configurations to the factory default. It will not remove any automatically generated certificates and keys. 2. **Full device reset**: After the device reboots from the first button press, press the **ERASE** button again before the device is connected to the internet to also remove generated certificates/keys. 3. **Firmware reversion**: Press and hold the **ERASE** button and then power on the device to boot to the version of firmware that was used prior to the current version.
5.**Ethernet port**	LAN-enabled by default.
6. **Serial port**	See Digi IX15 serial connector pinout for information about the serial port pin-out.
7. **Power suppy**	IX15 power supply requirements.



Item	Description
8. **SELECT**	The **SELECT** button is used to manually toggle between the two SIM slots.
9. **SIM slots**	See Install SIM cards for more details.

IX15 LEDs

The IX15 LEDs are located on the top front panel. During bootup, the front-panel LEDs light up in sequence to indicate boot progress.



Power (PWR)

	Off No power.
	Solid green DC power is connected to the device.
	Solid blue Device is ON and connected to the internet.
	Alternating green/blue Device is ON and either Ethernet or Cellular networks are connecting.

SIM

Indicates that a SIM is in use:

	Off No SIM is present
	Solid green SIM1 is active.
	Solid blue SIM 2 is active
	Solid red SIM failure.

LTE

Indicates that the status of the cellular module and the ETH Ethernet port connection:

	Solid yellow (or orange) Initializing or starting up.		
	Flashing yellow (or orange) In the process of connecting to the cellular network and to a device on its ETH port.		**Flashing white** ETH port connection established and in the process of connecting to the cellular network.
	Flashing green Connected to 2G or 3G and is in the process of connecting to any device on its ETH port, or nothing is connected to the port.		**Solid green** Connected to 2G or 3G and also has a device linked to its ETH port.
	Flashing blue Connected to 4G LTE and in the process of connecting to a device on its ETH port.		**Solid blue** Connected to the 4G LTE and also has a device link to its ETH port.
	Alternating Red/yellow (or orange) Upgrading firmware. **WARNING!** DO NOT POWER OFF DURING FIRMWARE UPGRADE.		

XBee

Indicates the XBee status:

	Off XBee interface is disabled.		**Flashing green** Network discovery requested by customer.
	Solid red XBee interface is enabled but the XBee service is not running.		**Alternating Red/yellow (or orange)** Updating local XBee. **WARNING!** DO NOT POWER OFF DURING UPDATE.
	Solid green XBee interface is enabled and XBee service is running.		**Flashing yellow (or orange)** Updating a remote XBee.
	Solid blue XBee active discovery running.		**Flashing red** Recovering local XBee.

Signal quality indicators

LEDs labeled **1** through **5** Indicate the cellular service quality level.

Signal bars	Weighted dBm	Signal strength %	Quality
▮	-113 to -99	0% to 23%	Bad
▮▮	-98 to -87	24% to 42%	Marginal
▮▮▮	-86 to -76	43% to 61%	OK
▮▮▮▮	-75 to -64	62% to 80%	Good
▮▮▮▮▮	-63 to -51	81% to 100%	Excellent

The weighted dBm measurements are negative numbers, meaning values closer to zero denote a larger number. For example, a -85 is a better signal than -90.

Note See Signal quality bars explained for more information regarding how signal strength is calculated and subsequently displayed via the LED indicators.

Ethernet Link and Activity

The LEDs on the **ETH** port indicate that the Ethernet network interface is up and there is activity on the network interface.



Left LED (on top of port connector)

- **Off**: No Ethernet link detected.
- **Blinking amber**: Indicates Ethernet traffic.

Right LED (on top of port connector)

- **Off**: No Ethernet link detected.
- **Solid green**: 10/100 Mbps link detected.

Signal quality bars explained

The signal status bars for the Digi IX15 measure more than simply signal strength. The value reported by the signal bars is calculated using an algorithm that takes into consideration the Reference Signals Received Power (RSRP), the Signal-to-noise ratio (SNR), and the Received Signal Strength Indication (RSSI) to provide an accurate indicator of the quality of the signal that the device is receiving.

For 3G networks (including HSPA+) and 2G networks, the signal strength bars are determined by the RSSI value.

4G LTE algorithms

For 4G LTE, the IX15 device determines the RSRP, SNR, and RSSI values separately and uses the following algorithms to display the signal quality:

```
RSRP > -85, rsrp_bars=5
-95 < RSRP <= -85, rsrp_bars=4
-105 < RSRP <= -95, rsrp_bars=3
-115 < RSRP <= -105, rsrp_bars=2
-199 < RSRP <= -115, if we're connected to the cellular network, rsrp_bars=1,
if not rsrp_bars=0
```

If RSRP <= -199, the device uses the RSSI as the value with the same algorithm:

```
SNR >= 13, snr_bars=5
4.5 <= SNR < 13, snr_bars=4
1 <= SNR < 4, snr_bars=3
-3 < SNR < 1, snr_bars=2
-99 < SNR <= -3, if we're connected to the cellular network, snr_bars=1, if not
snr_bars=0
```

Once the **snr_bars** and **rsrp_bars** values are determined, the device uses the lesser of the two as the reported signal a bars.

3G algorithm

For 3G, the IX15 determines RSSI signal strength:

```
RSSI > -80, bars=5
-90 < RSSI <= -80, bars=4
-100 < RSSI <= -90, bars=3
-106 < RSSI <= -100, bars=2
RSSI <= -106, if we're connected to the cellular network, bars=1, if not bars=0
```

bars is then reported as the signal strength bars.

2G algorithm

For 2G, the IX15 determines RSSI signal strength:

```
RSSI > -80, bars=5
-89 < RSSI <= -80, bars=4
-98 < RSSI <= -89, bars=3
-104 < RSSI <= -98, bars=2
RSSI <= -104, if we're connected to the cellular network, bars=1, if not bars=0
```

bars is then reported as the signal strength bars.

IX15 power supply requirements

IX15 is intended to be powered by a certified power supply with output between 12VDC/0.75 Amp to 24 VDC/0.375 Amp at 9 Watts minimum. The voltage tolerance supports +/- 10% (9 VDC to 30 VDC).

- For installations operating in an ambient temperature range from -40 to +74°C, use the Digi power supply accessory kit 76002104.
- If you are providing the DC power source with a non-Digi power supply, use a Mean Well WDR-60-xx series certified LPS power supply or one that is rated at either 12VDC/0.75A or 24

VDC/0.375A minimum and is suitable for the ambient temperature for which it is installed.

- For installations requiring protective earth grounding, connect the –ve terminal of the power connector to the system protective earth with a minimum 1mm2 stranded single insulated cable. Crimp terminals should be used for all connections.

Power consumption

This section contains information about the power consumption of the Digi IX15 Gateway. All measurements were performed running Digi Accelerated Linux version 20.11.32.138.

All presented results were measured at ambient temperature (25ºC).

Note These power consumption numbers should be considered guidelines only, never as fixed or absolute values. Actual values will depend entirely upon individual setup and system application.

Power consumption use cases

The power consumption of the IX15 was evaluated in the following use cases:

Power off mode

Long-term power off state. In this mode, the power-on key can be asserted to start the device. Resumption is done via a full reboot.

You can enter this mode by issuing the following command:

```
# poweroff
```

Observations:

- If not shut down prior to the **poweroff** command, the modem remains on.
- If not shut down prior to the **poweroff** command, the XBee reamins on.
- Power LED remains ON.
- Ethernet LEDs remain ON.

Suspend to ram mode

Low power mode state that allows system resume without a full reboot. The logic state of the system is stored in RAM, which stays in self-refresh state. Resumption is done via a wake-up event, which causes the system to perform a warm boot.

You can enter this mode by issuing the following command:

```
# suspend
```

System idle

System stays in Idle mode running for a while after boot.

Stress CPU State

This state is achieved by executing this command and leaving the device running for a while:

```
cat /dev/urandom > /dev/null &
```

Results

State	XBee status	Modem status	Current consumption (mA)	Power consumption (mW)
Power off	ON	ON	48 - 52	890 - 930
	ON	OFF	34	620
	OFF	ON	42 - 48	780 - 890
	OFF	OFF	31.6	580
Suspend to ram	ON	ON	52	1000
	ON	OFF	14	250
	OFF	ON	48	870
	OFF	OFF	10	180
Idle	ON	ON	78 - 96	1420 - 1750
	ON	OFF	35 - 40	630 - 730
	OFF	ON	74 - 93	1350 - 1700
	OFF	OFF	33 - 37	600 - 670
Stress CPU	ON	ON	89 - 107	1620 - 1950
	ON	OFF	46 - 53	840 - 950
	OFF	ON	85 - 104	1550 - 1900
	OFF	OFF	43 - 50	780 - 910

Note Disconnecting the serial port loopback connector, all measures are reduced by 3-4 mA.

Digi IX15 serial connector pinout

The IX15 is a DTE device. The pinout for the 10 pin RJ-45 serial connector is as follows:

Pin number	RS232 signal	RS485 half-duplex signal
1	RI	
2	DSR	
3	RTS	
4	GND	GND
5	Tx	TxD/RxD+
6	Rx	TxD/RxD-
7	GND	SG
8	CTS	
9	DTR	
10	DCD	

10-pin serial cabling options

Digi offers several cabling options for connecting a 10 pin RJ-45/RJ-50 serial port to DB9 and DB25 serial connectors. Digi recommends the RJ45/Bare Wire 48 inch cable, part number 76000723, which provides a customizable connector to connect EIA 422/485 Devices to Digi MEI products that have 10 pin RJ45 connectors. The *PortServer TS, Digi Connect, and Digi One Products Cable Guide* also provides information about additional Digi cabling options.

Antenna specifications for the cellular modem

This product's certification was obtained by using the antenna described here. End users who select their own antennas should use one that matches these specifications to maintain the module's certification. You can use antennas of the same type but specify equivalent or lower gain.

Specification	Values		
Model	2JW0924-C952B		
Input Impedance (ohms)	50		
Polarization	Linear		
Max Input Power (W)	25		
Connector Type	SMA – Male (RP-SMA)		
Type	1/2 λ Dipole		
Band	700/850/900	1700/1800/1900/2100	2600

Specification	Values		
Frequency (MHz)	698-960	1710-2170	2500-2700
VSWR	1.7:1	1.5:1	1.3:1
Peak Gain (dBi)	-0.6	2.6	2.3

Antenna specifications for the XBee RF Module

This product's certification was obtained by using the antenna described here. End users who select their own antennas should use one that matches these specifications to maintain the module's certification. You can use antennas of the same type but specify equivalent or lower gain.

The XBee antenna must be positioned a minimum of 20 cm (7.9 in) from cellular antennas.

Specification	Value
Model number	DC-ANT-24DT
Frequency range	2.4 GHz ~ 2.5 GHz
Impedance	50 Ω nominal
VSWR	1.92 maximum
Return loss	-10 dB maximum
Electrical wave	1/2 λ Dipole
Gain	1.8 dBi
Admitted power	1 W

QR code definition

A QR code is printed on the label attached to the device and on the loose label included in the box with the device components. The QR code contains information about the device.



QR code items

Semicolon separated list of:

ProductName;DeviceID;Password;SerialNumber;SKUPartNumber-SKUPartRevision

Example

IX15;00000000-00000000-112233FF-FF445566;PW1234567890;50001001-00

Hardware setup

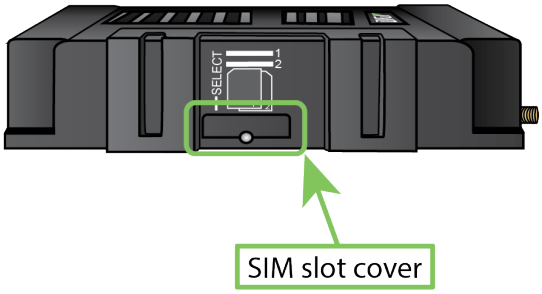
This chapter contains the following topics:

Install SIM cards 69
Connect data cables 70
Mount the IX15 device 70

Install SIM cards

Insert the activated SIM (2FF) card(s) provided by your cellular carrier into the device:

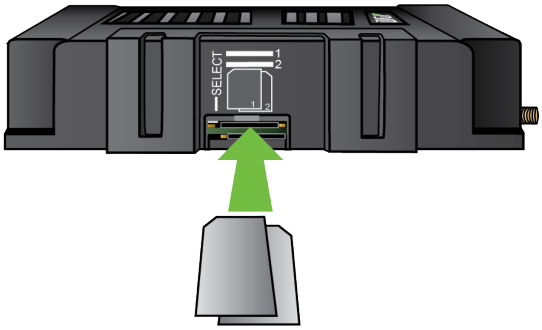
1. On the IX15 side panel, use a #1 Phillips-head screwdriver to remove the SIM door.



2. For high-vibration environments, apply a thin layer of dielectric grease to the SIM contacts.

Note If the Digi IX15 Gateway device is used in an environment with high vibration levels, SIM card contact fretting may cause unexpected SIM card failures. To protect the SIM cards, Digi strongly recommends that you apply a thin layer of dielectric grease to the SIM contacts prior to installing the SIM cards. See Apply Dielectric Grease over SIM Contacts for instructions.

3. Insert the SIM card(s) into the SIM sockets. Insert the end of each SIM card with the chamfered corner positioned as indicated. Push the SIM in until it clicks into place.



4. After all SIM cards are in place, use a #1 Phillips-head screwdriver to carefully replace the SIM door.

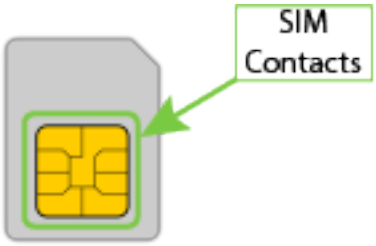


WARNING! Take care when you tighten the screws on the SIM door. If you apply too much pressure and over-tighten the screws, you can damage the SIM door or strip the screw threads. Torque to 2.9 inch/pounds.

Apply Dielectric Grease over SIM Contacts

Note Digi recommends using either …the Loctite® LB 8423 Dielectric Grease or Synco Lube® Silicone Dielectric Grease.

1. Use a sheet of paper or cardboard over the area where you intend to work.
2. Use isopropyl alcohol and a cotton-tipped applicator to gently clean the SIM contacts. Using isopropyl alcohol requires a well vented environment. Demineralized water can also be used as an alternative.



3. Once the surface is clean and dry, apply a small amount of dielectric grease in a thin layer over the contacts. Use a new cotton-tipped applicator to work the grease smoothy over the contacts. Apply gentle pressure.
4. When the dielectric grease has been applied, insert the SIM into the SIM slot as described above.

SIM removal

The IX15 has a PUSH-PUSH SIM connector. To insert, push each SIM in until it clicks, and repeat for removal. When you push to eject, the SIM ejects back out about 1/8 inch.

Tips for improving cellular signal strength

If the signal strength LEDs or the signal quality for your device indicate **Poor** or **No service**, try the following things to improve signal strength:

- Move the device to another location.
- Try connecting a different set of antennas, if available.
- Purchase a Digi Antenna Extender Kit: Antenna Extender Kit, 1m.

Connect data cables

The IX15 provides two types of data ports:

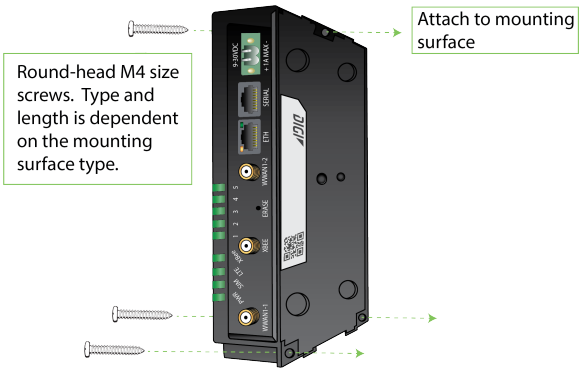
- **Ethernet** (RJ-45): Use a Cat 5e or Cat 6 Ethernet cable.
- **Serial** (RJ-45): Use a serial cable with an RJ45 connector to connect to the IX15 device. See 10-pin serial cabling options for information about Digi's 10-pin RJ-50 cables.

Mount the IX15 device

There are two options for mounting the IX15 device:

- Attach to a mounting surface by using the mounting tabs.
- Attach to DIN rail with clip.

Attach to a mounting surface by using the mounting tabs



Attach to DIN rail with clip

The DIN rail clip is an optional accessory included when the IX15 is purchased with accessories. The DIN rail clip kit part number is 76002095.

1. Attach the DIN rail clip to the back of the device:
   a. Attach the DIN rail clip to the back of the device with the screws provided.



b. Set the IX15 device onto a DIN rail and gently press until the clip snaps into the rail.



2. Attach the DIN rail clip to the bottom of the device:
   a. Attach the DIN rail clip to the bottom of the device with the screws provided.

WARNING! Using screws longer than 5.0 mm will cause damage to the IX15.





b. Set the IX15 device onto a DIN rail and gently press until the clip snaps into the rail.



WARNING! If being installed above head height on a wall or ceiling, ensure the device is fitted securely to avoid the risk of personal injury. Digi recommends that this device be installed by an accredited contractor.

Configuration and management

This chapter contains the following topics:

Review IX15 default settings 75
Change the default password for the admin user 76
Configuration methods 78
Using Digi Remote Manager 80
Access Digi Remote Manager 80
Using the web interface 80
Use the local REST API to configure the IX15 device 81
Using the command line 86
Access the command line interface 86
Log in to the command line interface 86
Exit the command line interface 87

Review IX15 default settings

You can review the default settings for your IX15 device by using the local WebUI or Digi Remote Manager:

Local WebUI

1. Log into the IX15 WebUI as a user with Admin access. See Using the web interface for details.
2. On the menu, click **System** > **Device Configuration**.

Digi Remote Manager

1. If you have not already done so, connect to your Digi Remote Manager account.
2. Click **Device Management** to display a list of your devices.
3. Locate and select your device as described in Use Digi Remote Manager to view and manage your device.
4. Click **Configure**.

The following tables list important factory default settings for the IX15.

Default interface configuration

Interface type	Preconfigured interfaces	Devices	Default configuration
Wireless Wide Area Network (WWAN)	▪ **Modem**	▪ **Modem**	▪ Firewall zone: External ▪ WAN priority: Metric=3 ▪ SIM failover after 5 attempts ▪ SureLink enabled for IPv4
Local Area Network (LAN)	▪ **ETH**	▪ Ethernet: **ETH**	▪ Firewall zone: Internal ▪ IP Address: 192.168.2.1/24 ▪ DHCP server enabled ▪ LAN priority: Metric=5
	▪ **Loopback**	▪ Ethernet: **Loopback**	▪ Firewall zone: Loopback ▪ IP address: 127.0.0.1/8

Interface type	Preconfigured interfaces	Devices	Default configuration
	■ **Default IP**	■ Ethernet: **ETH**	■ Firewall zone: Setup ■ IP address 192.168.210.1/24
	■ **Default Link-local IP**	■ Ethernet: **ETH**	■ Firewall zone: Setup ■ IP address 169.254.100.100/16

Other default configuration settings

Feature	Configuration
Central management	■ Digi Remote Manager enabled as the central management service.
Security policies	■ Packet filtering allows all outbound traffic. ■ SSH and web administration: • Enabled for local administration • Firewall zone: Internal
Monitoring	■ Device heath metrics uploaded to Digi Remote Manager at 60 minute interval. ■ SNMP: Disabled
Serial port	■ Enabled ■ Serial mode: Remote ■ Label: None ■ Baud rate: 9600 ■ Data bits: 8 ■ Parity: None ■ Stop bits: 1 ■ Flow control: None

Change the default password for the admin user

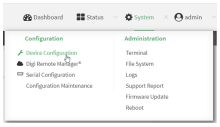
The unique, factory-assigned password for the default **admin** user account is printed on the bottom label of the device and on the loose label included in the package.

If you erase the device configuration or reset the device to factory defaults, the password for the **admin** user will revert to the original, factory-assigned default password.

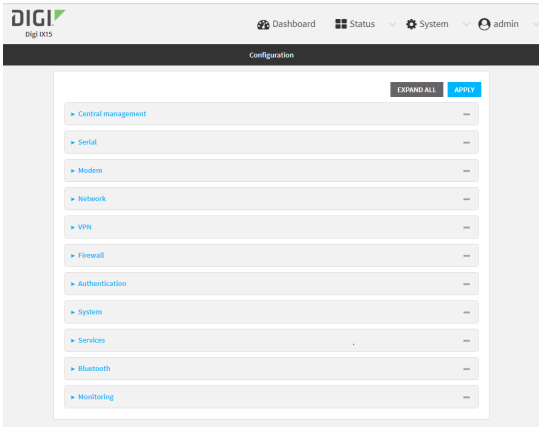
To change the default password for the admin user:

WebUI

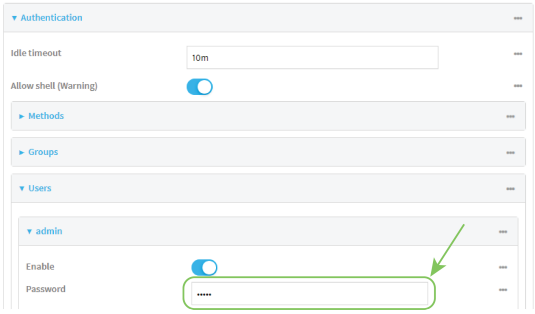
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Authentication** > **Users** > **admin**.
4. Enter a new password for the admin user. The password must be at least eight characters long and must contain at least one uppercase letter, one lowercase letter, one number, and one special character.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Set a new password for the admin user. The password must be at least eight characters long and must contain at least one uppercase letter, one lowercase letter, one number, and one special character.

```
(config)> auth user admin password new-password
(config)>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configuration methods

There are two primary methods for configuring your IX15 device:

- Web interface.

  The web interface can be accessed in two ways:

  - Central management using the Digi Remote Manager, a cloud-based device management and data enablement platform that allows you to connect any device to any application, anywhere. With the Remote Manager, you can configure your IX15 device and use the configuration as a basis for a profile which can be applied to other similar devices. See Using Digi Remote Manager for more information about using the Remote Manager to manage and configure your IX15 device.
  - The local web interface. See Using the web interface for more information about using the local web interface to manage and configure your IX15 device.

    Note Changes made to the device's configuration by using the local web interface will not be automatically reflected in Digi Remote Manager. You must manually refresh Remote Manager for the changes to be displayed.

  Web-based instructions in this guide are applicable to both the Remote Manager and the local web interface.

- Command line.

  A robust command line allows you to perform all configuration and management tasks from within a command shell. Both the Remote Manager and the local web interface also have the option to open a terminal emulator for executing commands on your IX15 device. See Using the command line for more information about using the command line to manage and configure your IX15 device.

In this guide, task topics show how to perform tasks:

WebUI

Shows how to perform a task by using the local web interface.

Command line

Shows how to perform a task by using the command line interface.

Using Digi Remote Manager

By default, your IX15 device is configured to use Digi Remote Manager as its central management server. No configuration changes are required to begin using the Remote Manager.

For information about configuring central management for your IX15 device, see Central management.

Access Digi Remote Manager

To access Digi Remote Manager:

1. If you have not already done so, go to https://myaccount.digi.com/ to sign up for a Digi Remote Manager account.

   Check your email for Digi Remote Manager login instructions.
2. Go to remotemanager.digi.com.
1. Enter your username and password.

   The Digi Remote Manager Dashboard appears.

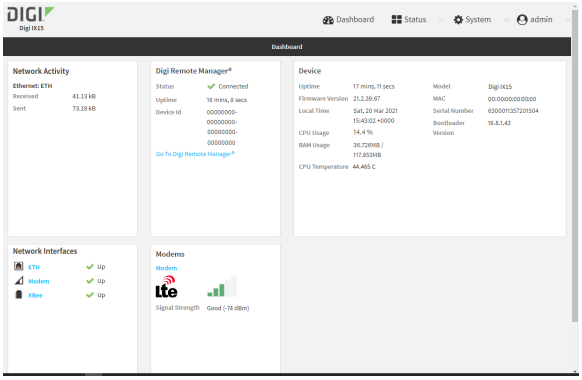
Using the web interface

To connect to the IX15 local WebUI:

1. Use an Ethernet cable to connect the IX15's **ETH** port to a laptop or PC.
2. Open a browser and go to **192.168.2.1**.
3. Log into the device using a configured user name and password.

   The default user name is **admin** and the default password is the unique password printed on the label packaged with your device.

After logging in, the local web admin dashboard is displayed.



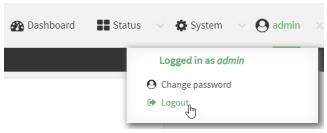
The dashboard shows the current state of the device.

Dashboard area	Description
Network activity	Summarizes network statistics: the total number of bytes sent and received over all configured bridges and Ethernet devices.

Dashboard area	Description
Digi Remote Manager	Displays the device connection status for Digi Remote Manager, the amount of time the connection has been up, and the Digi Remote Manager device ID. See Using Digi Remote Manager.
Device	Displays the IX15 device's status, statistics, and identifying information.
Network Interfaces	Displays the status of the network interfaces configured on the device.
Modems	Provides information about the signal strength and technology of the cellular modem(s).
XBee Network	Shows the XBee network status and information such as the protocol, network identifier, or the number of nodes.

Log out of the web interface

- On the main menu, click your user name. Click **Log out**.



Use the local REST API to configure the IX15 device

Your IX15 device includes a REST API that can be used to return information about the device's configuration and to make modifications to the configuration. You can view the REST API specification from your web browser by opening the URL:

https://*ip-address*/cgi-bin/config.cgi

For example:

https://192.168.210.1/cgi-bin/config.cgi

Use the GET method to return device configuration information

To return device configuration, issue the **GET** method. For example, using **curl**:

```
$ curl -k -u admin https://ip-address/cgi-bin/config.cgi/value/path -X GET
```

where:

- *ip-address* is the IP address of the IX15 device.
- *path* is the path location in the configuration for the information being returned.

  To determine allowed values for *path* from the Admin CLI:

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type **?** (question mark):

```
(config)> ?
auth                        Authentication
cloud                       Central management
firewall                    Firewall
monitoring                  Monitoring
network                     Network
serial                      Serial
service                     Services
system                      System
vpn                         VPN

(config)>
```

   The allowed values for *path* are listed in the first (left) column.
4. To determine further allowed path location values by using the **?** (question mark) with the path name:

```
(config> service ?

Services

 Additional Configuration
 ------------------------------------------------------------------------
------------
 dns                         DNS
 iperf                       IPerf
 location                    Location
 mdns                        Service Discovery (mDNS)
 modbus_gateway              Modbus Gateway
 multicast                   Multicast
 ntp                         NTP
 ping                        Ping responder
 snmp                        SNMP
 ssh                         SSH
 telnet                      Telnet
 web_admin                   Web administration

(config)> service
```

For example, to use **curl** to return the ssh configuration:

```
$ curl -k -u admin https://192.168.210.1/cgi-bin/config.cgi/value/service/ssh -
X GET
Enter host password for user 'admin':
{
ok": true,
       "result": {
                "type": "object",
                "path": "service.ssh"
,               "collapsed": {
"acl.zone.0": "internal"
,
"acl.zone.1": "edge"
,
"acl.zone.2": "ipsec"
,
"acl.zone.3": "setup"
,
"enable": "true"
,
"key": ""
,
"mdns.enable": "true"
,
"mdns.name": ""
,
"mdns.type": "_ssh._tcp."
,
"port": "22"
,
"protocol.0": "tcp"
                }
       }
}
$
```

You can also use the **GET** method to return the configuration parameters associated with an item:

```
curl -k -u admin https://192.168.210.1/cgi-bin/config.cgi/keys/service/ssh -X
GET
Enter host password for user 'admin':
{ "ok": true, "result": [ "acl", "enable", "key", "mdns", "port", "protocol" ]
}
$
```

Use the POST method to modify device configuration parameters and list arrays

Use the POST method to modify device configuration parameters

To modify configuration parameters, use the **POST** method with the **path** and **value** parameters.

```
$ curl -k -u admin "https://ip-address/cgi-
bin/config.cgi/value?path=path&value=new_value" -X POST
```

where:

- *path* is the path to the configuration parameter, in dot notation (for example, **ssh.service.enable**).
- *new_value* is the new value for the parameter.

For example, to disable the ssh service using **curl**:

```
$ curl -k -u admin "https://192.168.210.1/cgi-
bin/config.cgi/value?path=service.ssh.enable&value=false" -X POST
Enter host password for user 'admin':
{ "ok": true }
$
```

Use the POST method to add items to a list array

To add items to a list array, use the **POST** method with the **path** and **append** parameters. For example, to add the external firewall zone to the ssh service:

```
$ curl -k -u admin "https://192.168.210.1/cgi-
bin/config.cgi/value?path=service.ssh.acl.zone&append=true&value=external" -X
POST
Enter host password for user 'admin':
{ "ok": true, "result": "service.ssh.acl.zone.4" }
$
```

Use the POST method to add objects to a list array

Objects in an array that require one or more underlying values can be set using the **collapsed** URI parameter. We recommend including the -g option as well, to instruct curl to turn off globbing. The below example would add a new static route for the WAN interface for the 1.2.4.0/24 destination network:

```
$ curl -g -k -u admin "https://192.168.210.1/cgi-
bin/config.cgi/value?path=network.route.static&append=true&collapsed
[dst]=1.2.4.0/24&collapsed[interface]=/network/interface/wan" -X POST
Enter host password for user 'admin':
{ "ok": true, "result": "network.route.static.1" }
$
```

Use the DELETE method to remove items from a list array

To remove items from a list array, use the **DELETE** method. For example, using **curl**:

```
$ curl -k -u admin "https://192.168.210.1/cgi-bin/config.cgi/value?path=path
```

where *path* is the path to the list item, including the list number, in dot notation (for example, **service.ssh.acl.zone.4**).

For example, to remove the external firewall zone to the ssh service:

1. Use the **GET** method to determine the SSH service's list number for the external zone:

   ```
   $ curl -k -u admin "https://192.168.210.1/cgi-
   bin/config.cgi/value/service/ssh/acl/zone -X GET
   {
          "ok": true,
   ```

```
       "result": {
               "type": "array",
               "path": "service.ssh.acl.zone"
,              "collapsed": {
"0": "internal"
,
"1": "edge"
,
"2": "ipsec"
,
"3": "setup"
,
"4": "external"
               }
       }
}
$
```

2. Use the **DELETE** method to remove the external zone (list item 4).

```
$ curl -k -u admin https://192.168.210.1/cgi-
bin/config.cgi/value?path=service.ssh.acl.zone.4 -X DELETE
Enter host password for user 'admin':
{ "ok": true }
$
```

Using the command line

The Digi IX15 device provides a command-line interface that you can use to configure the device, display status and statistics, update firmware, and manage device files.

See Command line interface for detailed instructions on using the command line interface and see Command line reference for information on available commands.

Access the command line interface

You can access the IX15 command line interface using an SSH connection, a telnet connection, or a serial connection. You can use an open-source terminal software, such as PuTTY or TeraTerm, to access the device through one of these mechanisms.

You can also access the command line interface in the WebUI by using the **Terminal**, or the Digi Remote Manager by using the **Console**.

To access the command line, your device must be configured to allow access, and you must log in as a user who has been configured for the appropriate access. For further information about configuring access to these services, see:

- Serial: Serial port
- WebUI: Configure the web administration service
- SSH: Configure SSH access
- Telnet: Configure telnet access

Log in to the command line interface

Command line

1. Connect to the IX15 device by using a serial connection, SSH or telnet, or the **Terminal** in the WebUI or the **Console** in the Digi Remote Manager. See Access the command line interface for more information.
   - For serial connections, the default configuration is:
     - **115200** baud rate
     - **8** data bits
     - **no** parity
     - **1** stop bit
     - **no** flow control
   - For SSH and telnet connections, the default IP address of the device is**192.168.2.1** on the WAN/ETH1 .
2. At the login prompt, enter the username and password of a user with Admin access:

   ```
   login: admin
   Password: **********
   ```

   The default username is **admin**. The default unique password for your device is printed on the device label.

3. Depending on the device configuration, you may be presented with another menu, for example:

```
Access selection menu:

       a: Admin CLI
       s: Shell
       q: Quit

Select access or quit [admin] :
```

   Type **a** or **admin** to access the IX15 command line.

You will now be connected to the Admin CLI:

```
Connecting now...
 Press Tab to autocomplete commands
 Press '?' for a list of commands and details
 Type 'help' for details on navigating the CLI
 Type 'exit' to disconnect from the Admin CLI

>
```

See Command line interface for detailed instructions on using the command line interface.

Exit the command line interface

Command line

1. At the command prompt, type **exit**.

```
> exit
```

2. Depending on the device configuration, you may be presented with another menu, for example:

```
Access selection menu:

       a: Admin CLI
       s: Shell
       q: Quit

Select access or quit [admin] :
```

   Type **q** or **quit** to exit.

Manage an XBee network

The IX15 is designed to inspect, manage, and work with live XBee networks. The IX15 discovers the devices in the network, lists all the connected nodes, and provides information such as their 64-bit address, node identifier, role, and so on.

The IX15 network caches a list of known nodes that reflects the real XBee network. It adds new nodes to its network cache in these scenarios:

- When any kind of communication occurs between any remote node in the network and the IX15.
- When the IX15 actively searches for remote nodes in the network periodically or by customer request.

You can get information and manage your XBee network using the local web interface.

Access the IX15 local web interface

This section contains the following topics:

Review the current XBee network state 89
Discover the XBee network 89
Configure and update an XBee network 93
Export your network 98

Review the current XBee network state

The XBee Network Manager window displays information about the XBee network and allows you to perform several actions. To open the XBee Network Manager:

1. Open the IX15 WebUI and login.
2. On the top menu, click **XBee**.
3. Under **XBee Administration**, click **Network Manager**.

The Network Manager window contains three main panels:

- **Network Details** panel.
  Displays XBee network general information such as:
  - Network status
  - Pan ID
  - Operating channel
  - Encryption status
  - Number of devices
- **XBee Devices** panel.
  Displays the following information of the local and remote XBee devices in the network:
  - Node identifier
  - Role
  - 64-bit MAC address
  - 16-bit address
  - Firmware version
  - Hardware version
- **Network Actions** panel.
  Contains several actions that can be performed in the network:
  - **Discover Network**: triggers a one-shot network discovery. See One-shot discovery.
  - **Update Network**: takes you to the **Update Manager** window where you can perform profile updates on local and remote XBee devices. See Apply XBee profiles.
  - **Export Network**: Function not yet implemented.

Note See Show the network to list all discovered XBee nodes from the CLI.

Discover the XBee network

Nodes that appear in the **Network Manager** list are known to the IX15. The list accumulates known network nodes over time. To find new nodes, you can perform:

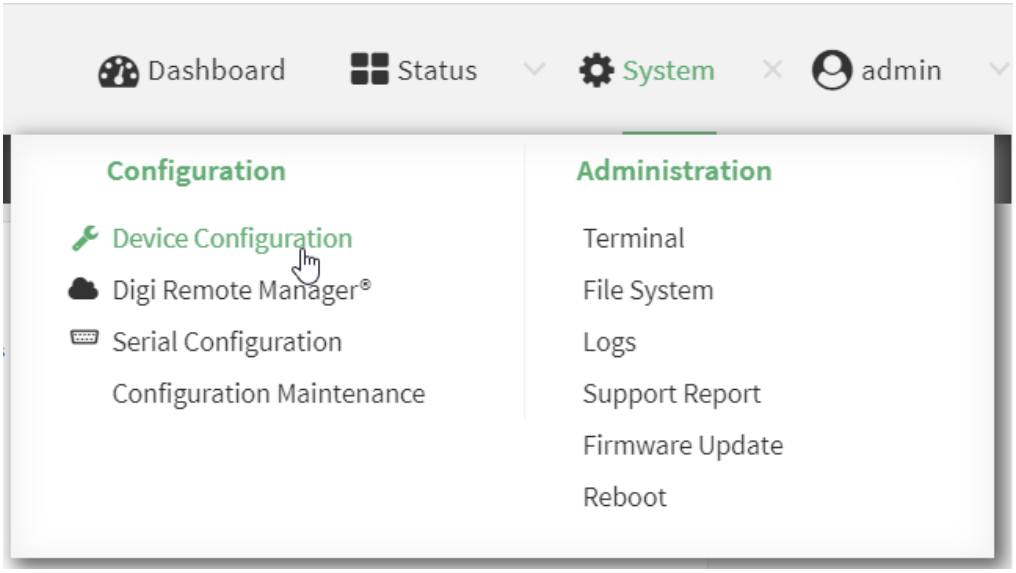
- Periodic active network discovery—configured.
- One-shot discovery—on demand.

Configure active discovery

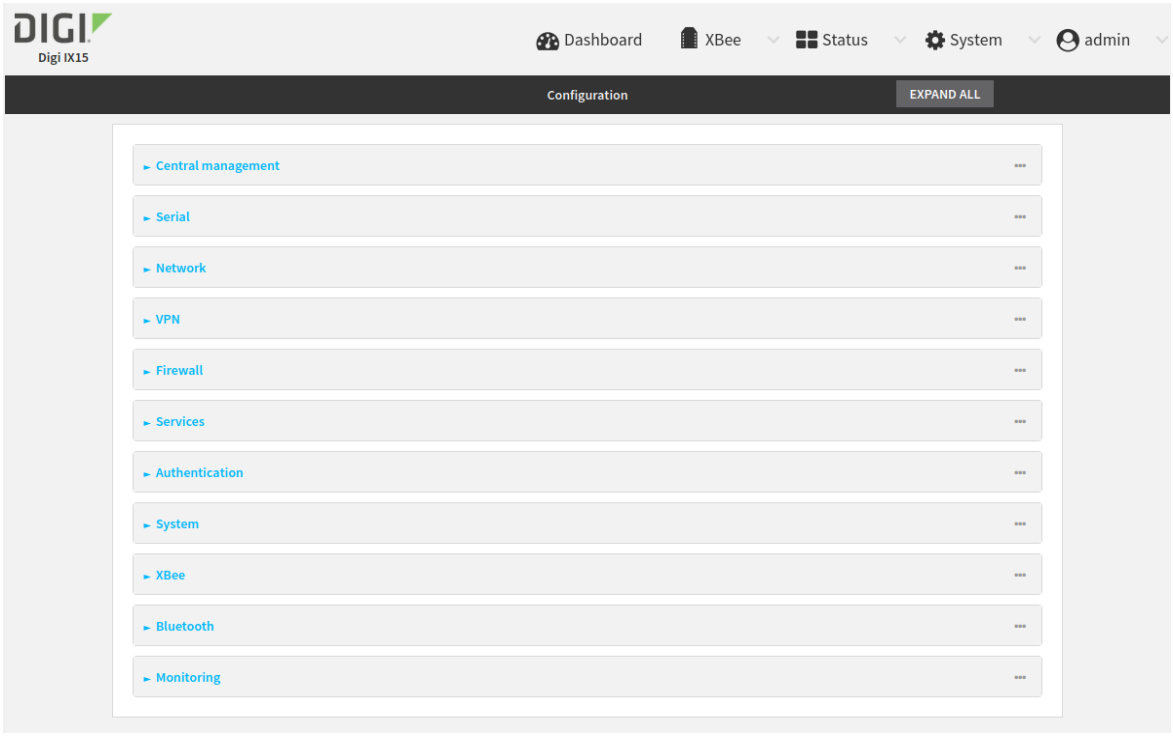
To configure active discovery options:

WebUI

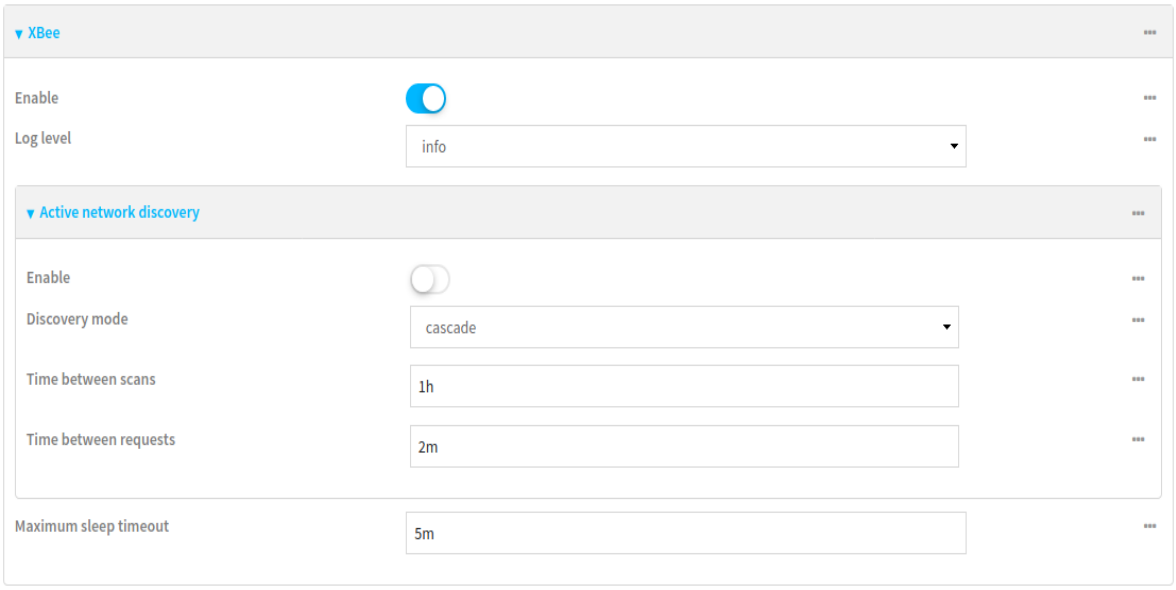
1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window displays.



3. Click **XBee** > **Active Network Discovery Settings** to display the power settings.



4. Click **Enable** to enable active discovery.
5. For **Discovery mode** choose the mode to discover the network:

Allowed values are:

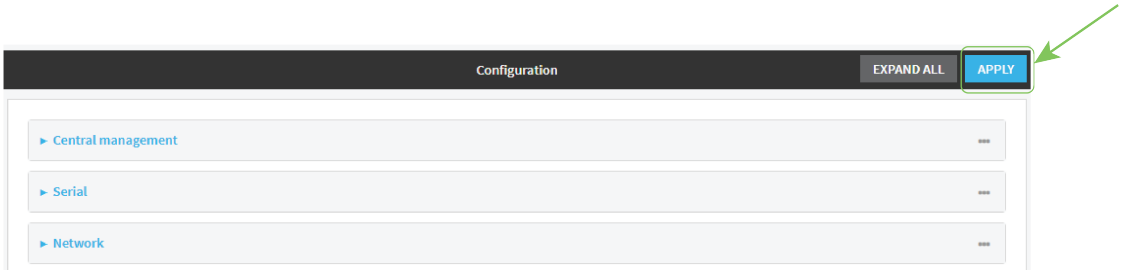
- **cascade**: node neighbors discovery is requested once the previous request finishes and after **Time between requests** elapses. This means only one discovery process is running at the same time. This might be a slower method, but it generates less traffic.
- **flood**: node neighbors discovery is requested when the node is found and after **Time between requests**. This means several discovery processes might be running at the same time. This might be a faster method, but it generates more traffic and might saturate the network. For large networks we recommend **cascade** discovery mode.

  The default is **cascade**.

6. Set **Time between scans**.
   This is the time between network discovery scans.
   Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format number{**w|d|h|m|s**}.
   For example, to set **Time between scans** to two hours, enter **2h** or **120m**.
   The minimum value is 1 minute and the maximum is 3 days. The default is 1 hour.
7. Configure the **Time between requests**.
   This is the time in seconds between node neighbor requests. The default is 60 seconds.
   - For **cascade** discovery mode: time to wait after completion of a node neighbor discovery process and before the next node request.
   - For **flood** discovery mode: minimum time to wait between each neighbor request.

   Allowed values are any number of minutes, or seconds, and take the format *number*{**m|s**}.
   For example, to set **Time between requests** to five minutes, enter **5m** or **300s**.
   The minimum value is 20 seconds and the maximum is 10 minutes. The default is 2 minutes.

8. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable active discovery:

```
(config)> xbee active_discovery enable true
```

4. Set the discovery mode.

```
(config)> config xbee active_discovery mode <discovery_mode>
```

   where **<discovery_mode>** is the discovery mode:

- **cascade**: node neighbors discovery is requested once the previous request finishes and after **Time between requests** elapses. This means only one discovery process is running at the same time. This might be a slower method, but it generates less traffic.
- **flood**: node neighbors discovery is requested when the node is found and after **Time between requests**. This means several discovery processes might be running at the same time. This might be a faster method, but it generates more traffic and might saturate the network. For large networks we recommend **cascade** discovery mode.

  The default is **cascade**.

5. Set the amount of time the IX15 should wait between discovery cycles. Allowed values are from 1 minute to 3 days. The default is 1 hour.

```
(config)> config xbee active_discovery time_between_scans <time>
```

   where **<time>** is any number of weeks, days, hours, minutes, or seconds, and takes the format *number*{**w**|**d**|**h**|**m**|**s**}.

   For example, to set the **time between scans** to two hours, enter either 2h or 120m:

```
(config)> config xbee active_discovery time_between_scans 120m
(config)>
```

6. Set the amount of time the IX15 should wait between node requests to ask for neighbors. Allowed values are from 20 seconds to 10 minutes. The default is 2 minutes.

```
(config)> config xbee active_discovery time_between_requests <time>
```

where **<time>** is any number of minutes, or seconds, and takes the format *number*{**m|s**}.

- For **cascade**: time to wait after completion of a node neighbor discovery process and before the next node request.
- For **flood**: minimum time to wait between each neighbor request.

For example, to set the **time between requests** to five minutes, enter either **5m** or **300s**:

```
(config)> config xbee active_discovery time_between_requests 300s
(config)>
```

7. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

8. Type **exit** to exit the Admin CLI.
Depending on your device configuration, you may be presented with an **Access selection** menu. Type **quit** to disconnect from the device.

One-shot discovery

1. Open the Network Manager
   a. Open the IX15 WebUI and login.
   b. On the top menu, click **XBee**.
   c. Under **XBee Administration**, click **Network Manager**.
2. Perform a one-shot discovery
   a. On the **Network Actions** panel, click **Discover Network**.
   b. On the **Network Discovery** pop up window, select whether you want to **Clear the devices list** before the new discovery and then click **Discover**.
   If you select **Clear devices list before discovering** before discovering, the list of the known nodes is erased. Then only nodes that responded to this one-shot discovery appear in the list.
   c. Nodes that respond to this query appear in the list as a known XBee.

Note See Discover the network for more information on how to perform a one-shot discovery from the CLI.

Configure and update an XBee network

The IX15 provides a gateway between Internet Protocol (IP) network devices and a network of XBee wireless devices. The IX15 supports local and over-the-air (OTA) update in the XBee network nodes. As XBee networks can involve a large number of nodes, IX15 provides a way to automatically schedule

XBee updates and manage profile files. The IX15 allows updating the settings, firmware, and file system of any XBee in the network using XBee profiles.

Note We recommend configuring any XBee, local or remote, using a profile. You may still need to read or set some specific value, for that you can use the CLI. See Configure individual XBee parameters.

- What is an XBee profile?
- Manage XBee profiles
- Upload the XBee profile
- Apply XBee profiles
- Configure a sleeping network to work with the IX15

What is an XBee profile?

A configuration profile is a snapshot of a specific device's firmware configuration, including the firmware, settings values, and other configuration information. A configuration profile is an **XPRO** file containing the following elements:

- Device firmware to be programmed in the XBee.
- Firmware settings to configure with their respective values.
- File system to, for example, include any MicroPython application in the node.
- Other configurations and metadata to identify the profile, such as the flash firmware policy, profile description, and so on.

XCTU allows you to create configuration profiles. A profile is useful in a production environment when you need to set the same parameters on multiple radios.

Note For more information, see Create a configuration profile in the *XCTU User Guide*.

Manage XBee profiles

The Profile Manager page allows you to manage the XBee profiles stored in the IX15. You can upload, download, and delete profiles. For each profile, it also shows whether it contains settings, firmware, and filesystem updates.

To open the XBee Profile Manager:

1. Open the IX15 WebUI and login.
2. On the top menu, click **XBee**.
3. Under **XBee Administration**, click **Profile Manager**.

The Profile Manager window allows your to perform the following actions:

1. Upload a profile
   To upload a profile from your computer, click **Upload Profile** on the top right corner of the **Available XBee Profiles** panel. This opens a local file browser that allows you to select the profile file to upload to the device.
2. Download a profile
   To download a profile from the device to the computer, click **Download Profile** on the right side of the profile row.

3. Delete a profile
   To delete a profile from the device, click **Delete Profile** on the right side of the profile row.

Note You cannot delete the default IX15 profiles: **xbee_gateway_zigbee**, **xbee_gateway_digimesh**, and **xbee_gateway_802.15.4**.

Note An XBee configuration profile may contain settings, firmware, and file system updates for an XBee. The Profile Manager shows whether those contents are included or not for each available profile.

Upload the XBee profile

Once you create the desired XBee profile, follow these steps to upload it to the IX15:

1. Open the IX15 WebUI and login.
2. On the top menu, click **XBee**.
3. Under **XBee Administration**, click **Profile Manager**.
4. Click **Upload Profile** on the top right corner of the **Available XBee Profiles** panel. This opens a local file browser that allows you to select the profile file to upload to the device.
5. Select the profile you want to upload and click **Open**.
6. The profile is automatically transferred to the device and listed in the **Available XBee Profiles** panel.

Apply XBee profiles

The Update Manager allows you to configure and perform update tasks for the XBee devices in the network. While the update tasks are running, the Update Manager shows the progress of the different tasks.

To open the XBee Update Manager:

1. Open the IX15 WebUI and login.
2. On the top menu, click **XBee**.
3. Under **XBee Administration**, click **Update Manager**.

Note You can also get to the **XBee Update Manager** window by clicking **Update Network** in the **Network Actions** panel of the **XBee Network Manager** window.

The Update Manager window allows to perform the following actions:

1. Add update tasks
   a. On the **XBee Update Manager** window, click **Add Update Tasks**.
      A new **Add new update tasks** window displays.
   b. Select the profile from the **Available XBee Profiles** list and click **Next**.
   c. Select the target XBee devices to apply the profile to.
      There are several options to select devices:
      - All devices
      - By role

- By node ID pattern
- Specific devices (manually)

d. Once you have selected the devices, click **Add**. The new task appears in the **Current XBee Update Tasks** panel.

To add more update tasks to the list, click **Add Update Tasks** again.

2. Run update tasks
   a. Click **Start Update** to begin running the update tasks list. The window refreshes showing the update progress.

From the CLI you can:

- Change the configuration of any XBee in the network using a profile. See Apply XBee profiles.
- Get and set a single setting of any XBee in the network. See Configure individual XBee parameters.

Configure a sleeping network to work with the IX15

Nodes in a network can sleep. To work with sleeping nodes, they must be properly configured so the IX15 can manage them.

Zigbee

End devices can sleep long periods of time, but their parents only buffer their messages during a maximum of 30 seconds—see RF packet buffering timeout. This means that end devices with extended sleep periods must notify when they wake to inform the network they can receive data, and stay awake enough time so other nodes, such as the IX15, can send messages to the end device.

Short sleep: nodes that sleep less than 30 seconds

They can receive data transmissions at any time since their parents buffer data long enough for the end devices to wake and poll to receive the data.

- End devices:

1. Set the child table timeout—**ET**—to a greater value than the expected end device sleep time.

- Routers and coordinator:

1. Set the sleep period—**SP**—to the largest value in any node in the network. This ensures that the RF packet buffering, poll timeout, and transmission timeouts are set correctly.

Extended sleep: nodes that sleep more than 30 seconds

They cannot receive data transmissions reliably unless you take certain design approaches.

- End devices:
  1. Must notify when they are awake and can receive data. This can be done by using:
     - I/O sampling: configure to send an IO message when the node awakes.
     - From MicroPython, transmit any data to the IX15 when it wakes up.
     - Any other intelligence that sends anything when the node wakes up.
  2. Must stay awake to provide other nodes, including the IX15, with enough time to send messages to the end device.

a. For cyclic sleep end devices (**SM** = **4**, **SM** = **5**)
   i. Configure **SO** to wake for the full **ST** time.
   ii. Configure a reasonable value for **ST** time to allow transmissions.
b. For pin sleep (**SM** = **1**) and MicroPython sleep (**SM** = **6**) end devices
   i. Stay awake enough time so other nodes can communicate with the end device.

3. Set the child table timeout—**ET**—to a greater value than the expected end device sleep time.

- Routers and coordinator:
  1. **SP** and **SN** should be set such that (**SP** * **SN**) matches the longest expected sleep time.

DigiMesh

Nodes in a DigiMesh network can sleep asynchronously or synchronously.

Synchronous sleep

- The IX15 must be eligible for sleep coordinator:
  - Sleep mode—**SM**—must be **7**: Synchronous cyclic sleep support mode.
  - Sleep options—**SO**—must have bit 1 to **0** to be able to act as sleep coordinator.
- Sleep period—**SP**— set the same value on all nodes in the network.
- Wake time—**ST**—:
  - Set the same value on all nodes in the network.
  - Keep its value sufficiently large so that it won't be affected by an inadvertent change of **NH**, **NN**, **RN**, or **MT**—undesirable overriding syncs.
  - The wake time must be long enough to:
    - Transmit/receive the sync message.
    - Transmit desired data.
    - Provide other nodes with enough time to send messages to sleeping devices.

Asynchronous sleep

Sleeping devices:

1. Must notify when they are awake and can receive data. This can be done by using:
   - I/O sampling: configure to send an IO message when the node awakes.
   - From MicroPython, transmit any data to the IX15 when it wakes up.
   - Any other intelligence that sends anything when the node wakes up.
2. Must stay awake to provide other nodes, including the IX15, with enough time to send messages to the end device.
   - For cyclic sleep end devices (**SM** = **4**, **SM** = **5**)
     a. Configure **SO** to wake for the full **ST** time.
     b. Configure a reasonable value for **ST** time to allow transmissions.
   - For pin sleep (**SM** = **1**) and MicroPython sleep (**SM** = **6**) devices
     a. Stay awake enough time so other nodes can communicate with the sleeping device.

802.15.4

Sleeping devices:

1. Must notify when they are awake and can receive data. This can be done by using:
   - I/O sampling: configure to send an IO message when the node awakes.
   - From MicroPython, transmit any data to the IX15 when it wakes up.
   - Any other intelligence that sends anything when the node wakes up.
2. Must stay awake to provide other nodes, including the IX15, with enough time to send messages to the end device.
   - For cyclic sleep end devices (**SM** = **4**, **SM** = **5**)
     a. Configure **SO** to wake for the full **ST** time.
     b. Configure a reasonable value for **ST** time to allow transmissions.
   - For pin sleep (**SM** = **1**) and MicroPython sleep (**SM** = **6**) devices
     a. Stay awake enough time so other nodes can communicate with the sleeping device.

Export your network

You can export the network to save its current state and import it in the XBee Network Assistant to work offline.

1. Open the Network Manager
   a. Open the IX15 WebUI and login.
   b. On the top menu, click **XBee**.
   c. Under **XBee Administration**, click **Network Manager**.
2. Export the IX15 XBee network to a file
   a. On the **Network Actions** panel, click **Export Network**. The **XBee network** dialog opens.
   b. Type the name of the network and a brief description.
   c. Click **Export**. A *.xnet file with your IX15 XBee network is downloaded.

The exported file can be imported into the XBee Network Assistant to work offline. See Import an XBee network.

Note See Export the network for more information on how to export the network from the CLI.

Bluetooth Low Energy

Bluetooth® Low Energy (BLE) is an RF protocol that enables you to connect your IX15 to another device. Both devices must have BLE enabled.

The IX15 provides Bluetooth Low Energy connectivity through the XBee device interface. The XBee device is able to work in dual mode: XBee protocol + Bluetooth Low Energy (BLE).

For example, you can use your cellphone to connect to the XBee device of the IX15, and then from your phone, interact with the IX15 using the XBee Python API.

Digi created the Digi XBee Mobile SDK, a set of libraries, examples and documentation that help you develop mobile applications to interact with XBee devices through their BLE interface. For this purpose, we provide two easy-to-use libraries that allow you to create XBee mobile native apps:

- XBee Library for Xamarin, to develop cross-platform mobile applications using C# language — iOS and Android.
- XBee Library for Android, to develop Android applications using Java.

The XBee is the server and allows client devices, such as a cellphone, to configure the XBee or data transfer with the User Data Relay frame. The XBee cannot communicate with another XBee over BLE, as the XBee is strictly a BLE server. The possibilities are:

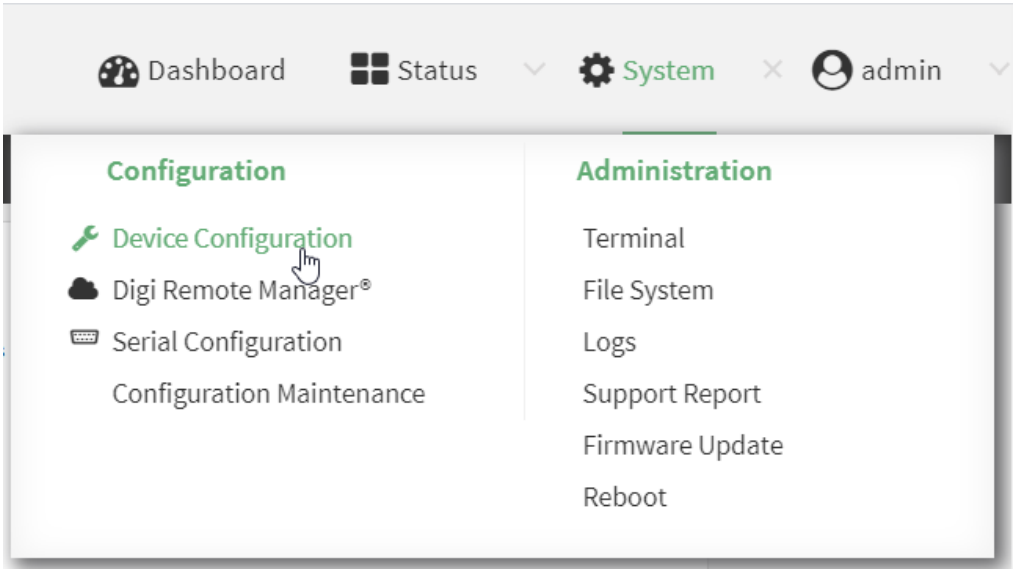
- XBee 3: can communicate with mobile devices over BLE.
- XBee 3: can communicate with third party devices such as the Nordic nRF and SiLabs BGM over BLE.
- XBee 3: cannot communicate with another XBee 3 over BLE.

Configure Bluetooth Low Energy

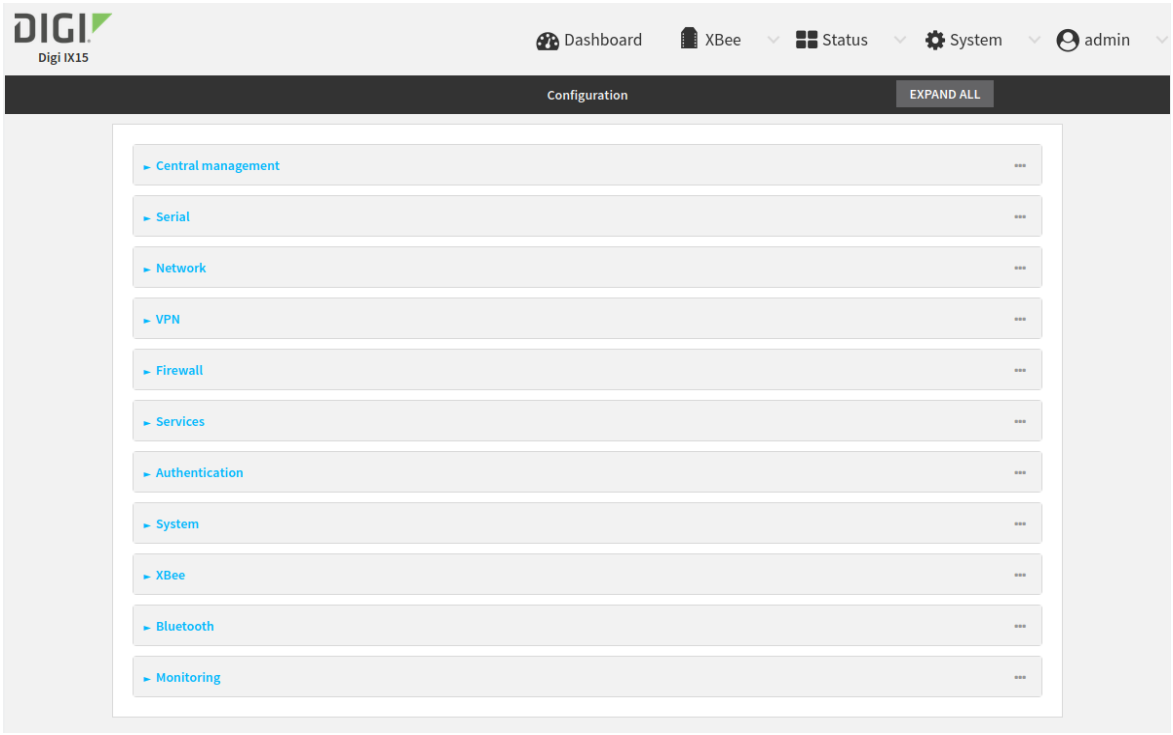
To change the Bluetooth Low Energy configuration:

WebUI

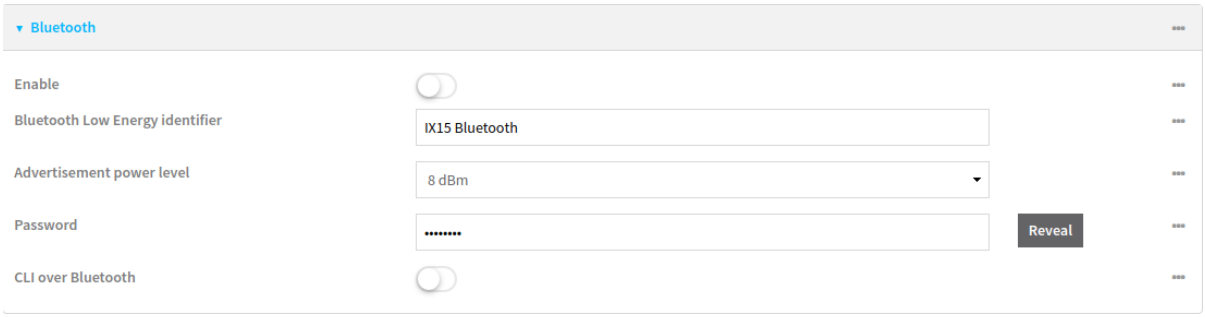
1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window displays.

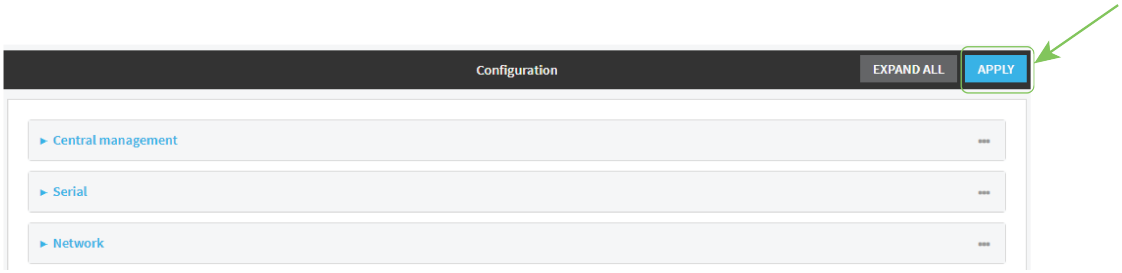


3. Click **Bluetooth** to display the power settings.



4. Click **Enable** to enable Bluetooth Low Energy.
5. For **Bluetooth Low Energy identifier**, type the name for advertising.
6. For **Advertisement power level**, select the output power level to use in BLE advertisements. The allowed values are:
   - -20 dBm
   - -10 dBm
   - 0 dBm
   - 8 dBm

   The default is **8 dBm**.
7. For **Password**, enter the BLE authentication password. This is the password for external applications to establish a BLE communication.
8. Click **CLI over Bluetooth** to be able to execute CLI commands over Bluetooth.
9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable Bluetooth Low Energy:

```
(config)> bluetooth enable true
```

4. Set the identifier.

```
(config)> bluetooth identifier <id>
```

   where **<id>** is the name for BLE advertisements.

5. Set the advertisement power level:

```
(config)> bluetooth advertisement_power <n>
```

   where **<n>** is the power level:

   - **0**: -20 dBm
   - **1**: -10 dBm
   - **2**: 0 dBm
   - **3**: 8 dBm

   The default is **3**.

6. Set the BLE communication password

```
(config)> bluetooth password <ble_password>
```

   where **<ble_password>** is the password to configure.

7. Optionally, you can enable CLI over Bluetooth.

```
(config)> bluetooth enable_cli true
```

8. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.
   Depending on your device configuration, you may be presented with an **Access selection** menu. Type **quit** to disconnect from the device.

IX15 BLE configurator

The IX15 can be configured via Bluetooth. To do so:

1. Enable **Bluetooth** and **CLI over Bluetooth** options. See Configure Bluetooth Low Energy.
2. Use the **IX15BleConfiguratorDemo** app to configure your device via Bluetooth.
   The source code is available at xbee-csharp GitHub repository, inside the examples directory.
   This application is available for iOS or Android. Find it in the corresponding store:
   - Google Play
   - App Store

Verify BLE connectivity

You can use the Digi XBee Mobile application to verify that BLE is enabled and working on your IX15.

1. Download and install the **Digi XBee Mobile** application in your phone.
2. Open the Digi XBee Mobile application. The **Find XBee devices** screen appears and the app automatically begins scanning for devices. All nearby devices with BLE enabled are displayed in a list.
3. Scroll through the list to find your IX15.
   The first time you open the app on a phone and scan for devices, the device list contains only the name of the device and the BLE signal strength. No identifying information for the device displays. After you have authenticated the device, the device information is cached on the phone. The next time the app on this phone connects to the XBee device, the IMEI for the device displays in the app device list.
4. Tap the device name in the list. A password dialog appears.
5. Enter the password you previously configured for the device in the IX15.
6. Tap **OK**. The **Device Information** screen displays. You are now connected to the IX15 through BLE.

Power management

Most of the time the IX15 will be powered by a plug in the wall so power consumption might not be a problem. However, if you plan to power your device with batteries you must consider power saving.

The IX15 offers several options to reduce power consumption during standard operating mode and when required, can make consumption minimal by sending the device to suspend mode.

The following two mechanisms will help you to include power saving when designing your project:

- Power profiles
- Suspend mode

Configure a power profile 105
Suspend mode 109

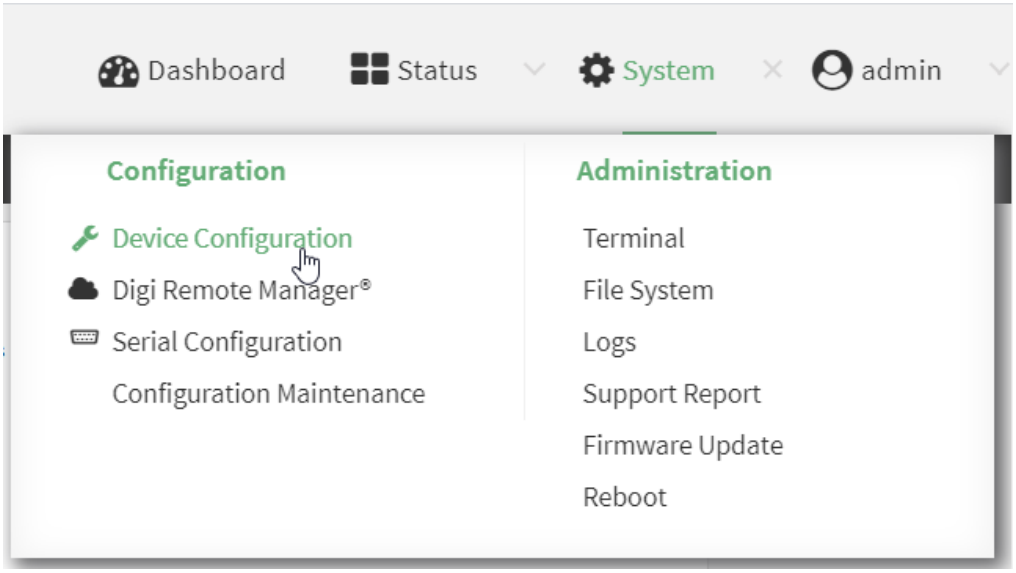
Configure a power profile

A Power profile is a group of settings that determine how the system will behave in terms of power consumption during standard operating mode. You can choose to preserve power, performance or to balance both.

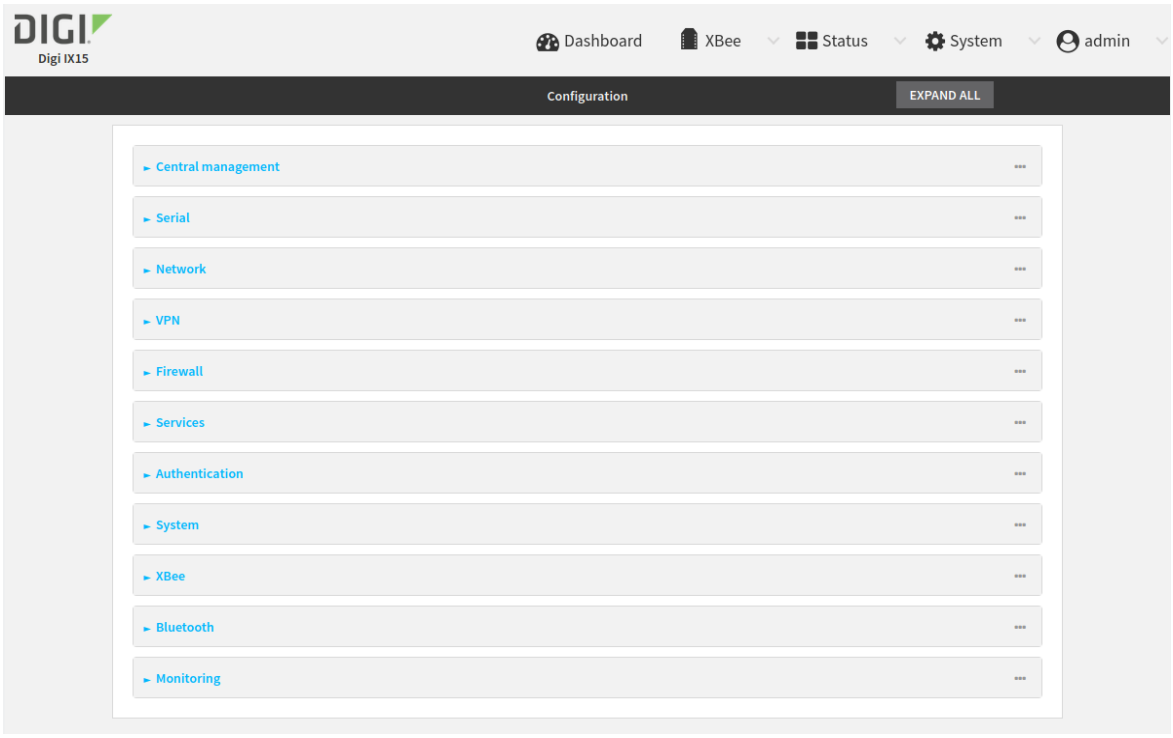
To change the active power profile:

WebUI

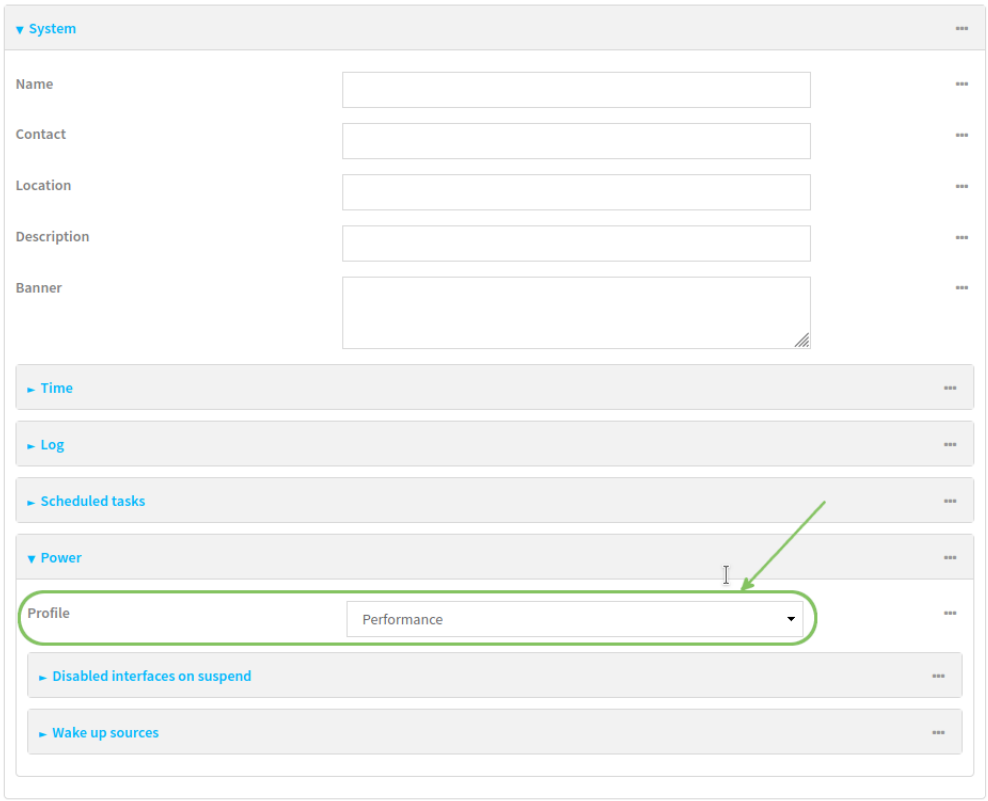
1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window displays.



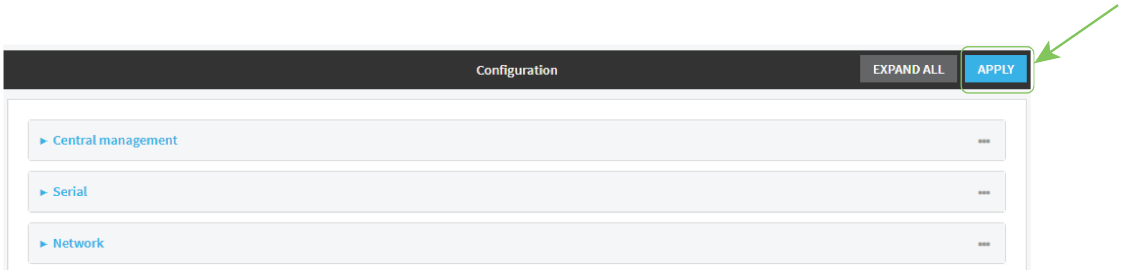
3. Click **System** > **Power** to display the power settings.



4. The **Profile** setting displays the active power profile and allows you to change it. The available options are:

- **Performance**: The CPU clock frequency is scaled up to work in the highest available frequency and provide a better system performance.
- **Auto**: The CPU clock frequency is dynamically scaled up and down to provide better performance during high demanding conditions and also to save power during inactivity periods.
- **Power save**: The CPU clock frequency is scaled down to work in the lowest available frequency and save power.
- **Manual**: Allows you to manually set the working frequency of the CPU. When this option is selected, the setting **Custom frequency** is available to set the CPU working frequency manually:
  - 198 KHz
  - 396 KHz
  - 528 KHz
  - 792 KHz

5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Set the **profile** you prefer:

```
(config)> system power profile <profile_name>
```

   where **<profile_name>** is either:

   - **auto**: The CPU clock frequency is dynamically scaled up and down to provide better performance during high demanding conditions and also to save power during inactivity periods.
   - **manual**: Allows you to manually set the working frequency of the CPU.
   - **performance**: The CPU clock frequency is scaled up to work in the highest available frequency and provide a better system performance.
   - **powersave**: The CPU clock frequency is scaled down to work in the lowest available frequency and save power.

     The default is **performance**.
4. If **profile** is set to **manual**:
   a. Set the CPU working frequency:

```
(config)> system power custom_freq <frequency>
```

   Allowed values are:

   - **198000**
   - **396000**
   - **528000**
   - **792000**

     The default is **792000**.
5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.
   Depending on your device configuration, you may be presented with an **Access selection** menu. Type **quit** to disconnect from the device.

Suspend mode

Suspend mode is a special state where the CPU, most of the RAM, and most of the digital peripherals are powered off to save as much power as possible.

The IX15 is able to enter suspend mode on demand to reduce power consumption to the minimum when no operation is required during a certain time.

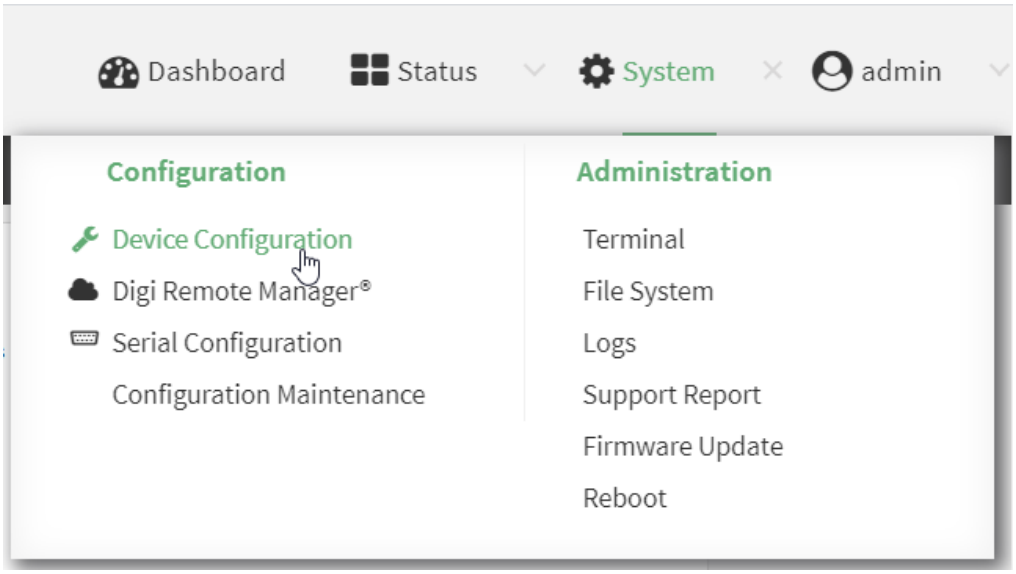
Configure wake up sources

The wake up sources refer to the different mechanisms and triggers used to wake up the device and put it in normal operating mode again when the device is in suspend mode.

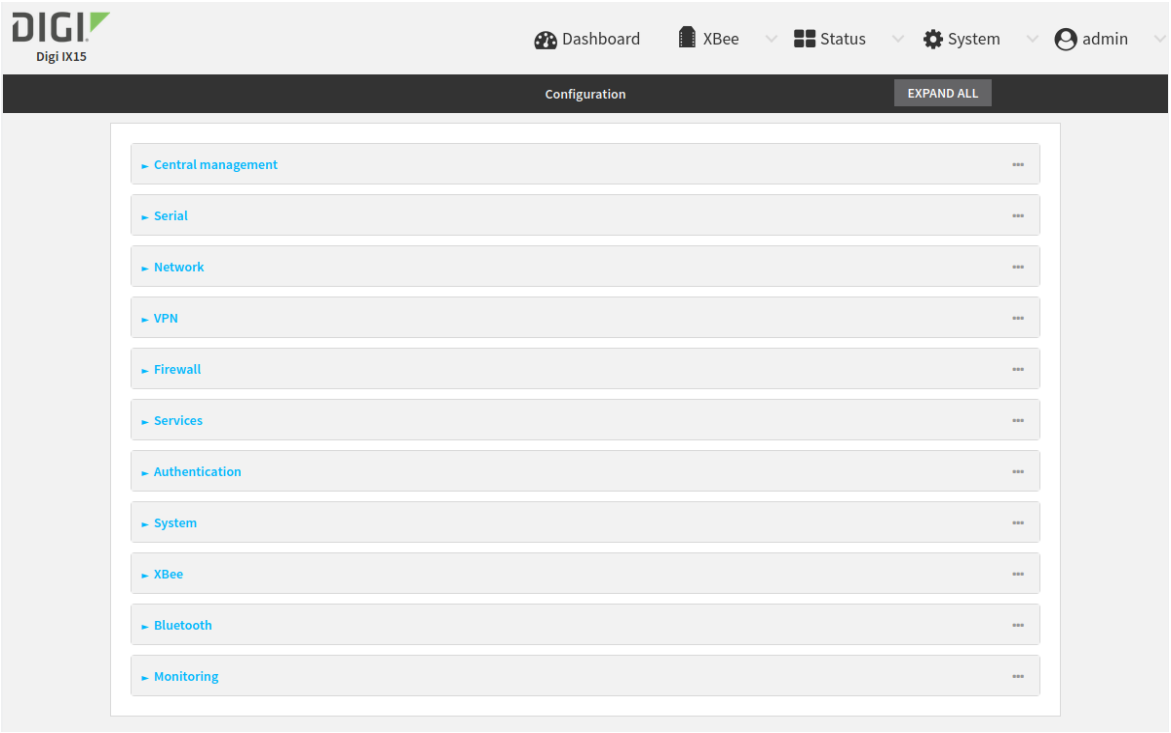
To configure the IX15 wake up sources:

WebUI

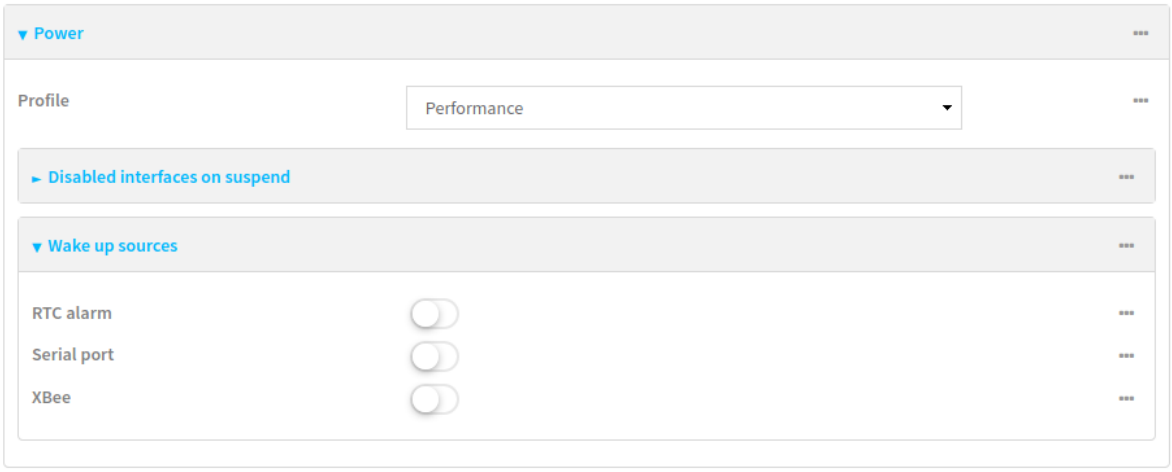
1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window displays.

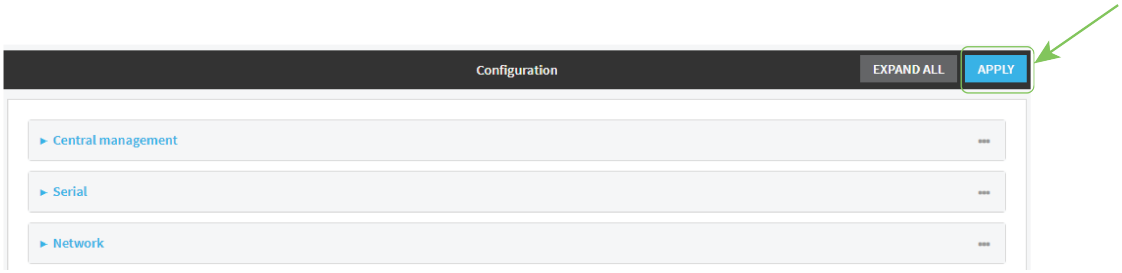


3. Click **System** > **Power** > **Wake up sources**.



4. Toggle on the wake up sources to enable them:
   - **RTC alarm**: Configures an alarm and wake up the device when the alarm triggers.
   - **Serial port**: Wakes up the device when any data is received in the serial port.
   - **XBee**: Wakes up the device when any data is received in the XBee interface.
5. If **RTC alarm** is enabled, set the the alarm date and time in **RTC alarm date and time** following the format: YYYY-MM-DD HH:MM[:SS].

6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Configure the wake up sources:

```
(config)> system power profile <profile_name>
```

   where **<wakeup_src>** is either:
   - **rtc**: Configures an alarm and wake up the device when the alarm triggers.
   - **serial**: Wakes up the device when any data is received in the serial port.
   - **xbee**: Wakes up the device when any data is received in the XBee interface.

     By default none is enabled.
4. If **rtc** is enabled:
   a. Set the RTC alarm date and time:

```
(config)> system power wakeup_sources rtc_time <date>
```

   where **<date>** must follow the format: YYYY-MM-DD HH:MM[:SS]
5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.
   Depending on your device configuration, you may be presented with an **Access selection** menu. Type **quit** to disconnect from the device.

Enter suspend mode

You can command the IX15 to enter suspend mode at any time using the CLI interface. To do so:

1. Connect to the IX15 CLI by using a serial connection, SSH, or the Terminal in the WebUI.
2. Log into the IX15 command line as a user with full Admin access rights.
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
3. Run the following command to send the IX15 to suspend mode:

```
powerctrl state suspend
```

To avoid leaving the device in a forever-suspended state, the IX15 will not enter suspend mode if there is no wake up source configured.

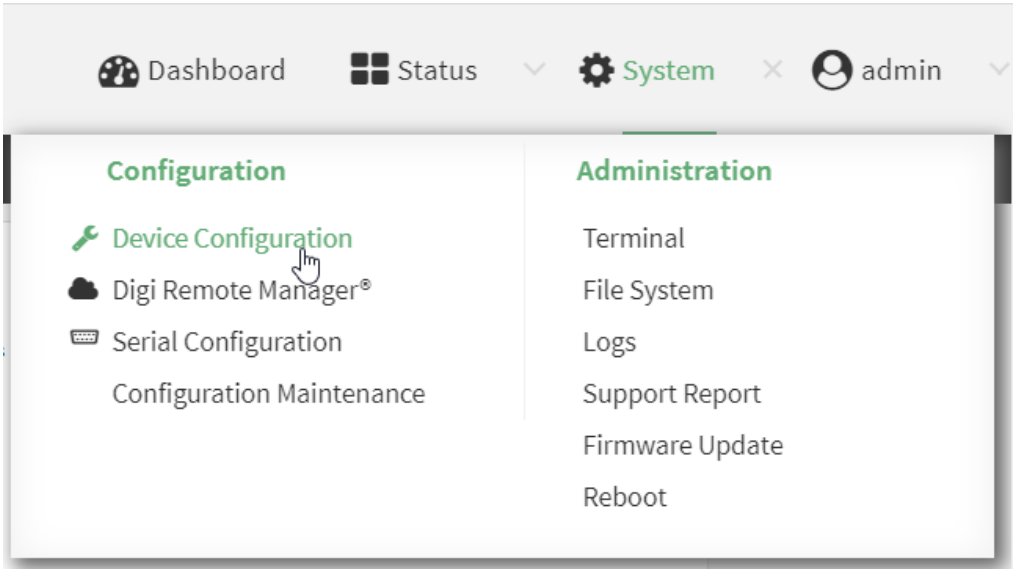
Disable interfaces on suspend

When the IX15 enters suspend mode you can configure whether to disable or not disable some of the device's interfaces.

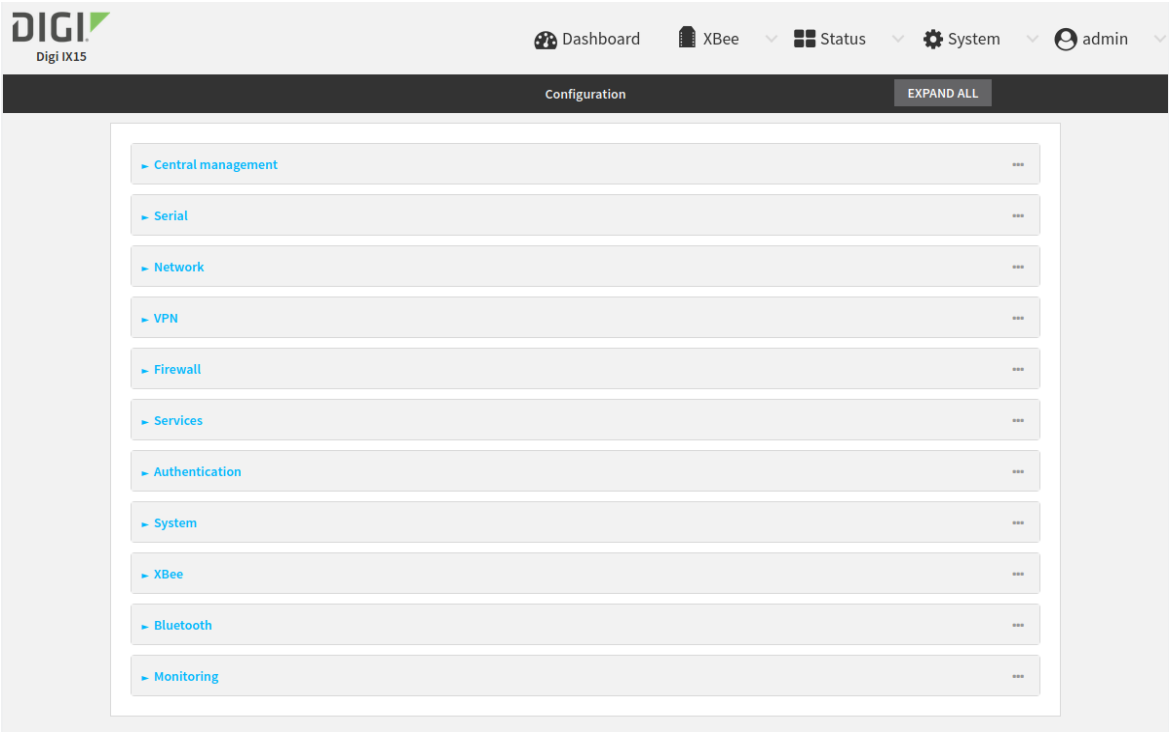
To configure the interfaces to disable on suspend:

WebUI

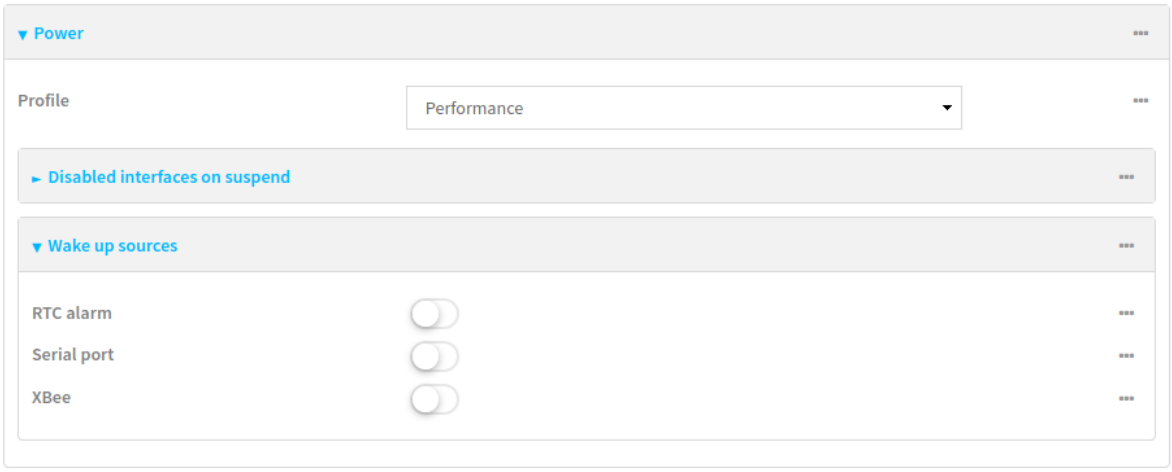
1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window displays.



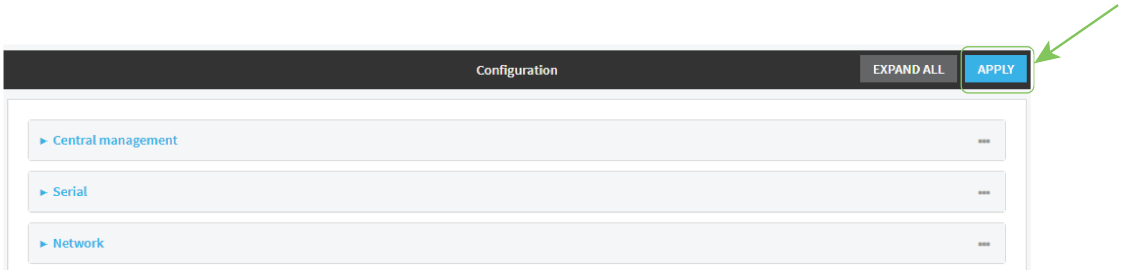
3. Click **System** > **Power** > **Disabled interfaces on suspend**.



4. The interfaces are listed:
   - **Modem**: Turn on this setting to disable the modem interface when the IX15 enters suspend mode.

   By default, all interfaces are enabled when going to suspend. Click available interfaces to toggle them to disable on suspend.

5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Disable the interfaces on suspend:

```
(config)> system power suspend_interfaces <interface> true
```

   where **<interface>** is either:

   - **modem**: Turn on this setting to disable the modem interface when the IX15 enters suspend mode.

     By default none is disabled on suspend.
4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.
   Depending on your device configuration, you may be presented with an **Access selection** menu. Type **quit** to disconnect from the device.

Interfaces

Digi IX15 Gateway devices have several physical communications interfaces. These interfaces can be bridged in a Local Area Network (LAN) or assigned to a Wide Area Network (WAN).

This chapter contains the following topics:

Wireless Wide Area Networks (WWANs) 116
Local Area Networks (LANs) 177
Show Surelink status and statistics 208

Wireless Wide Area Networks (WWANs)

A Wireless Wide Area Network (WWAN) provides connectivity to the internet or a remote network through a cellular connection. A WWAN configuration consists of the following:

- A cellular modem.
- Several networking parameters for the WAN, such as firewall configuration and IPv4 and IPv6 support.
- Several parameters controlling failover.

Configure SureLink active recovery to detect modem failures

Problems can occur beyond the immediate modem connection that prevent some IP traffic from reaching its destination. Normally this kind of problem does not cause the IX15 device to detect that the modem has failed, because the connection continues to work while the core problem exists somewhere else in the network.

Using Digi SureLink, you can configure the IX15 device to regularly probe connections through the modem to determine if the modem connection has failed.

Required configuration items

- Enable SureLink.

  SureLink can be enabled for both IPv4 and IPv6 configurations. By default, SureLink is enabled for IPv4 for the preconfigured **Modem** interface. It is disabled for IPv6.

  When SureLink is configured for Wireless WANs, SureLink tests are only run if the cellular modem is connected and has an IP address. Use the **SIM failover** options to configure the IX15 device to automatically recover the modem in the event that it cannot obtain an IP address. See Configure a Wireless Wide Area Network (WWAN) for details about **SIM failover**.
- The type of probe test to be performed, one of:
  - Test another interface's status: Used to create a failover or coupled relationship between two interfaces. Requires the name of the alternate interface, the IP version to be tested, and the expected status of the alternate interface (either up or down).
  - Ping: Requires the hostname or IP address of the host to be pinged.
  - DNS query: You can perform a DNS query to a named DNS server, or to the DNS servers configured for the WAN.
  - HTTP test: Requires the URL of the host to be tested.
  - Test DNS servers configured for this interface: Tests communication with this interface's DNS servers that are either provided by DHCP, or statically configured.
  - Interface status: Determines if the interface has an IP address assigned to it, that the physical link is up, and that a route is present to send traffic out of the network interface.

  The preconfigured **Modem** interface is configured by default to use SureLink to both test the interface status and perform a test DNS query.

Additional configuration items

- The behavior of the IX15 device upon test failure:
  - The default behavior, which is to restart the **Modem** interface.
  - SIM switching (also enabled by default).

    - Reset the modem.
    - Reboot the device.
- The interval between connectivity tests.
- The number of probe attempts before the Modem interface is considered to have failed.
- The amount of time that the device should wait for a response to a probe attempt before considering it to have failed.
- If the type of probe test is:
    - Ping: Configure the number of bytes in the ping packet.
    - Interface status: Configure the amount of time that the interface is down before it is considered to have failed, and the amount of time it takes to make an initial connection before it is considered down.
- Additional test targets.
- If more than one test target is configured, determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets.

Order of precedence for SureLink actions

If multiple SureLink actions, such as restarting the interface and rebooting the device, are enabled, the following order of precedence is used:

1. Restart interface
2. Switch to the alternate SIM
3. Reset the modem
4. Reboot the device

For example, if SureLink is configured as follows:

- **Restart interface**:
    - **Enabled**
    - **Restart fail count**: **2**
- **Reset modem**:
    - **Enabled**
    - **Reset mode fail count**: **3**
- **Switch SIM**:
    - **Enabled**
    - **Switch SIM fail count**: **5**
- **Reboot device**:
    - **Enabled**
    - **Reboot fail count**: **7**

SureLink will: The device will:

1. First SureLink failure: Nothing will happen.
2. Second SureLink failure: The interface will restart.
3. Third SureLink failure: The modem will reset.
4. Fourth SureLink failure: The interface will restart again.

5. Fifth SureLink failure: The device will switch to the alternate SIM.

   Note The **Switch SIM** behavior only applies if the modem is connected, but SureLink tests are failing. If the modem is not connected, **SIM failover** applies. See Configure a Wireless Wide Area Network (WWAN) for more information about the **SIM failover** option.

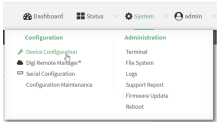
6. Sixth SureLink failure: The interface will restart again.
7. Seventh Surelink failure: The device will reboot.

To configure the IX15 device to regularly probe connections through the WWAN:

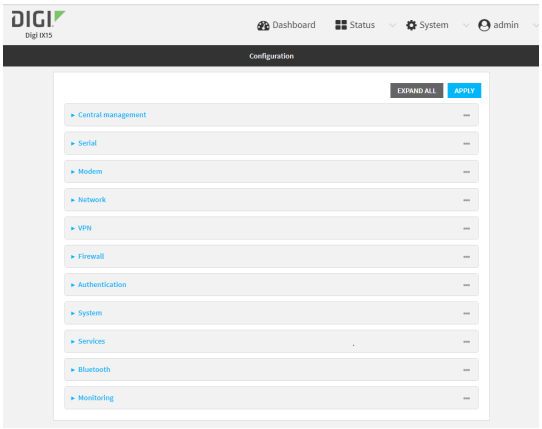
WebUI

SureLink can be configured for both IPv4 and IPv6.

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

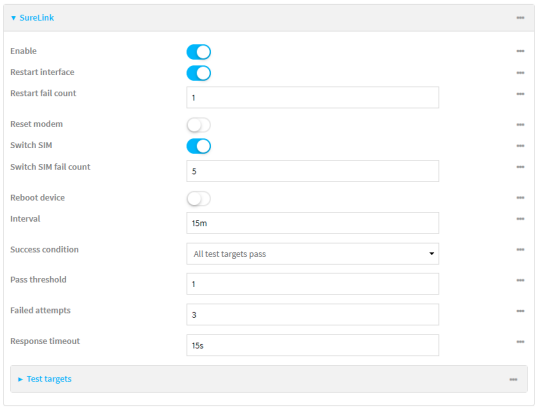


   The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. Create a new WWAN or select an existing one:
   - To create a new WWAN, see Configure a Wireless Wide Area Network (WWAN).
   - To edit an existing WWAN, click to expand the appropriate WWAN.

5. After creating or selecting the WWAN, click **IPv4** (or **IPv6**) > **SureLink**.



6. **Enable** SureLink.

   SureLink can be enabled for both IPv4 and IPv6 configurations. By default, SureLink is enabled for IPv4 for the preconfigured WWAN (**Modem**). It is disabled for IPv6.

7. **Restart interface** is enabled by default.
   - (Optional) For **Restart fail count**, type or select the number of times that the Surelink test must fail before the interface is restarted. The default is **1**.

8. (Optional) Enable **Reset modem**.
   - (Optional) For **Reset modem fail count**, type or select the number of times that the Surelink test must fail before the modem is reset. The default is **1**.

9. **Switch SIM** is enabled by default. Click to disable.
   - If **Switch SIM** is enabled, for **Switch SIM fail count**, type or select the number of times that the Surelink test must fail before the modem switches to the alternate SIM. The default is **5**.

Note The SureLink **Switch SIM** option differs from the **SIM failover** option, which is set during WWAN configuration. **SIM failover** applies when the modem is unable to connect to a cellular network, whereas **Switch SIM** applies when the WWAN interface is failing SureLink tests even though the modem is connected. See Configure a Wireless Wide Area Network (WWAN) for more information about the **SIM failover** option.

10. (Optional) Enable **Reboot device**.

    If **Reboot device** is enabled at the same time as either **Restart interface** or **Reset modem**, **Reboot device** takes precedence.
    - For **Reboot fail count**, type or select the number of times that the Surelink test must fail before the device is rebooted. The default is **1**.

11. Click to expand **Test targets**.
12. For **Add Test Target**, click ✚.



13. Select the **Test type**:
    - **Test another interface's status**: Allows you to test another interface's status, to create a failover or coupled relationship between interfaces. If **Test another interface's status** is selected:
      - For **Test Interface**, select the alternate interface to be tested.
      - For **IP version**, select the alternate interface's IP version. This allows you to determine the alternate interface's status for a particular IP version.
      - For **Expected status**, select whether the expected status of the alternate interface is **Up** or **Down**. For example, if **Expected status** is set to **Down**, but the alternate interface is determined to be up, then this test will fail.
    - **Ping test**: Tests connectivity by sending an ICMP echo request to the hostname or IP address specified in **Ping host**. You can also optionally change the number of bytes in the **Ping payload size**.
    - **DNS test**: Tests connectivity by sending a DNS query to the specified **DNS server**.
    - **HTTP test**: Tests connectivity by sending an HTTP or HTTPS GET request to the URL specified in **Web servers**. The URL should take the format of **http[s]://*hostname*/[*path*]**.
    - **Test DNS servers configured for this interface**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
    - **Test the interface status**: The interface is considered to be down based on:
      - **Down time**: The amount of time that the interface can be down before this test is considered to have failed.

        Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

        For example, to set **Down time** to ten minutes, enter **10m** or **600s**.

        The default is 60 seconds.
      - **Initial connection time**: The amount of time to wait for an initial connection to the interface before this test is considered to have failed.

        Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

        For example, to set **Initial connection time** to ten minutes, enter **10m** or **600s**.

        The default is 60 seconds.
14. Optional active recovery configuration parameters:
    a. Change the **Interval** between connectivity tests.

       Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

       For example, to set **Interval** to ten minutes, enter **10m** or **600s**.

       The default is 15 minutes.

    b. If more than one test target is configured, for **Success condition**, determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets.

    c. For **Pass threshold**, type or select the number of times that the test must pass after failure, before the interface is determined to be working and is reinstated.

d. For **Failed attempts**, type the number of probe attempts before the WAN is considered to have failed.

e. For **Response timeout**, type the amount of time that the device should wait for a response to a probe attempt before considering it to have failed.

Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

For example, to set **Response timeout** to ten minutes, enter **10m** or **600s**.

The default is 15 seconds.

15. (Optional) Repeat this procedure for IPv6.
16. Click **Apply** to save the configuration and apply the change.



Command line

Active recovery can be configured for both IPv4 and IPv6. These instructions are for IPv4; to configure IPv6 active recovery, replace **ipv4** in the command line with **ipv6**.

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Create a new WWAN, or edit an existing one:
   - To create a new WWAN, see Configure a Wireless Wide Area Network (WWAN).
   - To edit an existing WWAN, change to the WWAN's node in the configuration schema. For example, for a WWAN named **my_wwan**, change to the **my_wwan** node in the configuration schema:

```
(config)> network interface my_wwan
(config network interface my_wwan)>
```

4. Enable SureLink.

   SureLink can be enabled for both IPv4 and IPv6 configurations. By default, SureLink is enabled for IPv4 for the preconfigured WWAN (modem). It is disabled for IPv6.

```
(config network interface my_wwan> ipv4 surelink enable true
(config network interface my_wwan)>
```

5. By default, the device is configured to restart the interface when its connection is considered to have failed.

To disable:

```
(config network interface my_wwan ipv4 surelink)> restart false
(config network interface my_wwan ipv4 surelink>
```

- (Optional) Set the number of times that the Surelink test must fail before the interface is restarted:

  ```
  (config network interface my_wwan ipv4 surelink)> restart_attempts
  int
  (config network interface my_wwan ipv4 surelink>
  ```

  where *int* is any number greater than **0**. The default is **1**.

6. (Optional) Set the device to reset the modem:

```
(config network interface my_wwan ipv4 surelink)> reset_modem true
(config network interface my_wwan ipv4 surelink>
```

- If **reset_modem** is enabled, set the number of times that Surelink tests must fail prior to resetting the modem:

  ```
  (config network interface my_wwan ipv4 surelink)> reset_modem_
  attempts int
  (config network interface my_wwan ipv4 surelink>
  ```

  where *int* is an integer between **1** through **5**. The default is **1**.

7. SIM switching is enabled by default. To disable:

```
(config network interface my_wwan ipv4 surelink)> switch_sim false
(config network interface my_wwan ipv4 surelink>
```

Note The SureLink **switch_sim** option differs from the **sim_failover** option, which is set during WWAN configuration. **sim_failover** applies when the modem is unable to connect to a cellular network, whereas **switch_sim** applies when the WWAN interface is failing SureLink tests even though the modem is connected. See Configure a Wireless Wide Area Network (WWAN) for more information about the **sim_failover** option.

- If **switch_sim** is enabled, set the number of times that Surelink tests must fail prior to switching SIMS:

  ```
  (config network interface my_wwan ipv4 surelink)> switch_sim_
  attempts int
  (config network interface my_wwan ipv4 surelink>
  ```

  where *int* is an integer between **1** through **5**. The default is **5**.

8. (Optional) Set the device to reboot when the interface is considered to have failed:

```
(config network interface my_wwan ipv4 surelink)> reboot true
(config network interface my_wwan ipv4 surelink>
```

Note If the reboot parameter is enabled at the same time as either the **restart** or **reset_modem** parameters, the **reboot** parameter takes precedence.

- (Optional) Set the number of times that the Surelink test must fail before the device is rebooted:

```
(config network interface my_wwan ipv4 surelink)> reboot_attempts int
(config network interface my_wwan ipv4 surelink>
```

where *int* is any number greater than **0**. The default is **1**.

9. Add a test target:

```
(config network interface my_wwan)> add ipv4 surelink target end
(config network interface my_wwan ipv4 surelink target 0)>
```

10. Set the test type:

```
(config network interface my_wwan ipv4 surelink target 0)> test value
(config network interface my_wwan ipv4 surelink target 0)>
```

where *value* is one of:

- **ping**: Tests connectivity by sending an ICMP echo request to a specified hostname or IP address.
  - Specify the hostname or IP address:

```
(config network interface my_wwan ipv4 surelink target 0)> ping_host host
(config network interface my_wwan ipv4 surelink target 0)>
```

  - (Optional) Set the size, in bytes, of the ping packet:

```
(config network interface my_wwan ipv4 surelink target 0)> ping_size [num]
(config network interface my_wwan ipv4 surelink target 0)>
```

- **dns**: Tests connectivity by sending a DNS query to the specified DNS server.
  - Specify the DNS server. Allowed value is the IP address of the DNS server.

```
(config network interface my_wwan ipv4 surelink target 0)> dns_server ip_address
(config network interface my_wwan ipv4 surelink target 0)>
```

- **dns_configured**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
- **http**: Tests connectivity by sending an HTTP or HTTPS GET request to the specified URL.
  - Specify the url:

```
(config network interface my_wwan ipv4 surelink target 0)> http_
url value
(config network interface my_wwan ipv4 surelink target 0)>
```

where *value* uses the format **http[s]://*hostname*/[*path*]**

- **interface_up**: The interface is considered to be down based on the interfaces down time, and the amount of time an initial connection to the interface takes before this test is considered to have failed.
  - (Optional) Set the amount of time that the interface can be down before this test is considered to have failed:

    ```
    (config network interface my_wwan ipv4 surelink target 0)>
    interface_down_time value
    (config network interface my_wwan ipv4 surelink target 0)>
    ```

    The default is 60 seconds.
  - (Optional) Set the amount of time to wait for an initial connection to the interface before this test is considered to have failed:

    ```
    (config network interface my_wwan ipv4 surelink target 0)>
    interface_timeout value
    (config network interface my_wwan ipv4 surelink target 0)>
    ```

    The default is 60 seconds.
  - **other**: Allows you to test another interface's status, to create a failover or coupled relationship between interfaces:

    ```
    (config network interface my_wwan ipv4 surelink target 0)> other
    value
    (config network interface my_wwan ipv4 surelink target 0)>
    ```

    If **other** is set:
    - Set the alternate interface to be tested:
      i. Use the **?** to determine available interfaces:
      ii. Set the interface. For example:

      ```
      (config network interface my_wan ipv4 surelink target
      0)> other_interface /network/interface/eth1
      (config network interface my_wan ipv4 surelink target
      0)>
      ```
    - Set the alternate interface's IP version. This allows you to determine the alternate interface's status for a particular IP version.

      ```
      (config network interface my_wwan ipv4 surelink target 0)>
      other_ip_version value
      (config network interface my_wwan ipv4 surelink target 0)>
      ```

      where *value* is one of: **any**, **both**, **ipv4**, or **ipv6**.

- Set the expected status of the alternate interface:

```
(config network interface my_wwan ipv4 surelink target 0)>
other_status value
(config network interface my_wwan ipv4 surelink target 0)>
```

where *value* is either **up** or **down**. For example, if **other_status** is set to **down**, but the alternate interface is determined to be up, then this test will fail.

(Optional) Repeat to add additional test targets.

11. Optional active recovery configuration parameters:
    a. Move back two levels in the configuration by typing **.. ..**:

```
(config network interface my_wwan ipv4 surelink target 0)> .. ..
(config network interface my_wwan ipv4 surelink>
```

    b. Set the **Interval** between connectivity tests:

```
(config network interface my_wwan ipv4 surelink)> interval value
(config network interface my_wwan ipv4 surelink>
```

    The default is 15 minutes.

    c. If more than one test target is configured, determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets:

```
(config network interface my_wwan ipv4 surelink)> success_condition
value
(config network interface my_wwan ipv4 surelink>
```

    Where *value* is either **one** or **all**.

    d. Set the number of probe attempts before the WAN is considered to have failed:

```
(config network interface my_wwan ipv4 surelink)> attempts num
(config network interface my_wwan ipv4 surelink>
```

    The default is **3**.

    e. Set the amount of time that the device should wait for a response to a probe attempt before considering it to have failed:

```
(config network interface my_wwan ipv4 surelink)> timeout value
(config network interface my_wwan ipv4 surelink>
```

    The default is 15 seconds.

12. (Optional) Repeat this procedure for IPv6.
13. Save the configuration and apply the change:

```
(config network interface my_wwan ipv4 surelink)> save
Configuration saved.
>
```

14. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure the device to reboot when a failure is detected

Using SureLink, you can configure the IX15 device to reboot when it has determined that an interface has failed.

Required configuration items

- Enable SureLink.

  SureLink can be enabled for both IPv4 and IPv6 configurations. By default, SureLink is enabled for IPv4 for the preconfigured **Modem** interface. It is disabled for IPv6.

  When SureLink is configured for Wireless WANs, SureLink tests are only run if the cellular modem is connected and has an IP address. Use the **SIM failover** options to configure the IX15 device to automatically recover the modem in the event that it cannot obtain an IP address. See Configure a Wireless Wide Area Network (WWAN) for details about **SIM failover**.
- Enable device reboot upon interface failure.
- The type of probe test to be performed, one of:
  - Test another interface's status: Used to create a failover or coupled relationship between two interfaces. Requires the name of the alternate interface, the IP version to be tested, and the expected status of the alternate interface (either up or down).
  - Ping: Requires the hostname or IP address of the host to be pinged.
  - DNS query: You can perform a DNS query to a named DNS server, or to the DNS servers configured for the WAN.
  - HTTP test: Requires the URL of the host to be tested.
  - Test DNS servers configured for this interface: Tests communication with this interface's DNS servers that are either provided by DHCP, or statically configured.
  - Interface status: Determines if the interface has an IP address assigned to it, that the physical link is up, and that a route is present to send traffic out of the network interface.

Additional configuration items

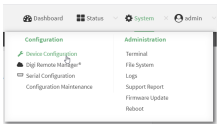
- See Configure SureLink active recovery to detect modem failures for optional SureLink configuration parameters.

To configure the IX15 device to reboot when an interface has failed:

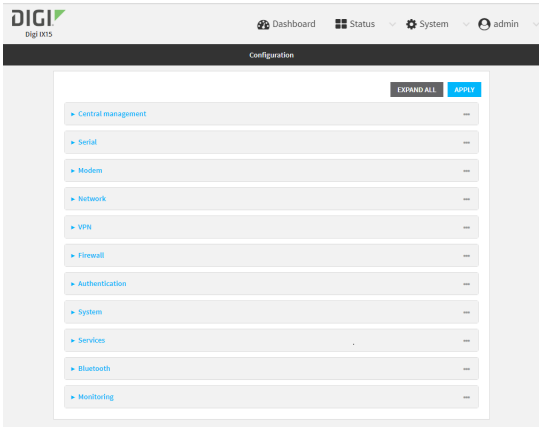
WebUI

SureLink can be configured for both IPv4 and IPv6.

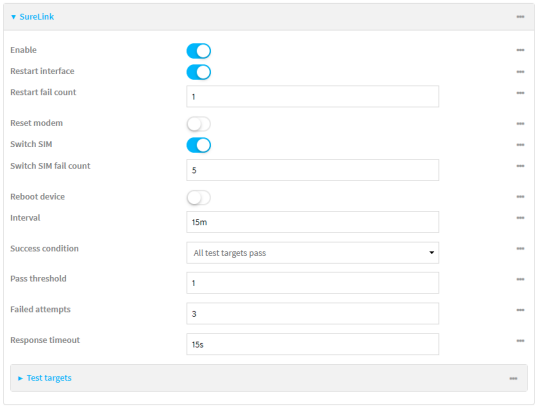
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. Create a new interface or select an existing one:
   - To create a new interface, see Configure a Wireless Wide Area Network (WWAN).
   - To edit an existing interface, click to expand the appropriate interface.
5. After creating or selecting the interface, click **IPv4** (or **IPv6**) > **SureLink**.



6. **Enable** SureLink.

SureLink can be enabled for both IPv4 and IPv6 configurations. By default, SureLink is enabled for IPv4 for the preconfigured WWAN (**Modem**). It is disabled for IPv6.

7. **Restart interface** is enabled by default.
   - (Optional) For **Restart fail count**, type or select the number of times that the Surelink test must fail before the interface is restarted. The default is **1**.
8. Enable **Reboot device**.

   If **Reboot device** is enabled at the same time as **Restart interface**, **Reboot device** takes precedence.
   - For **Reboot fail count**, type or select the number of times that the Surelink test must fail before the device is rebooted. The default is **1**.
9. Click to expand **Test targets**.
10. For **Add Test Target**, click ✚.



11. Select the **Test type**:
    - **Test another interface's status**: Allows you to test another interface's status, to create a failover or coupled relationship between interfaces. If **Test another interface's status** is selected:
      - For **Test Interface**, select the alternate interface to be tested.
      - For **IP version**, select the alternate interface's IP version. This allows you to determine the alternate interface's status for a particular IP version.
      - For **Expected status**, select whether the expected status of the alternate interface is **Up** or **Down**. For example, if **Expected status** is set to **Down**, but the alternate interface is determined to be up, then this test will fail.
    - **Ping test**: Tests connectivity by sending an ICMP echo request to the hostname or IP address specified in **Ping host**. You can also optionally change the number of bytes in the **Ping payload size**.
    - **DNS test**: Tests connectivity by sending a DNS query to the specified **DNS server**.
    - **HTTP test**: Tests connectivity by sending an HTTP or HTTPS GET request to the URL specified in **Web servers**. The URL should take the format of **http[s]://*hostname*/[*path*]**.
    - **Test DNS servers configured for this interface**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
    - **Test the interface status**: The interface is considered to be down based on:
      - **Down time**: The amount of time that the interface can be down before this test is considered to have failed.

        Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

        For example, to set **Down time** to ten minutes, enter **10m** or **600s**.

        The default is 60 seconds.
      - **Initial connection time**: The amount of time to wait for an initial connection to the interface before this test is considered to have failed.

Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

For example, to set **Initial connection time** to ten minutes, enter **10m** or **600s**.

The default is 60 seconds.

12. Optional active recovery configuration parameters:
    a. Change the **Interval** between connectivity tests.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Interval** to ten minutes, enter **10m** or **600s**.

    The default is 15 minutes.

    b. If more than one test target is configured, for **Success condition**, determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets.

    c. For **Pass threshold**, type or select the number of times that the test must pass after failure, before the interface is determined to be working and is reinstated.

    d. For **Failed attempts**, type the number of probe attempts before the WAN is considered to have failed.

    e. For **Response timeout**, type the amount of time that the device should wait for a response to a probe attempt before considering it to have failed.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Response timeout** to ten minutes, enter **10m** or **600s**.

    The default is 15 seconds.
13. (Optional) Repeat this procedure for IPv6.
14. Click **Apply** to save the configuration and apply the change.



Command line

Active recovery can be configured for both IPv4 and IPv6. These instructions are for IPv4; to configure IPv6 active recovery, replace **ipv4** in the command line with **ipv6**.

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Create a new interface, or edit an existing one:
   - To create a new interface, see Configure a Wireless Wide Area Network (WWAN).
   - To edit an existing interface, change to the interface's node in the configuration schema. For example, for a interface named **my_wwan**, change to the **my_wwan** node in the configuration schema:

     ```
     (config)> network interface my_wwan
     (config network interface my_wwan)>
     ```

4. Enable SureLink.

   SureLink can be enabled for both IPv4 and IPv6 configurations. By default, SureLink is enabled for IPv4 for the preconfigured WWAN (modem). It is disabled for IPv6.

   ```
   (config network interface my_wwan> ipv4 surelink enable true
   (config network interface my_wwan)>
   ```

5. By default, the device is configured to restart the interface when its connection is considered to have failed.

   To disable:

   ```
   (config network interface my_wwan ipv4 surelink)> restart false
   (config network interface my_wwan ipv4 surelink>
   ```

   - (Optional) Set the number of times that the Surelink test must fail before the interface is restarted:

     ```
     (config network interface my_wwan ipv4 surelink)> restart_attempts
     int
     (config network interface my_wwan ipv4 surelink>
     ```

     where *int* is any number greater than **0**. The default is **1**.

6. Set the device to reboot when the interface is considered to have failed:

   ```
   (config network interface my_wwan ipv4 surelink)> reboot true
   (config network interface my_wwan ipv4 surelink>
   ```

   Note If the reboot parameter is enabled at the same time as the **restart** parameter, the **reboot** parameter takes precedence.

   - (Optional) Set the number of times that the Surelink test must fail before the device is rebooted:

     ```
     (config network interface my_wwan ipv4 surelink)> reboot_attempts
     int
     (config network interface my_wwan ipv4 surelink>
     ```

     where *int* is any number greater than **0**. The default is **1**.

7. Add a test target:

```
(config network interface my_wwan)> add ipv4 surelink target end
(config network interface my_wwan ipv4 surelink target 0)>
```

8. Set the test type:

```
(config network interface my_wwan ipv4 surelink target 0)> test value
(config network interface my_wwan ipv4 surelink target 0)>
```

where *value* is one of:

- **ping**: Tests connectivity by sending an ICMP echo request to a specified hostname or IP address.
  - Specify the hostname or IP address:

    ```
    (config network interface my_wwan ipv4 surelink target 0)> ping_
    host host
    (config network interface my_wwan ipv4 surelink target 0)>
    ```

  - (Optional) Set the size, in bytes, of the ping packet:

    ```
    (config network interface my_wwan ipv4 surelink target 0)> ping_
    size [num]
    (config network interface my_wwan ipv4 surelink target 0)>
    ```

- **dns**: Tests connectivity by sending a DNS query to the specified DNS server.
  - Specify the DNS server. Allowed value is the IP address of the DNS server.

    ```
    (config network interface my_wwan ipv4 surelink target 0)> dns_
    server ip_address
    (config network interface my_wwan ipv4 surelink target 0)>
    ```

- **dns_configured**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
- **http**: Tests connectivity by sending an HTTP or HTTPS GET request to the specified URL.
  - Specify the url:

    ```
    (config network interface my_wwan ipv4 surelink target 0)> http_
    url value
    (config network interface my_wwan ipv4 surelink target 0)>
    ```

    where *value* uses the format **http[s]://*hostname*/[*path*]**
- **interface_up**: The interface is considered to be down based on the interfaces down time, and the amount of time an initial connection to the interface takes before this test is considered to have failed.
  - (Optional) Set the amount of time that the interface can be down before this test is considered to have failed:

```
(config network interface my_wwan ipv4 surelink target 0)>
interface_down_time value
(config network interface my_wwan ipv4 surelink target 0)>
```

The default is 60 seconds.

- (Optional) Set the amount of time to wait for an initial connection to the interface before this test is considered to have failed:

```
(config network interface my_wwan ipv4 surelink target 0)>
interface_timeout value
(config network interface my_wwan ipv4 surelink target 0)>
```

The default is 60 seconds.

- **other**: Allows you to test another interface's status, to create a failover or coupled relationship between interfaces:

```
(config network interface my_wwan ipv4 surelink target 0)> other
value
(config network interface my_wwan ipv4 surelink target 0)>
```

If **other** is set:

- Set the alternate interface to be tested:
  i. Use the **?** to determine available interfaces:
  ii. Set the interface. For example:

```
(config network interface my_wan ipv4 surelink target
0)> other_interface /network/interface/eth1
(config network interface my_wan ipv4 surelink target
0)>
```

- Set the alternate interface's IP version. This allows you to determine the alternate interface's status for a particular IP version.

```
(config network interface my_wwan ipv4 surelink target 0)>
other_ip_version value
(config network interface my_wwan ipv4 surelink target 0)>
```

where *value* is one of: **any**, **both**, **ipv4**, or **ipv6**.

- Set the expected status of the alternate interface:

```
(config network interface my_wwan ipv4 surelink target 0)>
other_status value
(config network interface my_wwan ipv4 surelink target 0)>
```

where *value* is either **up** or **down**. For example, if **other_status** is set to **down**, but the alternate interface is determined to be up, then this test will fail.

(Optional) Repeat to add additional test targets.

9. Optional active recovery configuration parameters:

a. Move back two levels in the configuration by typing **.. ..**:

```
(config network interface my_wwan ipv4 surelink target 0)> .. ..
(config network interface my_wwan ipv4 surelink>
```

b. Set the **Interval** between connectivity tests:

```
(config network interface my_wwan ipv4 surelink)> interval value
(config network interface my_wwan ipv4 surelink>
```

The default is 15 minutes.

c. If more than one test target is configured, determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets:

```
(config network interface my_wwan ipv4 surelink)> success_condition
value
(config network interface my_wwan ipv4 surelink>
```

Where *value* is either **one** or **all**.

d. Set the number of probe attempts before the WAN is considered to have failed:

```
(config network interface my_wwan ipv4 surelink)> attempts num
(config network interface my_wwan ipv4 surelink>
```

The default is **3**.

e. Set the amount of time that the device should wait for a response to a probe attempt before considering it to have failed:

```
(config network interface my_wwan ipv4 surelink)> timeout value
(config network interface my_wwan ipv4 surelink>
```

The default is 15 seconds.

10. (Optional) Repeat this procedure for IPv6.
11. Save the configuration and apply the change:

```
(config network interface my_wwan ipv4 surelink)> save
Configuration saved.
>
```

12. Type **exit** to exit the Admin CLI.

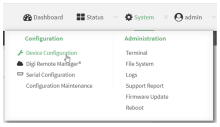
    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Disable SureLink

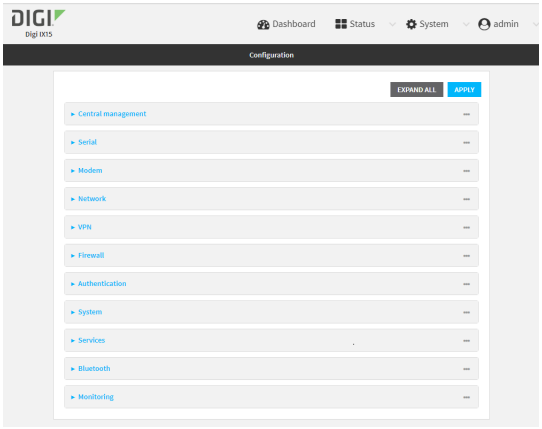
If your device uses a private APN with no Internet access, or your device has a restricted wired WAN connection that doesn't allow DNS resolution, follow this procedure to disable the default SureLink connectivity tests. You can also disable DNS lookup or other internet activity, while retaining the SureLink interface test.

WebUI

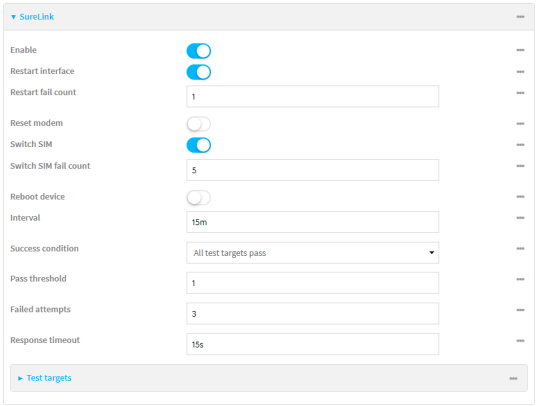
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



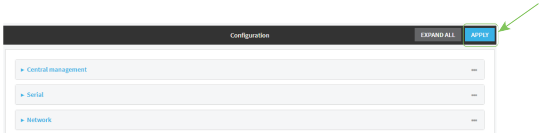
The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. Select the appropriate WAN or WWAN on which SureLink should be disabled..
5. After selecting the WAN or WWAN, click **IPv4** > **SureLink**.



6. Toggle off **Enable** to disable SureLink.
7. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Change to the WAN or WWAN's node in the configuration schema. For example, to disable SureLink for the Modem interface:

```
(config)> network interface modem
(config network interface modem)>
```

4. Disable SureLink:

```
(config network interface modem> ipv4 surelink enable false
(config network interface modem)>
```

5. Save the configuration and apply the change:

```
(config network interface my_wwan ipv4 surelink)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

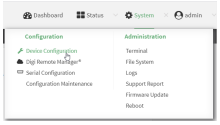
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Disable DNS lookup

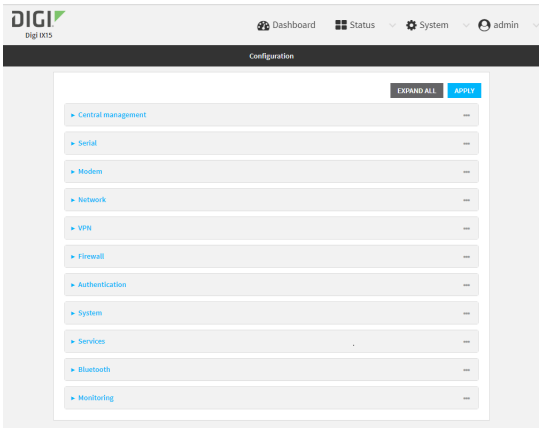
Alternatively, you can disable DNS lookup or other internet activity for device that use a private APN with no Internet access, or that have restricted wired WAN connections that do not allow DNS resolution, while retaining the SureLink interface test. The SureLink interface test determines if the interface has an IP address assigned to it, that the physical link is up, and that a route is present to send traffic out of the network interface.

WebUI

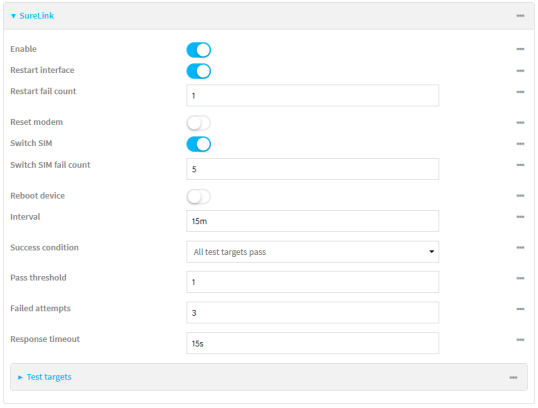
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



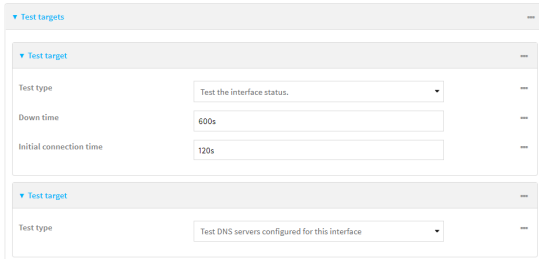
   The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. Select the appropriate WAN or WWAN on which SureLink should be disabled..
5. After selecting the WAN or WWAN, click **IPv4** > **SureLink**.



6. Click to expand **Test targets**.
7. Click to expand the second test target. This test target has its **Test type** set to **Test DNS servers configured for this interface**.



8. Click the menu icon **(...)** next to the target and select **Delete**.



9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Change to WAN or WWAN's node in the configuration schema. For example, to disable SureLink for an interface named my_wan:

```
(config)> network interface my_wan
(config network interface my_wan)>
```

4. Determine the index number of the target:

```
(config network interface my_wan)> show ipv4 surelink target
0
        interface_down_time 600s
        interface_timeout 120s
        test interface_up
1
        test dns_configured
(config network interface my_wan)>
```

5. Delete the target:

```
(config network interface my_wan> del ipv4 surelink target 1
(config network interface my_wan)>
```

6. Save the configuration and apply the change:

```
(config network interface my_wan ipv4 surelink)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Using cellular modems in a Wireless WAN (WWAN)

The IX15 supports one cellular modem, named **Modem**, which is included in a preconfigured Wireless WAN, also named **Modem**.

The cellular modem can have only one active SIM slot at any one time. For example, **Modem** can have either SIM1 or SIM2 up at one time.

Typically, you configure SIM1 of the cellular modem as the primary cellular interface, and SIM2 as the backup cellular interface. In this way, if the IX15 device cannot connect to the network using SIM1, it automatically fails over to SIM2. IX15 devices automatically use the correct cellular module firmware for each carrier when switching SIMs.

Configure cellular modem

Configuring the IX15's cellular modem involves configuring the following items:

Required configuration items

- Enable the cellular modem.

  The cellular modem is enabled by default.
- Configure the criteria used to determine which modem this modem configuration applies to.
- Determine the SIM slot that will be used when connecting to the cellular network.
- Configure the maximum number of interfaces that can use the modem.
- Enable carrier switching, which allows the modem to automatically match the carrier for the active SIM.

  Carrier switching is enabled by default.
- Configure the access technology.
- Determine which cellular antennas to use.

Additional configuration items

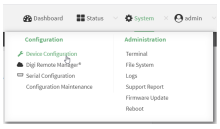
- If **Active SIM slot** is set to **Any**, determine the preferred SIM slot.

  In the event of a failover to a non-preferred SIM, or if manual SIM switching is used to switch to a non-preferred SIM, the modem will attempt to reconnect to the SIM in the preferred SIM slot.

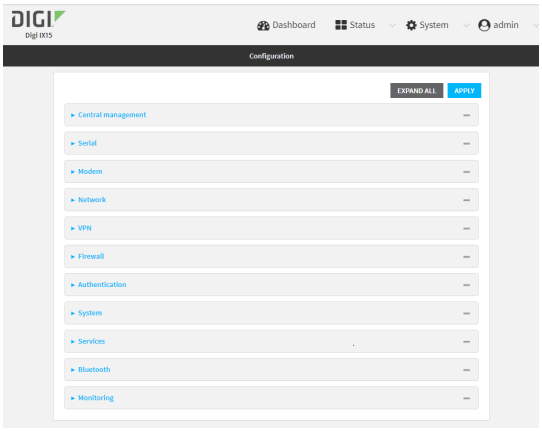
To configure the modem:

WebUI

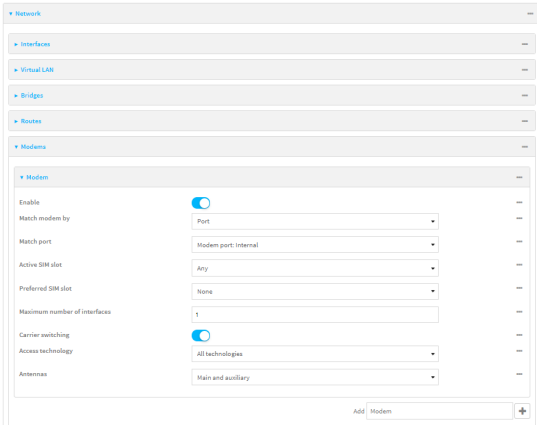
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

   The **Configuration** window is displayed.



3. Click **Network** > **Modems** > **Modem**.



4. Modem are enabled by default. Click to toggle **Enable** to off to disable.
5. For **Match modem by**, select the matching criteria used to determine if this modem configuration applies to the currently attached modem:
   - **Any modem**: Applies this configuration to any modem that is attached.
   - **IMEI**: Applies this configuration only to a modem that matches the identified IMEI.
     - If **IMEI** is selected, for **Match IMEI**, type the IMEI of the modem that this configuration should be applied to.
   - **Port**: Applies this configuration to a modem attached to the identified physical port.
     - If **Port** is selected, for **Match Port**, select the modem's port.

   The default is **Any modem**.
6. For **Active SIM slot**, select the SIM slot that should be used by the modem, or select **Any** to use any SIM slot. The default is **Any**.
7. If **Active SIM slot** is set to **Any**, for **Preferred SIM slot**, select the SIM slot that should be considered the preferred slot for this modem, or select **None**. In the event of a failover to a non-preferred SIM, or if manual SIM switching is used to switch to a non-preferred SIM, the modem will attempt to reconnect to the SIM in the preferred SIM slot. **None** is the default.

8. For Maximum number of interfaces, type the number of interfaces that can be configured to use this modem. This is used when using dual-APN SIMs. The default is **1**.
9. Enable **Carrier switching** to allow the modem to automatically match the carrier for the active SIM. **Carrier switching** is enabled by default.
10. For **Access technology**, select the type of cellular technology that this modem should use to access the cellular network, or select **All technologies** to configure the modem to use the best available technology. The default is **All technologies**.
11. For **Antennas**, select whether the modem should use the main antenna, the auxiliary antenna, or both the main and auxiliary antennas.
12. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Modem configurations are enabled by default. To disable:

```
(config)> network modem modem enable false
(config)>
```

4. Set the matching criteria used to determine if this modem configuration applies to the currently attached modem:

```
(config)> network modem modem match value
(config)>
```

   where *value* is one of the following:

   - **any**: Applies this configuration to any modem that is attached.
   - **imei**: Applies this configuration only to a modem that matches the identified IMEI.
     - If **imei** is used, set the IMEI of the modem that this configuration should be applied to:

```
(config)> network modem modem imei value
(config)>
```

       where *value* is the IMEI of the modem.

- **port**: Applies this configuration to a modem attached to the identified physical port.
  - If **port** is used, set modem's port:
    a. Determine available ports and correct syntax by using the **?**:

```
(config)> network modem modem port ?

Match port: The physical port that the modem device is
attached to.
Format:
    /device/usb/modem/module
Default value: /device/usb/modem/module
Current value: /device/usb/modem/module

(config)> network modem modem port
```

    b. Set the port:

```
(config)> network modem modem port /device/usb/modem/module
(config)>
```

  The default is **any**.

5. Set the SIM slot that should be used by the modem:

```
(config)> network modem modem sim_slot value
(config)>
```

   where *value* is one of the following:
   - **any**: Uses either SIM slot.
   - **1**: Uses the first SIM slot.
   - **2**. Uses the second SIM slot.

   The default is **any**.

6. If **sim_slot** is set to **any**, set the SIM slot that should be considered the preferred slot for this modem:

```
(config)> network modem modem sim_slot_preference value
(config)>
```

   where *value* is one of the following:
   - **none**: Does not consider either SIM slot to be the preferred slot.
   - **1**: Configures the first SIM slot as the preferred SIM slot.
   - **2**. Configures the second SIM slot as the preferred SIM slot.

   In the event of a failover to a non-preferred SIM, or if manual SIM switching is used to switch to a non-preferred SIM, the modem will attempt to reconnect to the SIM in the preferred SIM slot. The default is **none**.

7. Set the maximum number of interfaces. This is used when using dual-APN SIMs. The default is **1**.

```
(config)> network modem modem max_intfs int
(config)>
```

8. Carrier switching allows the modem to automatically match the carrier for the active SIM. **Carrier switching** is enabled by default. To disable:

```
(config)> network modem modem carrier_switch false
(config)>
```

9. Set the type of cellular technology that this modem should use to access the cellular network:

```
(config)> network modem modem access_tech value
(config)>
```

Available options for *value* vary depending on the modem type. To determine available options:

```
(config)> network modem modem access_tech ?

Access technology: The cellular network technology that the modem may
use.
Format:
    2G
    3G
    4G
    4GM
    4GT
    all
Default value: all
Current value: all

(config)>
```

The default is **all**, which uses the best available technology.

10. Set whether the modem should use the main antenna, the auxiliary antenna, or both the main and auxiliary antennas:

```
(config)> network modem modem antenna value
(config)>
```

where *value* is one of the following:

- **main**
- **aux**
- **both**

11. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

12. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

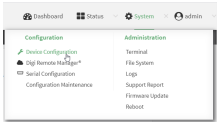
Configure cellular modem APNs

The IX15 device uses a preconfigured list of Access Point Names (APNs) when attempting to connect to a cellular carrier for the first time. After the device has successfully connected, it will remember the correct APN. As a result, it is generally not necessary to configure APNs. However, you can configure the system to use a specified APN.

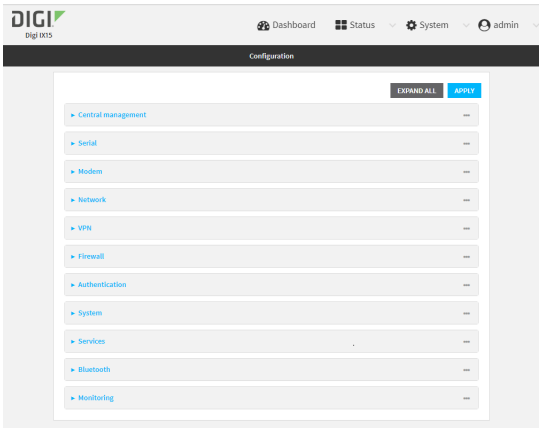
To configure the APN:

WebUI

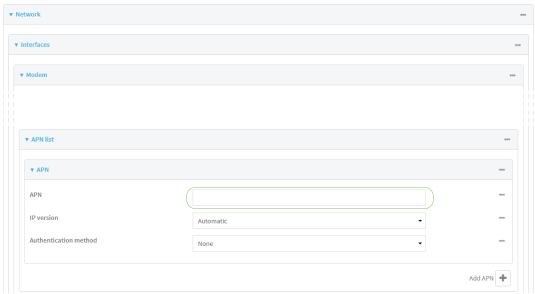
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **Network** > **Interfaces** > **Modem** > **APN list** > **APN**.



4. For **APN**, type the Access Point Name (APN) to be used when connecting to the cellular carrier.
5. (Optional) **IP version**:

For **IP version**, select one of the following:

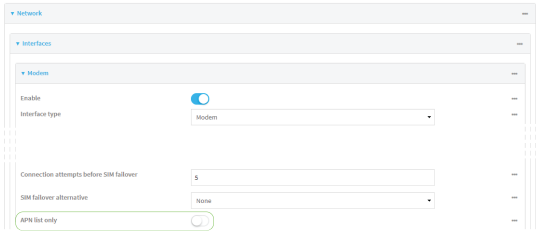
- **Automatic**: Requests both IPv4 and IPv6 address.
- **IPv4**: Requests only an IPv4 address.
- **IPv6**: Requests only an IPv6 address.

The default is **Automatic**.

6. (Optional) For **Authentication method**, select one of the following:
   - **None**: No authentication is required.
   - **Automatic**: The device will attempt to connect using CHAP first, and then PAP.
   - **CHAP**: Uses the Challenge Handshake Authentication Profile (CHAP) to authenticate.
   - **PAP**: Uses the Password Authentication Profile (PAP) to authenticate.

   If **Automatic**, **CHAP**, or **PAP** is selected, enter the **Username** and **Password** required to authenticate.

   The default is **None**.
7. To add additional APNs, for **Add APN**, click **+** and repeat the preceding instructions.
8. (Optional) To configure the device to bypass its preconfigured APN list and only use the configured APNs, enable **APN list only**.



9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type:

```
(config)> network interface modem modem apn 0 apn value
(config)>
```

   where *value* is the APN for the SIM card.

4. (Optional) To add additional APNs:
   a. Use the **add** command to add a new APN entry. For example:

```
(config)> add network interface modem modem apn end
(config network interface modem modem apn 1)>
```

   b. Set the value of the APN:

```
(config network interface modem modem apn 1)> apn value
(config network interface modem modem apn 1)>
```

   where value is the APN for the SIM card.
5. (Optional) Set the IP version:

```
(config)> network interface modem modem apn 0 ip_version version
(config)>
```

   where *version* is one of the following:
   - **auto**: Requests both IPv4 and IPv6 address.
   - **ipv4**: Requests only an IPv4 address.
   - **ipv6**: Requests only an IPv6 address.

   The default is **auto**.
6. (Optional) Set the authentication method:

```
(config)> network interface modem modem apn 0 auth method
(config)>
```

   where *method* is one of the following:
   - **none**: No authentication is required.
   - **auto**: The device will attempt to connect using CHAP first, and then PAP.
   - **chap**: Uses the Challenge Handshake Authentication Profile (CHAP) to authenticate.
   - **pap**: Uses the Password Authentication Profile (PAP) to authenticate.

   If **auto**, **chap**, or **pap** is selected, enter the **Username** and **Password** required to authenticate:

```
(config)> network interface modem modem apn 0 username name
(config)> network interface modem modem apn 0 password pwd
(config)>
```

   The default is **none**.
7. (Optional) To configure the device to bypass its preconfigured APN list and only use the configured APNs:

```
(config)> network interface modem modem apn_lock true
(config)>
```

8. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show cellular status and statistics

You can view a summary status for all cellular modems, or view detailed status and statistics for a specific modem.

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **Status**.
3. Under **Connections**, click **Modems**.

   The modem status window is displayed

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Use the show modem command:
   - To view a status summary for the modem:

```
> show modem

Modem  SIM            Status     APN        Signal Strength
-----  -------------  ---------  ---------  --------------------
modem  1 (ready)      connected  1234       Good (-84 dBm)

>
```

   - To view detailed status and statistics, use the show modem name *name* command:

```
> show modem name modem

 modem: [Telit] LM940
 ------------------------------------------------------------------------
-----------
 IMEI                    : 781154796325698
 Model                   : LM940
 FW Version              : 24.01.541_ATT
 Revision                : 24.01.541

 Status
 ------
 State                   : connected
```

```
Signal Strength          : Good (-85 dBm)
Bars                     : 2/5
Access Mode              : 4G
Network Technology (CNTI): LTE
Band                     : B2
Temperature              : 34C

wwan1 Interface
---------------
APN                      : 1234
IPv4 surelink            : passing
IPv4 address             : 189.232.229.47
IPv4 gateway             : 189.232.229.1
IPv4 MTU                 : 1500
IPv4 DNS server(s)       : 245.144.162.207, 245.144.162.208

IPv6 surelink            : passing
IPv6 address             : 11f6:4680:0d67:59d2:552b:3429:81a8:f1ea
IPv6 gateway             : ff50:d95d:7e98:abe8:3030:9138:4f25:f51b
IPv6 MTU                 : 1500

TX bytes                 : 127941
RX bytes                 : 61026
Uptime                   : 10 hrs, 56 mins (39360s)

SIM
---
SIM Slot                 : 1
SIM Status               : ready
IMSI                     : 61582122197895
ICCID                    : 26587628655003992180
SIM Provider             : AT&T

4G
--
RSRQ                     : Good (-11.0 dB)
RSRP                     : Good (-93.0 dBm)
RSSI                     : Excellent (-64.0 dBm)
SNR                      : Good (6.4 dB)

>
```

Unlock a SIM card

A SIM card can be locked if a user tries to set an invalid PIN for the SIM card too many times. In addition, some cellular carriers require a SIM PIN to be added before the SIM card can be used. If the SIM card is locked, the Digi IX15 Gateway device cannot make a cellular connection.

Command line

To unlock a SIM card:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the Admin CLI prompt, use the modem puk unlock command to set a new PIN for the SIM card:

   ```
   > modem puk unlock puk_code new_pin modem_name
   >
   ```

   For example, to unlock a SIM card in the modem named **modem** with PUK code **12345678**, and set the new SIM PIN to **1234**:

   ```
   > modem puk unlock 12345678 1234 modem
   >
   ```
3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Note If the SIM remains in a locked state after using the unlock command, contact your cellular carrier.

Signal strength for 4G cellular connections

For 4G connections, the **RSRP** value determines signal strength.

- **Excellent**: > **-90 dBm**
- **Good**: **-90 dBm** to **-105 dBm**
- **Fair**: **-106 dBm** to **-115 dBm**
- **Poor**: **-116 dBm** to **-120 dBm**:
- **No service**: **< -120 dBm**

See Show cellular status and statistics for procedures to view this information.

Signal strength for 3G and 2G cellular connections

For 3G and 2G cellular connections, the current **RSSI** value determines signal strength.

- **Excellent**: > **-70 dBm**
- **Good**: **-70 dBm** to **-85 dBm**
- **Fair**: **-86 dBm** to **-100 dBm**
- **Poor**: **< -100 dBm** to **-109 dBm**
- **No service**: **-110 dBm**

See Show cellular status and statistics for procedures to view this information.

Tips for improving cellular signal strength

If the signal strength LEDs or the signal quality for your device indicate **Poor** or **No service**, try the following things to improve signal strength:

- Move the IX15 device to another location.
- Try connecting a different set of antennas, if available.
- Purchase a Digi Antenna Extender Kit:
  - Antenna Extender Kit, 1m
  - Antenna Extender Kit, 3m

AT command access

To run AT commands from the IX15 command line:

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the Admin CLI prompt, type **modem at-interactive** and press **Enter**. Type **n** if you do not want exclusive access. This allows you to send AT commands to the device while still allowing the device to connect, disconnect, and/or reconnect to the cellular network.
3. At the Admin CLI prompt, use the modem at-interactive command to begin an interactive AT command session:

```
> modem at-interactive


Do you want exclusive access to the modem?  (y/n) [y]:
```

4. Type **n** if you do not want exclusive access. This allows you to send AT commands to the device while still allowing the device to connect, disconnect, and/or reconnect to the cellular network.

   The following is an example interactive AT command:

```
> modem at-interactive

Do you want exclusive access to the modem? (y/n) [y]: n
Starting terminal access to modem AT commands.
Note that the modem is still in operation.

To quit enter '~.' ('~~.' if using an ssh client) and press ENTER

Connected
ati
Manufacturer: Sierra Wireless, Incorporated
Model: MC7455
Revision: SWI9X30C_02.24.03.00 r6978 CARMD-EV-FRMWR2 2017/03/02 13:36:45
MEID: 35907206045169
IMEI: 359072060451693
IMEI SV: 9
FSN: LQ650551070110
```

```
+GCAP: +CGSM
OK
```

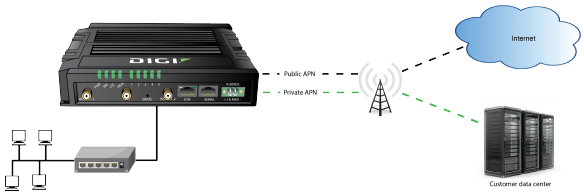
5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure dual APNs

Some cellular carriers offer a dual APN feature that allows a SIM card to be provisioned with two separate APNs that can be used simultaneously. For example, Verizon offers this service as its Split Data Routing feature. This feature provides two separate networking paths through a single cellular modem and SIM card, and allows for configurations such as:

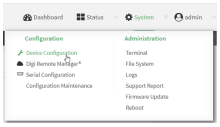
- Segregating public and private traffic, including policy-based routes to ensure that your internal network traffic always goes through the private connection.
- Separation of untrusted Internet traffic from trusted internal network traffic.
- Secure connection to internal customer network without using a VPN.
- Separate billing structures for public and private traffic.
- Site-to-site networking, without the overhead of tunneling for each device.



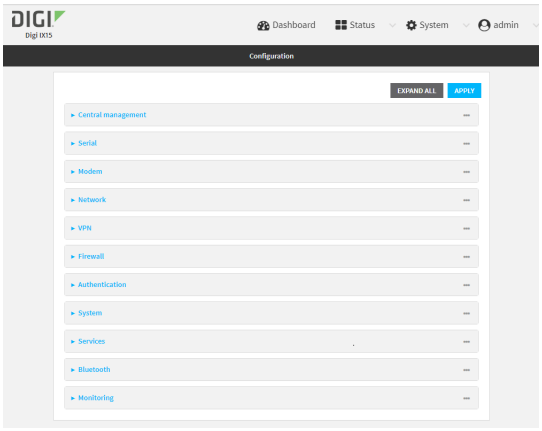
To accomplish this, we will create separate WWAN interfaces that use the same modem but use different APNs, and then use routing roles to forward traffic to the appropriate WWAN interface.

WebUI

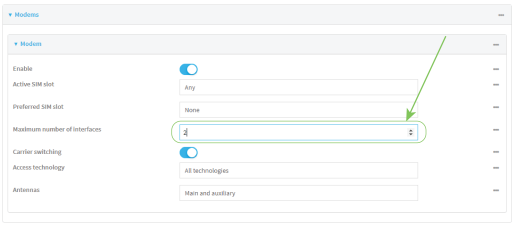
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.

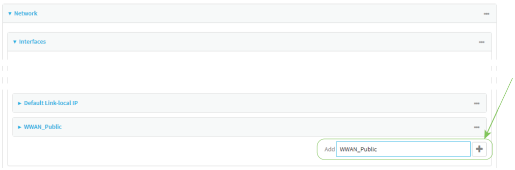


3. Increase the maximum number of interfaces allowed for the modem:
   a. Click **Network** > **Modems** > **Modem**.
   b. For **Maximum number of interfaces**, type **2**.

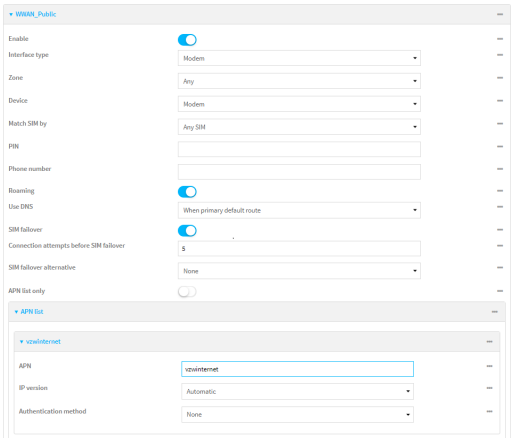


4. Create the WWAN interfaces:

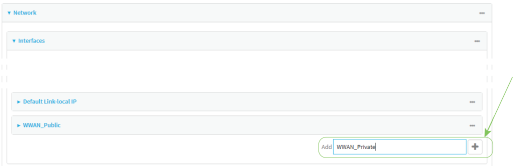
   In this example, we will create two interfaces named **WWAN_Public** and **WWAN_Private**.
   a. Click **Network** > **Interfaces**.
   b. For **Add Interface**, type **WWAN_Public** and click ✚.



   c. For **Interface type**, select **Modem**.
   d. For **Zone**, select **External**.
   e. For **Device**, select **Modem** .
   f. (Optional): Configure the public APN. If the public APN is not configured, the IX15 will attempt to determine the APN.
      i. Click to expand **APN list** > **APN**.
      ii. For **APN**, type the public APN for your cellular carrier.



g. For **Add Interface**, type **WWAN_Private** and click ✚.



h. For **Interface type**, select **Modem**.

i. For **Zone**, select **External**.

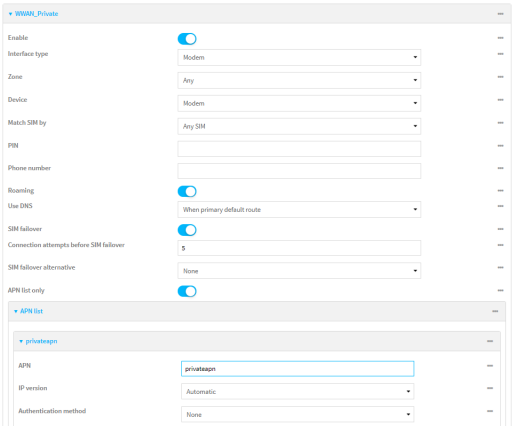
j. For **Device**, select **Modem** .

This should be the same modem selected for the **WWAN_Public** WWAN.

k. Enable **APN list only**.

l. Click to expand **APN list** > **APN**.

m. For **APN**, type the private APN provided to you by your cellular carrier.



5. Create the routing policies. For example, to route all traffic from a device with the IP address of 192.168.2.101 through the private APN:

a. Click **Network** > **Routes** > **Policy-based routing**.

b. Click the ✚ to add a new route policy.



c. For **Label**, enter **Route through private APN**.

d. For **Interface**, select **Interface: WWAN_Private**.

e. Configure the source address:

i. Click to expand **Source address**.

ii. For **Type**, select **IPv4 address**.

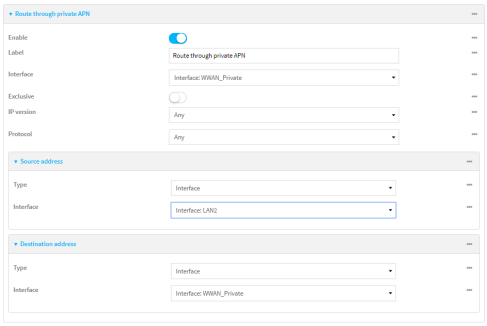
iii. For **Address**, type **192.168.2.101**.

f. Configure the destination address:

i. Click to expand **Destination address**.

ii. For **Type**, select **Interface**.

iii. For **Interface**, select **Interface: WWAN_Private**.



6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Set the maximum number of interfaces for the modem:

```
(config)> network modem modem max_intfs 2
(config)>
```

4. Create the WWAN interfaces:
   a. Create the **WWANPublic** interface:

```
(config)> add network interface WWANPublic
(config network interface WWANPublic)>
```

   b. Set the interface type to modem:

```
(config network interface WWANPublic)> type modem
(config network interface WWANPublic)>
```

   c. Set the modem device:

```
(config network interface WWANPublic)> modem device modem
(config network interface WWANPublic)>
```

   d. (Optional): Set the public APN. If the public APN is not configured, the IX15 will attempt to determine the APN.

```
(config network interface WWANPublic)> modem apn public_apn
(config network interface WWANPublic)>
```

   e. Use to periods (**..**) to move back one level in the configuration:

```
(config network interface WWANPublic)> ..
(config network interface)>
```

   f. Create the **WWANPrivate** interface:

```
(config network interface)> add WWANPrivate
(config network interface WWANPrivate)>
```

   g. Set the interface type to modem:

```
(config network interface WWANPrivate)> type modem
(config network interface WWANPrivate)>
```

   h. Set the modem device:

```
(config network interface WWANPrivate)> modem device modem
(config network interface WWANPrivate)>
```

   i. Enable **APN list only**:

```
(config network interface WWANPrivate)> apn_lock true
(config network interface WWANPrivate)>
```

j. Set the private APN:

```
(config network interface WWANPublic)> modem apn private_apn
(config network interface WWANPublic)>
```

5. Create the routing policies. For example, to route all traffic from a device with the IP address of 192.168.2.101 through the private APN:
   a. Add a new routing policy:

```
(config)> add network route policy end
(config network route policy 0)>
```

   b. Set the label that will be used to identify this route policy:

```
(config network route policy 0)> label "Route through private apn"
(config network route policy 0)>
```

   c. Set the interface:

```
(config network route policy 0)> interface
/network/interface/WWANPrivate
(config network route policy 0)>
```

   d. Configure the source address:
      i. Set the source type to **address**:

```
(config network route policy 0)> src type address
(config network route policy 0)>
```

      ii. Set the IP address to **192.168.2.101**:

```
(config network route policy 0)> src address 192.168.2.101
(config network route policy 0)>
```

   e. Configure the destination address:
      i. Set the type to **interface**:

```
(config network route policy 1)> dst type interface
(config network route policy 1)>
```

      ii. Set the interface to **WWANPrivate** :

```
(config network route policy 1)> interface
/network/interface/WWANPrivate
(config network route policy 1)>
```

6. Save the configuration and apply the change:

```
(config network route policy 1)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure a Wireless Wide Area Network (WWAN)

Configuring a Wireless Wide Area Network (WWAN) involves configuring the following items:

Required configuration items

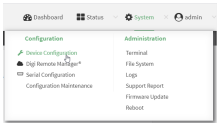
- The interface type: **Modem**.
- The firewall zone: **External**.
- The cellular modem that is used by the WWAN.

Additional configuration items

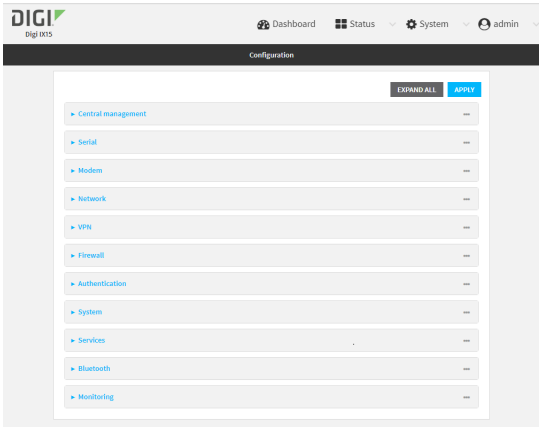
- SIM selection for this WWAN.
- The SIM PIN.
- The SIM phone number for SMS connections.
- Enable or disable roaming.
- SIM failover configuration.
- APN configuration.
- The custom gateway/netmask.
- IPv4 configuration:
  - The metric for IPv4 routes associated with the WAN.
  - The relative weight for IPv4 routes associated with the WAN.
  - The IPv4 management priority of the WAN. The active interface with the highest management priority will have its address reported as the preferred contact address for central management and direct device access.
  - The IPv4 Maximum Transmission Unit (MTU) of the WAN.
  - When to use DNS: always, never, or only when this interface is the primary default route.
  - SureLink active recovery configuration. See Configure SureLink active recovery to detect modem failures for further information.
- IPv6 configuration:
  - The metric for IPv6 routes associated with the WAN.
  - The relative weight for IPv6 routes associated with the WAN.
  - The IPv6 management priority of the WAN. The active interface with the highest management priority will have its address reported as the preferred contact address for central management and direct device access.
  - The IPv6 Maximum Transmission Unit (MTU) of the WAN.
  - When to use DNS: always, never, or only when this interface is the primary default route.
  - SureLink active recovery configuration. See Configure SureLink active recovery to detect modem failures for further information.

WebUI

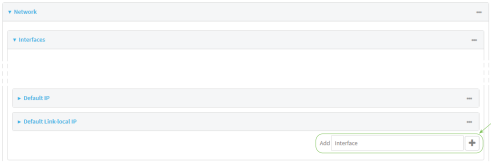
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.

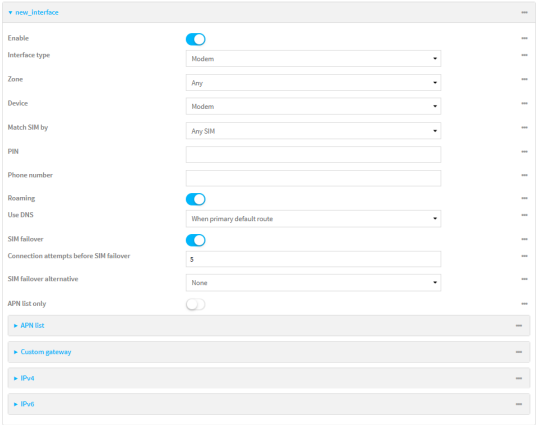


3. Click **Network** > **Interfaces**.
4. Create the WWAN or select an existing WWAN:
   - To create a new WWAN, for **Add interface**, type a name for the WWAN and click ✚.



   - To edit an existing WWAN, click to expand the WWAN.

     New WWANs are enabled by default. To disable, click **Enable**.
5. For **Interface type**, select **Modem**.



6. The WWAN is enabled by default. Click **Enable** to disable, or to enable if it has been disabled.
7. **Interface type** defaults to **Modem**.
8. For **Zone**, select **External**.
9. For **Device**, select the cellular modem.
10. For **Match SIM by**, select a SIM matching criteria to determine when this WWAN should be used:
    - If **SIM slot** is selected, for **Match SIM slot**, select which SIM slot must be in active for this WWAN to be used.
    - If **Carrier** is selected, for **Match SIM carrier**, select which cellular carrier must be in active for this WWAN to be used.
    - If **PLMN identifier** is selected, for **Match PLMN identifier**, type the PLMN id that must be in active for this WWAN to be used.
    - If **IMSI** is selected, for **Match IMSI**, type the International Mobile Subscriber Identity (IMSI) that must be in active for this WWAN to be used.
    - If **ICCID** is selected, for **Match ICCID**, type the unique SIM card ICCID that must be in active for this WWAN to be used.
11. Type the **PIN** for the SIM. Leave blank if no PIN is required.
12. Type the **Phone number** for the SIM, for SMS connections.

    Normally, this should be left blank. It is only necessary to complete this field if the SIM does not have a phone number or if the phone number is incorrect.
13. **Roaming** is enabled by default. Click to disable.
14. For **Carrier selection mode**, select one of the following:
    - **Automatic**: The cellular carrier is selected automatically by the device.
    - **Manual**: The cellular carrier must be manually configured. If the configured network is not available, no cellular connection will be established.
    - **Manual/Automatic**: The carrier is manually configured. If the configured network is not available, automatic carrier selection is used.

    If **Manual** or **Manual/Automatic** is selected:

    a. For **Network PLMN ID**, type the PLMN ID for the cellular network.

    b. For **Network technology**, select the technology that should be used. The default is **All technologies**, which means that the best available technology will be used.

    Note If **Manual** is configured for **Carrier selection mode** and a specific network technology is selected for the **Network technology**, your modem must support the selected technology or no cellular connection will be established. If you are using a cellular connection to perform this procedure, you may lose your connection and the device will no longer be accessible.
15. **SIM failover** is enabled by default, which means that the modem will automatically fail over from the active SIM to the next available SIM when the active SIM fails to connect. If enabled:

    a. For **Connection attempts before SIM failover**, type the number of times that the device should attempt to connect to the active SIM before failing over to the next available SIM.

    b. For **SIM failover alternative**, configure how SIM failover will function if automatic SIM switching is unavailable:

- **None**: The device will perform no alternative action if automatic SIM switching is unavailable.
- **Reset modem**: The device will reset the modem if automatic SIM switching is unavailable.
- **Reboot device**: The device will reboot if automatic SIM switching is unavailable.

16. For **APN list** and **APN list only**, the IX15 device uses a preconfigured list of Access Point Names (APNs) when attempting to connect to a cellular carrier for the first time. After the device has successfully connected, it will remember the correct APN. As a result, it is generally not necessary to configure APNs. See Configure cellular modem APNs for further information and instructions for setting an APN.
17. (Optional) To configure the IP address of a custom gateway or a custom netmask:
    a. Click **Custom gateway** to expand.
    b. Click **Enable**.
    c. For **Gateway/Netmask**, enter the IP address and netmask of the custom gateway. To override only the gateway netmask, but not the gateway IP address, use all zeros for the IP address. For example, **0.0.0.0./32** will use the network-provided gateway, but with a /32 netmask.
18. Optional IPv4 configuration items:
    a. Click **IPv4** to expand.
    b. IPv4 support is **Enabled** by default. Click to disable.
    c. Set the **MTU**.
    d. For **Use DNS**:
        - **Always**: DNS will always be used for this WWAN; when multiple interfaces have the same DNS server, the interface with the lowest metric will be used for DNS requests.
        - **When primary default route**: Only use the DNS servers provided for this WWAN when the WWAN is the primary route.
        - **Never**: Never use DNS servers for this WWAN.

    The default setting is **When primary default route**.
19. Optional IPv6 configuration items:
    a. Click **IPv6** to expand.
    b. IPv6 support is **Enabled** by default. Click to disable.
    c. Set the **MTU**.
    d. For **Use DNS**:
        - **Always**: DNS will always be used for this WWAN; when multiple interfaces have the same DNS server, the interface with the lowest metric will be used for DNS requests.
        - **When primary default route**: Only use the DNS servers provided for this WWAN when the WWAN is the primary route.
        - **Never**: Never use DNS servers for this WWAN.

    The default setting is **When primary default route**.

1. See Configure SureLink active recovery to detect modem failures for information about configuring **SureLink**.

Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Create a new WWAN or edit an existing one:
   - To create a new WWAN named **my_wwan**:

```
(config)> add network interface my_wwan
(config network interface my_wwan)>
```

   - To edit an existing WWAN named **my_wwan**, change to the my_wwan node in the configuration schema:

```
(config)> network interface my_wwan
(config network interface my_wwan)>
```

4. Set the appropriate firewall zone:

```
(config network interface my_wwan)> zone zone
(config network interface my_wwan)>
```

   See Firewall configuration for further information.
5. Select a cellular modem:
   a. Enter **modem device ?** to view available modems and the proper syntax.

```
(config network interface my_wwan)> modem device ?

Device: The modem used by this network interface.
Format:
  modem
Current value:

(config network interface my_wwan)> device
```

   b. Set the device:

```
(config network interface my_wwan)> modem device modem
(config network interface my_wwan)>
```

6. Set theSIM matching criteria to determine when this WWAN should be used:

```
(config network interface my_wwan)> modem match value
(config network interface my_wwan)>
```

Where *value* is one of:

- **any**
- **carrier**

  Set the cellular carrier must be in active for this WWAN to be used:

  a. Use **?** to determine available carriers:

```
(config network interface my_wwan)> modem carrier

Match SIM carrier: The SIM carrier match criteria. This
interface is applied when the SIM card is
provisioned from the carrier.
Format:
  AT&T
  Rogers
  Sprint
  T-Mobile
  Telstra
  Verizon
  Vodafone
  other
Default value: AT&T
Current value: AT&T

(config network interface my_wwan)>
```

  b. Set the carrier:

```
(config network interface my_wwan)> modem carrier value
(config network interface my_wwan)>
```

- **iccid**

  Set the unique SIM card ICCID that must be in active for this WWAN to be used:

```
(config network interface my_wwan)> modem iccid ICCID
(config network interface my_wwan)>
```

- **imsi**

  Set the International Mobile Subscriber Identity (IMSI) that must be in active for this WWAN to be used:

```
(config network interface my_wwan)> modem imsi IMSI
(config network interface my_wwan)>
```

- **plmn_id**

  Set the PLMN id that must be in active for this WWAN to be used:

```
(config network interface my_wwan)> modem plmn_id PLMN_ID
(config network interface my_wwan)>
```

- **sim_slot**

  Set which SIM slot must be in active for this WWAN to be used:

  ```
  (config network interface my_wwan)> modem sim_slot value
  (config network interface my_wwan)>
  ```

  where *value* is either **1** or **2**.

7. Set the PIN for the SIM. Leave blank if no PIN is required.

   ```
   (config network interface my_wwan)> modem pin value
   (config network interface my_wwan)>
   ```

8. Set the phone number for the SIM, for SMS connections:

   ```
   (config network interface my_wwan)> modem phone num
   (config network interface my_wwan)>
   ```

   Normally, this should be left blank. It is only necessary to complete this field if the SIM does not have a phone number or if the phone number is incorrect.

9. Roaming is enabled by default. To disable:

   ```
   (config network interface my_wwan)> modem roaming false
   (config network interface my_wwan)>
   ```

10. Set the carrier selection mode:

    ```
    (config network interface my_wwan)> modem operator_mode value
    (config network interface my_wwan)>
    ```

    where *value* is one of:

    - **automatic**: The cellular carrier is selected automatically by the device.
    - **manual**: The cellular carrier must be manually configured. If the configured network is not available, no cellular connection will be established.
    - **manual_automatic**: The carrier is manually configured. If the configured network is not available, automatic carrier selection is used.

    If **manual** or **manual_automatic** is set:

    a. Set the Network PLMN ID:

       ```
       (config network interface my_wwan)> modem operator PLMN_ID
       (config network interface my_wwan)>
       ```

    b. Set the cellular network technology:

       ```
       (config network interface my_wwan)> modem operator_technology value
       (config network interface my_wwan)>
       ```

       where *value* is one of:

       - **all**: The best available technology will be used.
       - **2G**: Only 2G technology will be used.
       - **3G**: Only 3G technology will be used.

- **4G**: Only 4G technology will be used.
- **NR5G-NSA**: Only 5G non-standalone technology will be used.
- **NR5G-SA**: Only 5G standalone technology will be used.

The default is **all**.

Note If **manual** is configured forthe carrier selection mode and a specific network technology is selected for the cellular network technology, your modem must support the selected technology or no cellular connection will be established. If you are using a cellular connection to perform this procedure, you may lose your connection and the device will no longer be accessible.

11. SIM failover is enabled by default, which means that the modem will automatically fail over from the active SIM to the next available SIM when the active SIM fails to connect. To disable:

```
(config network interface my_wwan)> modem sim_failover false
(config network interface my_wwan)>
```

If enabled:

a. Set the number of times that the device should attempt to connect to the active SIM before failing over to the next available SIM:

```
(config network interface my_wwan)> modem sim_failover_retries num
(config network interface my_wwan)>
```

The default setting is **5**.

b. Configure how SIM failover will function if automatic SIM switching is unavailable:

```
(config network interface my_wwan)> modem sim_failover_alt value
(config network interface my_wwan)>
```

where *value* is one of:

- **none**: The device will perform no alternative action if automatic SIM switching is unavailable.
- **reset**: The device will reset the modem if automatic SIM switching is unavailable.
- **reboot**: The device will reboot if automatic SIM switching is unavailable.

12. The IX15 device uses a preconfigured list of Access Point Names (APNs) when attempting to connect to a cellular carrier for the first time. After the device has successfully connected, it will remember the correct APN. As a result, it is generally not necessary to configure APNs. See Configure cellular modem APNs for further information and instructions for setting an APN.
13. (Optional) To configure the IP address of a custom gateway or a custom netmask:

a. Enable the custom gateway:

```
(config network interface my_wwan)> modem custom_gw enable true
(config network interface my_wwan)>
```

b. Set the IP address and netmask of the custom gateway:

```
(config network interface my_wwan)> modem custom_gw gateway ip_
address/netmask
(config network interface my_wwan)> modem custom_gw
```

To override only the gateway netmask, but not the gateway IP address, use all zeros for the IP address. For example, **0.0.0.0./32** will use the network-provided gateway, but with a /32 netmask.

14. Optional IPv4 configuration items:
    a. IPv4 support is enabled by default. To disable:

```
(config network interface my_wwan)> ipv4 enable false
(config network interface my_wwan)>
```

    b. Set the MTU:

```
(config network interface my_wwan)> ipv4 mtu num
(config network interface my_wwan)>
```

    c. Configure when the WWAN's DNS servers will be used:

```
(config network interface my_wwan)> ipv4  dns value
(config network interface my_wwan)>
```

    Where *value* is one of:

    - **always**: DNS will always be used for this WWAN; when multiple interfaces have the same DNS server, the interface with the lowest metric will be used for DNS requests.
    - **never**: Never use DNS servers for this WWAN.
    - **primary**: Only use the DNS servers provided for this WWAN when the WWAN is the primary route.

    The default setting is **primary**.

15. Optional IPv6 configuration items:
    a. IPv6 support is enabled by default. To disable:

```
(config network interface my_wwan)> ipv4 enable false
(config network interface my_wwan)>
```

    b. Set the MTU:

```
(config network interface my_wwan)> ipv4 mtu num
(config network interface my_wwan)>
```

    c. Configure when the WWAN's DNS servers will be used:

```
(config network interface my_wwan)> ipv4  dns value
(config network interface my_wwan)>
```

    Where *value* is one of:

- **always**: DNS will always be used for this WWAN; when multiple interfaces have the same DNS server, the interface with the lowest metric will be used for DNS requests.
- **never**: Never use DNS servers for this WWAN.
- **primary**: Only use the DNS servers provided for this WWAN when the WWAN is the primary route.

The default setting is **primary**.

d. See Configure SureLink active recovery to detect modem failures for information about configuring active recovery.

Show WWAN status and statistics

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. From the menu, click **Status**.
3. Under **Networking**, click **Interfaces**.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Enter the show network command at the Admin CLI prompt:

```
> show network

 Interface         Proto  Status   Address
 ----------------  -----  -------  ------------------------------
 defaultip         IPv4   up       192.168.210.1/24
 defaultlinklocal  IPv4   up       169.254.100.100/16
 eth1              IPv4   up       10.10.10.10/24
 eth1              IPv6   up       fe00:2404::240:f4ff:fe80:120/64
 eth              IPv4   up       192.168.2.1/24
 eth              IPv6   up       fd00:2704::1/48
 loopback          IPv4   up       127.0.0.1/8
 modem             IPv4   up       10.200.1.101/30
 modem             IPv6   down

>
```

3. Additional information can be displayed by using the show network verbose command:

```
> show network verbose

 Interface         Proto  Status   Type    Zone      Device    Metric
Weight
 ----------------  -----  -------  ------  --------  --------  ------  --
```

```
----
 defaultip         IPv4   up       static  setup     eth      10      10
 defaultlinklocal  IPv4   up       static  setup     eth      0       10
 eth1              IPv4   up       dhcp    external  eth1      1       10
 eth1              IPv6   up       dhcp    external  eth1      1       10
 eth              IPv4   up       static  internal  eth      5       10
 eth              IPv6   up       static  internal  eth      5       10
 loopback          IPv4   up       static  loopback  loopback  0       10
 modem             IPv4   up       modem   external  wwan1     3       10
 modem             IPv6   down     modem   external  wwan1     3       10

>
```

4. Enter **show network interface *name*** at the Admin CLI prompt to display additional information about a specific WAN. For example, to display information about ETH1, enter **show network interface *eth1***:

```
> show network interface eth1

 wan1 Interface Status
 ---------------------
 Device              : eth1
 Zone                : external

 IPv4 Status         : up
 IPv4 Type           : dhcp
 IPv4 Address(es)    : 10.10.10.10/24
 IPv4 Gateway        : 10.10.10.1
 IPv4 MTU            : 1500
 IPv4 Metric         : 1
 IPv4 Weight         : 10
 IPv4 DNS Server(s)  : 10.10.10.2, 10.10.10.3

 IPv6 Status         : up
 IPv6 Type           : dhcpv6
 IPv6 Address(es)    : fe00:2404::240:f4ff:fe80:120/64
 IPv6 Gateway        : ff80::234:f3ff:ff0e:4320
 IPv6 MTU            : 1500
 IPv6 Metric         : 1
 IPv6 Weight         : 10
 IPv6 DNS Server(s)  : fd00:244::1, fe80::234:f3f4:fe0e:4320

>
```

5. Type **exit** to exit the Admin CLI.

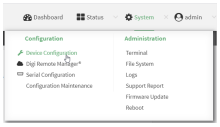
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Delete a WWAN.

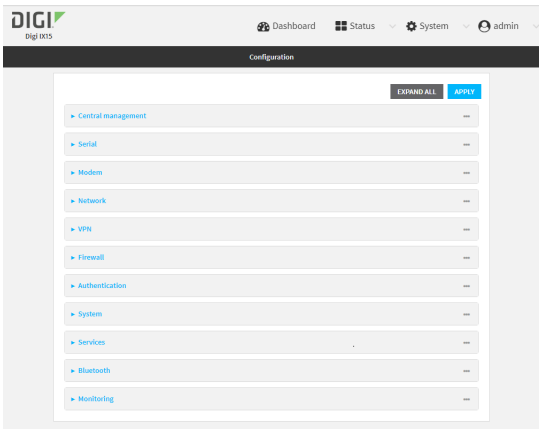
Follow this procedure to delete any WANs and WWANs that have been added to the system. You cannot delete the preconfigured WAN, **ETH1**, or the preconfigured WWAN, **Modem**.

WebUI

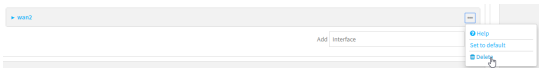
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. Click the menu icon (**...**) next to the name of the WAN or WWAN to be deleted and select **Delete**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Use the **del** command to delete the WAN or WWAN. For example, to delete a WWAN named my_wwan:

```
(config)> del network interface my_wwan
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure a Wide Area Network (WAN)

Configuring a Wide Area Network (WAN) involves configuring the following items:

Required configuration items

- The interface type: **Ethernet**.
- The firewall zone: **External**.
- The network device that is used by the WAN.
- Configure the WAN as a DHCP client.

Additional configuration items

- Additional IPv4 configuration:
  - The metric for IPv4 routes associated with the WAN.
  - The relative weight for IPv4 routes associated with the WAN.
  - The IPv4 management priority of the WAN. The active interface with the highest management priority will have its address reported as the preferred contact address for central management and direct device access.
  - The IPv4 Maximum Transmission Unit (MTU) of the WAN.
  - When to use DNS: always, never, or only when this interface is the primary default route.
  - When to use DNS servers for this interface.
  - Whether to include the IX15 device's hostname in DHCP requests.
  - SureLink active recovery configuration. See Configure SureLink active recovery to detect modem failures for further information.
- IPv6 configuration:
  - The metric for IPv6 routes associated with the WAN.
  - The relative weight for IPv6 routes associated with the WAN.

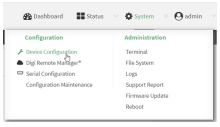
- The IPv6 management priority of the WAN. The active interface with the highest management priority will have its address reported as the preferred contact address for central management and direct device access.
- The IPv6 Maximum Transmission Unit (MTU) of the WAN.
- When to use DNS: always, never, or only when this interface is the primary default route.
- When to use DNS servers for this interface.
- Whether to include the IX15 device's hostname in DHCP requests.
- Active recovery configuration. See Configure SureLink active recovery to detect modem failures for further information.

■ MAC address denylist and allowlist.

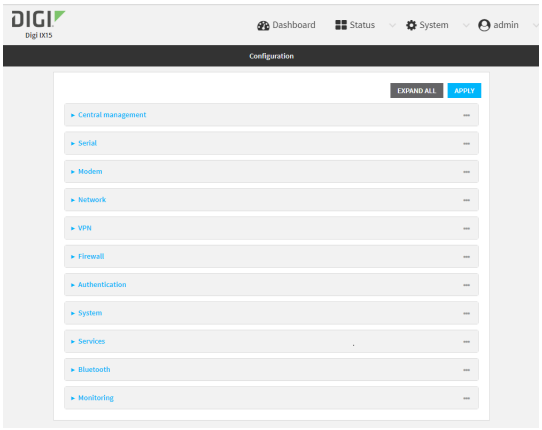
To create a new WAN or edit an existing WAN:

WebUI

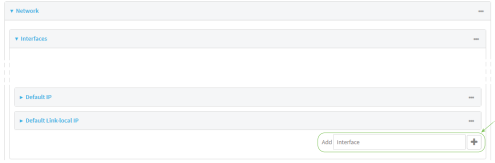
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.

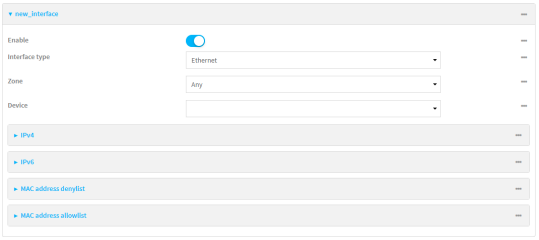


3. Click **Network** > **Interfaces**.
4. Create the WAN or select an existing WAN:
   - To create a new WAN, for **Add interface**, type a name for the WAN and click ✚.



   - To edit an existing WAN, click to expand the WAN.

The Interface configuration window is displayed.



New WANs are enabled by default. To disable, click **Enable**.

5. For **Interface type**, leave at the default setting of **Ethernet**.
6. For **Zone**, select **External**.
7. For **Device**, select an Ethernet device.
8. (Optional) Click to expand **802.1x** to configure 802.1x port based network access control.

   The IX15 can function as an 802.1x authenticator; it does not function as an 802.1x supplicant.

   a. Click to expand **Authentication**.
   b. Click **Enable server** to enable the 802.1x authenticator on the IX15 device.
   c. Type the **Server IP address** of the authentication server.
   d. **Server Port number** defaults to **1812**. Type a new port number for the authentication server if different than the default.
   e. Type the **Server Password** for the authentication server.
   f. Set the **Reauth** period.
   g. (Optional) Click to expand **Accounting**.
   h. Click **Enable server** to enable 802.1x authentication auditing on the IX15 device.
   i. Type the **Server IP address** of the auditing server.
   j. **Server Port number** defaults to **1813**. Type a new port number for the auditing server if different than the default.
   k. Type the **Server Password** for the auditing server.
9. Configure IPv4 settings:

   a. Click to expand **IPv4**.

      IPv4 support is enabled by default.
   b. For **Type**, select **DHCP address**.
   c. Optional IPv4 configuration items:

      i. Set the **MTU**.
      ii. For **Use DNS**, select one of the following:

      - **Always**: DNS will always be used for this WAN; when multiple interfaces have the same DNS server, the interface with the lowest metric will be used for DNS requests.
      - **When primary default route**: Only use the DNS servers provided for this interface when the interface is the primary route.
      - **Never**: Never use DNS servers for this interface.

iii. Enable **DHCP Hostname** to instruct the IX15 device to include the device's system name with DHCP requests as the Client FQDN option. The DHCP server can then be configured to register the device's hostname and IP address with an associated DNS server.

- See RFC4702 for further information about DHCP server support for the Client FQDN option.
- See Configure system information for information about setting the IX15 device's system name.

d. See Configure SureLink active recovery to detect modem failures for information about configuring **Active recovery**.

10. (Optional) Configure IPv6 settings:
    a. Click to expand **IPv6**.
    b. **Enable** IPv6 support.
    c. For **Type**, select **DHCPv6 address**.
    d. For **Prefix length**, type the minimum length of the prefix to assign to this LAN. If the minimum length is not available, then a longer prefix will be used.
    e. For **Prefix ID**, type the identifier used to extend the prefix to the assigned length. Leave blank to use a random identifier.
    f. Set the **MTU**.
    g. For **Use DNS**:
    - **Always**: DNS will always be used for this WAN; when multiple interfaces have the same DNS server, the interface with the lowest metric will be used for DNS requests.
    - **When primary default route**: Only use the DNS servers provided for this interface when the interface is the primary route.
    - **Never**: Never use DNS servers for this interface.

    h. Enable **DHCP Hostname** to instruct the IX15 device to include the device's system name with DHCP requests as the Client FQDN option. The DHCP server can then be configured to register the device's hostname and IP address with an associated DNS server.
    - See RFC4702 for further information about DHCP server support for the Client FQDN option.
    - See Configure system information for information about setting the IX15 device's system name.

11. (Optional) Click to expand **MAC address denylist**.

    Incoming packets will be dropped from any devices whose MAC addresses is included in the **MAC address denylist**.
    a. Click to expand **MAC address denylist**.
    b. For **Add MAC address**, click ✚.
    c. Type the **MAC address**.

12. (Optional) Click to expand **MAC address allowlist**.

    If allowlist entries are specified, incoming packets will only be accepted from the listed MAC addresses.

a. Click to expand **MAC address allowlist**.
b. For **Add MAC address**, click ✚.
c. Type the **MAC address**.

13. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Create a new WAN or edit an existing one:
   - To create a new WAN named **my_wan**:

```
(config)> add network interface my_wan
(config network interface my_wan)>
```

   - To edit an existing WAN named **my_wan**, change to the **my_wan** node in the configuration schema:

```
(config)> network interface my_wan
(config network interface my_wan)>
```

4. Set the appropriate firewall zone:

```
(config network interface my_wan)> zone zone
(config network interface my_wan)>
```

   See Firewall configuration for further information.

5. Select an Ethernet device.
   a. Enter **device ?** to view available devices and the proper syntax.

```
(config network interface my_wan)> device ?

Device: The network device used by this network interface.
Format:
  /network/device/eth
  /network/device/loopback
Current value:

(config network interface my_wan)> device
```

b. Set the device for the LAN:

```
(config network interface my_wan)> device device
(config network interface my_wan)>
```

6. Configure IPv4 settings:
   - IPv4 support is enabled by default. To disable:

```
(config network interface my_wan)> ipv4 enable false
(config network interface my_wan)>
```

   - Configure the WAN to be a DHCP client:

```
(config network interface my_wan)> ipv4 type dhcp
(config network interface my_wan)>
```

   a. Optional IPv4 configuration items:
      i. Set the MTU:

```
(config network interface my_wan)> ipv4 mtu num
(config network interface my_wan)>
```

      ii. Configure how to use DNS:

```
(config network interface my_wan)> ipv4 use_dns value
(config network interface my_wan)>
```

      where *value* is one of:
      - **always**: DNS will always be used for this WAN; when multiple interfaces have the same DNS server, the interface with the lowest metric will be used for DNS requests.
      - **primary**: Only use the DNS servers provided for this interface when the interface is the primary route.
      - **never**: Never use DNS servers for this interface.

      iii. Enable DHCP Hostname to instruct the IX15 device to include the device's system name with DHCP requests as the Client FQDN option. The DHCP server can then be configured to register the device's hostname and IP address with an associated DNS server.

```
(config network interface my_wan)> ipv4 dhcp_hostname true
(config network interface my_wan)>
```

      - See RFC4702 for further information about DHCP server support for the Client FQDN option.
      - See Configure system information for information about setting the IX15 device's system name.

   b. See Configure SureLink active recovery to detect modem failures for information about configuring **Active recovery**.

7. (Optional) Configure IPv6 settings:

a. Enable IPv6 support:

```
(config network interface my_wan)> ipv6 enable true
(config network interface my_wan)>
```

b. Set the IPv6 type to DHCP:

```
(config network interface my_wan)> ipv6 type dhcpv6
(config network interface my_wan)>
```

c. Generally, the default settings for IPv6 support are sufficient. You can view the default IPv6 settings by using the question mark (**?**):

```
(config network interface my_wan)> ipv6 ?

IPv6

 Parameters                Current Value
 ------------------------------------------------------------------------
----------
 dhcp_hostname             false              DHCP Hostname
 enable                    true               Enable
 metric                    0                  Metric
 mgmt                      0                  Management priority
 mtu                       1500               MTU
 type                      dhcpv6             Type
 use_dns                   always             Use DNS
 weight                    10                 Weight

 Additional Configuration
 ------------------------------------------------------------------------
----------
 connection_monitor        Active recovery

(config network interface my_wan)>
```

d. Modify any of the remaining default settings as appropriate.

8. (Optional) To configure 802.1x port based network access control:

Note The IX15 can function as an 802.1x authenticator; it does not function as an 802.1x supplicant.

a. Enable the 802.1x authenticator on the IX15 device:

```
(config network interface my_wan)> 802_1x authentication enable true
(config network interface my_wan)>
```

b. Set the IP address of the authentication server:

```
(config network interface my_wan)> 802_1x authentication ip IPv4_
address
(config network interface my_wan)>
```

c. Set the password for the authentication server:

```
(config network interface my_wan)> 802_1x authentication password
password
(config network interface my_wan)>
```

d. The authentication server port number defaults to **1812**. To set an alternate port number:

```
(config network interface my_wan)> 802_1x authentication port port
(config network interface my_wan)>
```

e. Set the frequency period for reauthorization:

```
(config network interface my_wan)> 802_1x authentication reauth_period
value
(config network interface my_wan)>
```

where *value* is an integer between **0** and **86400**. The default is **3600**.

f. (Optional) Configure 802.1x authentication auditing:

i. Enable authentication auditing on the IX15 device:

```
(config network interface my_wan)> 802_1x accounting enable true
(config network interface my_wan)>
```

ii. Set the IP address of the accounting server:

```
(config network interface my_wan)> 802_1x accounting ip IPv4_
address
(config network interface my_wan)>
```

iii. Set the password for the accounting server:

```
(config network interface my_wan)> 802_1x accounting password
password
(config network interface my_wan)>
```

iv. The accounting server port number defaults to **1813**. To set an alternate port number:

```
(config network interface my_wan)> 802_1x accounting port port
(config network interface my_wan)>
```

9. (Optional) Configure the MAC address deny list.

Incoming packets will be dropped from any devices whose MAC addresses is included in the MAC address denylist.

a. Add a MAC address to the denylist:

```
(config network interface my_wan)> add mac_denylist end mac_address
(config network interface my_wan)>
```

where *mac_address* is a hyphen-separated MAC address, for example, 32-A6-84-2E-81-58.

b. Repeat for each additional MAC address.

10. (Optional) Configure the MAC address allowlist.

    If allowlist entries are specified, incoming packets will only be accepted from the listed MAC addresses.

    a. Add a MAC address to the allowlist:

    ```
    (config network interface my_wan)> add mac_allowlist end mac_address
    (config network interface my_wan)>
    ```

    where *mac_address* is a hyphen-separated MAC address, for example, 32-A6-84-2E-81-58.

    b. Repeat for each additional MAC address.

11. Save the configuration and apply the change:

    ```
    (config network interface my_wan)> save
    Configuration saved.
    >
    ```

12. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Local Area Networks (LANs)

The IX15 device is preconfigured with the following Local Area Networks (LANs):

Interface type	Preconfigured interfaces	Devices	Default configuration
Local Area Network (LAN)	■ **ETH**	■ Ethernet: **ETH**	■ Firewall zone: Internal ■ IP Address: 192.168.2.1/24 ■ DHCP server enabled ■ LAN priority: Metric=5
	■ **Loopback**	■ Ethernet: **Loopback**	■ Firewall zone: Loopback ■ IP address: 127.0.0.1/8
	■ **Default IP**	■ Ethernet: **ETH**	■ Firewall zone: Setup ■ IP address 192.168.210.1/24
	■ **Default Link-local IP**	■ Ethernet: **ETH**	■ Firewall zone: Setup ■ IP address 169.254.100.100/16

You can modify configuration settings for **ETH**, and you can create new LANs.

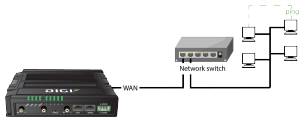
This section contains the following topics:

About Local Area Networks (LANs) 178
Configure a LAN 178
Show LAN status and statistics 185
Delete a LAN 187
DHCP servers 189
Create a Virtual LAN (VLAN) route 205

About Local Area Networks (LANs)

A Local Area Network (LAN) connects network devices together in a logical Layer-2 network.

The following diagram shows a LAN connected to the **ETH** Ethernet device. Once the LAN is configured and enabled, the devices connected to the network interfaces can communicate with each other, as demonstrated by the **ping** commands.



Configure a LAN

Configuring a Local Area Network (LAN) involves configuring the following items:

Required configuration items

- The interface type: either **Ethernet**, **IP Passthrough**, or **PPPoE**.
- The firewall zone: **Internal**.
- The network device that is used by the LAN.
- The IPv4 address and subnet mask for the LAN. While it is not strictly necessary for a LAN to have an IP address, if you want to send traffic from other networks to the LAN, you must configure an IP address.

  Note By default, **ETH** is set to an IP address of 192.168.2.1 and uses the IP subnet of 192.168.2.0/24. If the **WAN/ETH1** Ethernet device is being used by a WAN with the same IP subnet, you should change the default IP address and subnet of LAN1.

Additional configuration items

- Additional IPv4 configuration:
  - The metric for IPv4 routes associated with the LAN.
  - The relative weight for IPv4 routes associated with the LAN.
  - The IPv4 management priority of the LAN. The active interface with the highest management priority will have its address reported as the preferred contact address for central management and direct device access.
  - The IPv4 Maximum Transmission Unit (MTU) of the LAN.
  - When to use DNS: always, never, or only when this interface is the primary default route.
  - IPv4 DHCP server configuration. See DHCP servers for more information.
- IPv6 configuration:
  - The metric for IPv6 routes associated with the LAN.
  - The relative weight for IPv6 routes associated with the LAN.
  - The IPv6 management priority of the LAN. The active interface with the highest management priority will have its address reported as the preferred contact address for central management and direct device access.

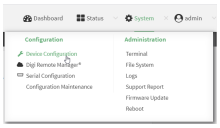
- The IPv6 Maximum Transmission Unit (MTU) of the LAN.
- When to use DNS: always, never, or only when this interface is the primary default route.
- The IPv6 prefix length and ID.
- IPv6 DHCP server configuration. See DHCP servers for more information.

- MAC address denylist and allowlist.

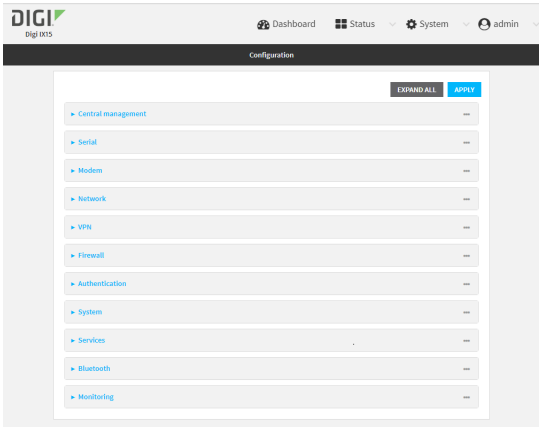
To create a new LAN or edit an existing LAN:

WebUI

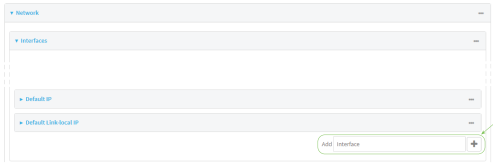
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.

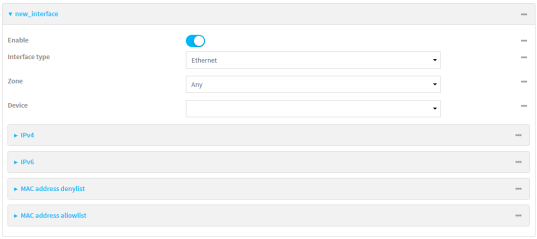


3. Click **Network** > **Interfaces**.
4. Create the LAN or select an existing LAN:
   - To create a new LAN, for **Add interface**, type a name for the LAN and click ✚.



   - To edit an existing LAN, click to expand the LAN.

The Interface configuration window is displayed.



New LANs are enabled by default. To disable, click **Enable**.

5. For **Interface type**, leave at the default setting of **Ethernet**.
6. For **Zone**, select the appropriate firewall zone. See Firewall configuration for further information.
7. For **Device**, select an Ethernet device.
8. (Optional) Click to expand **802.1x** to configure 802.1x port based network access control.

   The IX15 can function as an 802.1x authenticator; it does not function as an 802.1x supplicant.

   a. Click to expand **Authentication**.
   b. Click **Enable server** to enable the 802.1x authenticator on the IX15 device.
   c. Type the **Server IP address** of the authentication server.
   d. **Server Port number** defaults to **1812**. Type a new port number for the authentication server if different than the default.
   e. Type the **Server Password** for the authentication server.
   f. Set the **Reauth** period.
   g. (Optional) Click to expand **Accounting**.
   h. Click **Enable server** to enable 802.1x authentication auditing on the IX15 device.
   i. Type the **Server IP address** of the auditing server.
   j. **Server Port number** defaults to **1813**. Type a new port number for the auditing server if different than the default.
   k. Type the **Server Password** for the auditing server.
9. Configure IPv4 settings:
   a. Click to expand **IPv4**.

      IPv4 support is enabled by default.
   b. For **Type**, select **Static IP address**.
   c. For **Address**, type the IP address and subnet of the LAN interface. Use the format *IPv4_address/netmask*, for example, 192.168.2.1/24.
   d. Optional IPv4 configuration items:
      i. Set the **MTU**.
   e. Enable the DHCP server:
      i. Click to expand **DHCP server**.
      ii. Click **Enable**.

      See DHCP servers for information about configuring the DHCP server.
10. See Configure DHCP relay for information about configuring **DHCP relay**.
11. (Optional) Configure IPv6 settings:

a. Click to expand **IPv6**.
b. **Enable** IPv6 support.
c. For **Type**, select **IPv6 prefix delegration**.
d. For **Prefix length**, type the minimum length of the prefix to assign to this LAN. If the minimum length is not available, then a longer prefix will be used.
e. For **Prefix ID**, type the identifier used to extend the prefix to the assigned length. Leave blank to use a random identifier.
f. Set the **MTU**.

12. (Optional) Click to expand **MAC address denylist**.

    Incoming packets will be dropped from any devices whose MAC addresses is included in the **MAC address denylist**.

    a. Click to expand **MAC address denylist**.
    b. For **Add MAC address**, click ✚.
    c. Type the **MAC address**.

13. (Optional) Click to expand **MAC address allowlist**.

    If allowlist entries are specified, incoming packets will only be accepted from the listed MAC addresses.

    a. Click to expand **MAC address allowlist**.
    b. For **Add MAC address**, click ✚.
    c. Type the **MAC address**.

14. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Create a new LAN or edit an existing one:

   - To create a new LAN named **my_lan**:

```
(config)> add network interface my_lan
(config network interface my_lan)>
```

   - To edit an existing LAN named **my_lan**, change to the **my_lan** node in the configuration schema:

```
(config)> network interface my_lan
(config network interface my_lan)>
```

4. Set the appropriate firewall zone:

```
(config network interface my_lan)> zone zone
(config network interface my_lan)>
```

See Firewall configuration for further information.

5. Select an Ethernet device.
   a. Enter **device ?** to view available devices and the proper syntax.

```
(config network interface my_lan)> device ?

Device: The network device used by this network interface.
Format:
  /network/device/eth
  /network/device/loopback
Current value:

(config network interface my_lan)> device
```

   b. Set the device for the LAN:

```
(config network interface my_lan)> device device
(config network interface my_lan)>
```

6. Configure IPv4 settings:
   - IPv4 support is enabled by default. To disable:

```
(config network interface my_lan)> ipv4 enable false
(config network interface my_lan)>
```

   - The LAN is configured by default to use a static IP address for its IPv4 configuration. To configure the LAN to be a DHCP client, rather than using a static IP addres:

```
(config network interface my_lan)> ipv4 type dhcp
(config network interface my_lan)>
```

     These instructions assume that the LAN will use a static IP address for its IPv4 configuration.

   a. Set the IPv4 address and subnet of the LAN interface. Use the format *IPv4_address*/*netmask*, for example, 192.168.2.1/24.

```
(config network interface my_lan)> ipv4 address ip_address/netmask
(config network interface my_lan)>
```

   b. Optional IPv4 configuration items:

i. Set the MTU:

```
(config network interface my_lan)> ipv4 mtu num
(config network interface my_lan)>
```

c. Enable the DHCP server:

```
(config network interface my_lan)> ipv4 dhcp_server enable true
```

See DHCP servers for information about configuring the DHCP server.

7. (Optional) Configure IPv6 settings:

a. Enable IPv6 support:

```
(config network interface my_lan)> ipv6 enable true
(config network interface my_lan)>
```

b. Set the IPv6 type to DHCP:

```
(config network interface my_lan)> ipv6 type dhcpv6
(config network interface my_lan)>
```

c. Generally, the default settings for IPv6 support are sufficient. You can view the default IPv6 settings by using the question mark (**?**):

```
(config network interface my_lan)> ipv6 ?

IPv6

 Parameters                Current Value
 ------------------------------------------------------------------------
----------
 enable                    true              Enable
 metric                    0                 Metric
 mgmt                      0                 Management priority
 mtu                       1500              MTU
 prefix_id                 1                 Prefix ID
 prefix_length             48                Prefix length
 type                      prefix_delegation  Type
 weight                    10                Weight

 Additional Configuration
 ------------------------------------------------------------------------
----------
 connection_monitor        Active recovery
 dhcpv6_server             DHCPv6 server

(config network interface my_lan)>
```

View default settings for the IPv6 DHCP server:

```
(config network interface my_lan)> ipv6 dhcpv6_server ?
```

```
DHCPv6 server: The DHCPv6 server settings for this network interface.

 Parameters                Current Value
 ------------------------------------------------------------------------
----------
 enable                    true           Enable

(config network interface my_lan)>
```

d. Modify any of the remaining default settings as appropriate.

8. (Optional) To configure 802.1x port based network access control:

Note The IX15 can function as an 802.1x authenticator; it does not function as an 802.1x supplicant.

a. Enable the 802.1x authenticator on the IX15 device:

```
(config network interface my_lan)> 802_1x authentication enable true
(config network interface my_lan)>
```

b. Set the IP address of the authentication server:

```
(config network interface my_lan)> 802_1x authentication ip IPv4_
address
(config network interface my_lan)>
```

c. Set the password for the authentication server:

```
(config network interface my_lan)> 802_1x authentication password
password
(config network interface my_lan)>
```

d. The authentication server port number defaults to **1812**. To set an alternate port number:

```
(config network interface my_lan)> 802_1x authentication port port
(config network interface my_lan)>
```

e. Set the frequency period for reauthorization:

```
(config network interface my_lan)> 802_1x authentication reauth_period
value
(config network interface my_lan)>
```

where *value* is an integer between **0** and **86400**. The default is **3600**.

f. (Optional) Configure 802.1x authentication auditing:

i. Enable authentication auditing on the IX15 device:

```
(config network interface my_lan)> 802_1x accounting enable true
(config network interface my_lan)>
```

ii. Set the IP address of the accounting server:

```
(config network interface my_lan)> 802_1x accounting ip IPv4_
address
(config network interface my_lan)>
```

iii. Set the password for the accounting server:

```
(config network interface my_lan)> 802_1x accounting password
password
(config network interface my_lan)>
```

iv. The accounting server port number defaults to **1813**. To set an alternate port number:

```
(config network interface my_lan)> 802_1x accounting port port
(config network interface my_lan)>
```

9. (Optional) Configure the MAC address deny list.

   Incoming packets will be dropped from any devices whose MAC addresses is included in the MAC address denylist.

   a. Add a MAC address to the denylist:

   ```
   (config network interface my_lan)> add mac_denylist end mac_address
   (config network interface my_lan)>
   ```

   where *mac_address* is a hyphen-separated MAC address, for example, 32-A6-84-2E-81-58.

   b. Repeat for each additional MAC address.

10. (Optional) Configure the MAC address allowlist.

    If allowlist entries are specified, incoming packets will only be accepted from the listed MAC addresses.

    a. Add a MAC address to the allowlist:

    ```
    (config network interface my_lan)> add mac_allowlist end mac_address
    (config network interface my_lan)>
    ```

    where *mac_address* is a hyphen-separated MAC address, for example, 32-A6-84-2E-81-58.

    b. Repeat for each additional MAC address.

11. Save the configuration and apply the change:

    ```
    (config network interface my_lan)> save
    Configuration saved.
    >
    ```

12. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show LAN status and statistics

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. From the menu, click **Status**.
3. Under **Networking**, click **Interfaces**.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Enter the show network command at the Admin CLI prompt:

```
> show network

 Interface         Proto  Status   Address
 ----------------  -----  -------  ------------------------------
 defaultip         IPv4   up       192.168.210.1/24
 defaultlinklocal  IPv4   up       169.254.100.100/16
 eth1              IPv4   up       10.10.10.10/24
 eth1              IPv6   up       fe00:2404::240:f4ff:fe80:120/64
 eth              IPv4   up       192.168.2.1/24
 eth              IPv6   up       fd00:2704::1/48
 loopback          IPv4   up       127.0.0.1/8
 modem             IPv4   up       10.200.1.101/30
 modem             IPv6   down

>
```

3. Additional information can be displayed by using the show network verbose command:

```
> show network verbose

 Interface         Proto  Status   Type    Zone      Device    Metric
Weight
 ----------------  -----  -------  ------  --------  --------  ------  --
----
 defaultip         IPv4   up       static  setup     eth      10      10
 defaultlinklocal  IPv4   up       static  setup     eth      0       10
 eth1              IPv4   up       dhcp    external  eth1      1       10
 eth1              IPv6   up       dhcp    external  eth1      1       10
 eth              IPv4   up       static  internal  eth      5       10
 eth              IPv6   up       static  internal  eth      5       10
 loopback          IPv4   up       static  loopback  loopback  0       10
 modem             IPv4   up       modem   external  wwan1     3       10
 modem             IPv6   down     modem   external  wwan1     3       10

>
```

4. Enter **show network interface *name*** at the Admin CLI prompt to display additional information about a specific LAN. For example, to display information about ETH, enter **show network interface *eth***:

```
> show network interface eth

 lan1 Interface Status
 ---------------------
 Device              : eth
 Zone                : internal

 IPv4 Status         : up
 IPv4 Type           : static
 IPv4 Address(es)    : 192.168.2.1/24
 IPv4 Gateway        :
 IPv4 MTU            : 1500
 IPv4 Metric         : 5
 IPv4 Weight         : 10
 IPv4 DNS Server(s)  :

 IPv6 Status         : up
 IPv6 Type           : prefix
 IPv6 Address(es)    : fd00:2704::1/48
 IPv6 Gateway        :
 IPv6 MTU            : 1500
 IPv6 Metric         : 5
 IPv6 Weight         : 10
 IPv6 DNS Server(s)  :

>
```

5. Type **exit** to exit the Admin CLI.

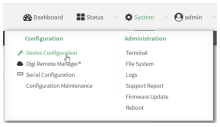
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Delete a LAN

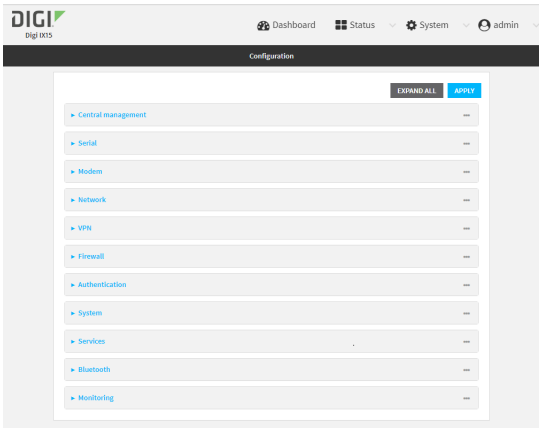
Follow this procedure to delete any LANs that have been added to the system. You cannot delete the preconfigured LAN, **LAN1**.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. Click the menu icon (**...**) next to the name of the LAN to be deleted and select **Delete**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Use the **del** command to delete the LAN. For example, to delete a LAN named my_lan:

```
(config)> del network interface my_lan
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

DHCP servers

You can enable DHCP on your IX15 device to assign IP addresses to clients, using either:

- The DHCP server for the device's local network, which assigns IP addresses to clients on the device's local network. Addresses are assigned from a specified pool of IP addresses. For a local network, the device uses the DHCP server that has the IP address pool in the same IP subnet as the local network.

  When a host receives an IP configuration, the configuration is valid for a particular amount of time, known as the lease time. After this lease time expires, the configuration must be renewed. The host renews the lease time automatically.
- A DHCP relay server, which forwards DHCP requests from clients to a DHCP server that is running on a separate device.

Configure a DHCP server

Note These instructions assume you are configuring the device to use its local DHCP server. For instructions about configuring the device to use a DHCP relay server, see Configure DHCP relay.

Required configuration items

- Enable the DHCP server.

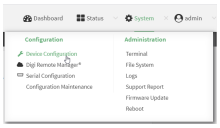
Additional configuration items

- The lease address pool: the range of IP addresses issued by the DHCP server to clients.
- Lease time: The length, in minutes, of the leases issued by the DHCP server.

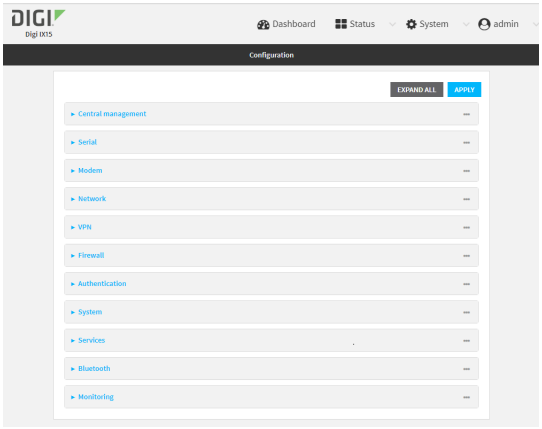
- The Maximum Transmission Units (MTU).
- The domain name suffix appended to host names.
- The IP gateway address given to clients.
- The IP addresses of the preferred and alternate Domain Name Server (DNS), NTP servers, and WINS severs that are given to clients.
- The TFTP server name.
- The filepath and name of the bootfile on the TFTP server.
- Custom DHCP options. See Configure DHCP options for information about custom DHCP options.
- Static leases. See Map static IP addresses to hosts for information about static leases.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. Click to expand an existing LAN, or create a new LAN. See Configure a LAN.
5. Click to expand **IPv4** > **DHCP server**.
6. **Enable** the DHCP server.
7. (Optional) For **Lease time**, type the amount of time that a DHCP lease is valid.

   Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

   For example, to set **Lease time** to ten minutes, enter **10m** or **600s**.

   The default is 12 hours.

8. (Optional) For **Lease range start** and **Lease range end**, type the lowest and highest IP address that the DHCP server will assign to a client. This value represents the low order byte of the address (the final triplet in an IPv4 address, for example, 192.168.2.**xxx**). The remainder of the IP address will be based on the LAN's static IP address as defined in the **Address** field.

   Allowed values are between **1** and **254**, and the default is **100** for **Lease range start** and **250** for **Lease range end**.
9. Optional DHCP server settings:
   a. Click to expand **Advanced settings**.
   b. For **Gateway**, select either:
      - **None**: No gateway is broadcast by the DHCP server. Client destinations must be resolvable without a gateway.
      - **Automatic**: Broadcasts the IX15 device's gateway.
      - **Custom**: Allows you to identify the IP address of a **Custom gateway** to be broadcast.

      The default is **Automatic**.
   c. For **MTU**,
      - **None**: An MTU of length **0** is broadcast. This is not recommended.
      - **Automatic**: No MTU is broadcast and clients will determine their own MTU.
      - **Custom**: Allows you to identify a **Custom MTU** to be broadcast.

      The default is **Automatic**.
   d. For **Domain name suffix**, type the domain name that should be appended to host names.
   e. For **Primary** and **Secondary DNS**, **Primary** and **Secondary NTP server**, and **Primary** and **Secondary WINS server**, select either:
      - **None**: No server is broadcast.
      - **Automatic**: Broadcasts the IX15 device's server.
      - **Custom**: Allows you to identify the IP address of the server.
   f. For **Bootfile name**, type the relative path and file name of the bootfile on the TFTP server.
   g. For **TFTP server** name, type the IP address or host name of the TFTP server.
10. See Configure DHCP options for information about **Custom DHCP options**.
11. See Map static IP addresses to hosts for information about **Static leases**.
12. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable the DHCP server for an existing LAN. For example, to enable the DHCP server for a LAN named **my_lan**:

```
(config)> network interface my_lan ipv4 dhcp_server enable true
(config)>
```

See Configure a LAN for information about creating a LAN.

4. (Optional) Set the amount of time that a DHCP lease is valid:

```
(config)> network interface my_lan ipv4 dhcp_server lease_time value
(config)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w**|**d**|**h**|**m**|**s**}.

For example, to set **network interface my_lan ipv4 dhcp_server lease_time** to ten minutes, enter either **10m** or **600s**:

```
(config)> network interface my_lan ipv4 dhcp_server lease_time 600s
(config)>
```

5. (Optional) Set the lowest IP address that the DHCP server will assign to a client. This value represents the low order byte of the address (the final triplet in an IPv4 address, for example, 192.168.2.**xxx**). The remainder of the IP address will be based on the LAN's static IP address as defined in the **address** parameter.

```
(config)> network interface my_lan ipv4 dhcp_server lease_start num
(config)>
```

Allowed values are between **1** and **254**, and the default is **100**.

6. (Optional) Set the highest IP address that the DHCP server will assign to a client:

```
(config)> network interface my_lan ipv4 dhcp_server lease_end num
(config)>
```

Allowed values are between **1** and **254**, and the default is **250**.

7. Optional DHCP server settings:
   a. Click to expand **Advanced settings**.
   b. Determine how the DHCP server should broadcast the gateway server:

```
(config)> network interface my_lan ipv4 dhcp_server advanced gateway
value
(config)>
```

where **value** is one of:

- **none**: No gateway is broadcast by the DHCP server. Client destinations must be resolvable without a gateway.

- **auto**: Broadcasts the IX15 device's gateway.
- **custom**: Allows you to identify the IP address of a custom gateway to be broadcast:

```
(config)> network interface my_lan ipv4 dhcp_server advanced
gateway_custom ip_address
(config)>
```

The default is **auto**.

c. Determine how the DHCP server should broadcast the the MTU:

```
(config)> network interface my_lan ipv4 dhcp_server advanced mtu value
(config)>
```

where **value** is one of:

- **none**: An MTU of length **0** is broadcast. This is not recommended.
- **auto**: No MTU is broadcast and clients will determine their own MTU.
- **custom**: Allows you to identify a custom MTU to be broadcast:

```
(config)> network interface my_lan ipv4 dhcp_server advanced
mtu_custom mtu
(config)>
```

The default is **auto**.

d. Set the domain name that should be appended to host names:

```
(config)> network interface my_lan ipv4 dhcp_server advanced domain_
suffix name
(config)>
```

e. Set the IP address or host name of the primary and secondary DNS, the primary and secondary NTP server, and the primary and secondary WINS servers:

```
(config)> network interface my_lan ipv4 dhcp_server advanced primary_
dns value
(config)> network interface my_lan ipv4 dhcp_server advanced
secondary_dns value
(config)> network interface my_lan ipv4 dhcp_server advanced primary_
ntp value
(config)> network interface my_lan ipv4 dhcp_server advanced
secondary_ntp value
(config)> network interface my_lan ipv4 dhcp_server advanced primary_
wins value
(config)> network interface my_lan ipv4 dhcp_server advanced
secondary_wins value
(config)>
```

where **value** is one of:

- **none**: No server is broadcast.
- **auto**: Broadcasts the IX15 device's server.

- **custom**: Allows you to identify the IP address of the server. For example:

```
(config)> network interface my_lan ipv4 dhcp_server advanced
primary_dns_custom ip_address
(config)>
```

The default is **auto**.

f. Set the IP address or host name of the TFTP server:

```
(config)> network interface my_lan ipv4 dhcp_server advanced nftp_
server ip_address
(config)>
```

g. Set the relative path and file name of the bootfile on the TFTP server:

```
(config)> network interface my_lan ipv4 dhcp_server advanced bootfile
filename
(config)>
```

8. See Configure DHCP options for information about custom DHCP options.
9. See Map static IP addresses to hosts for information about static leases.
10. Save the configuration and apply the change:

```
(config network interface my_lan ipv4 dhcp_server advanced static_lease
0)> save
Configuration saved.
>
```

11. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Map static IP addresses to hosts

You can configure the DHCP server to assign static IP addresses to specific hosts.

Required configuration items

- IP address that will be mapped to the device.
- MAC address of the device.

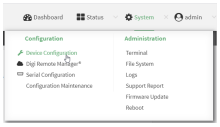
Additional configuration items

- A label for this instance of the static lease.

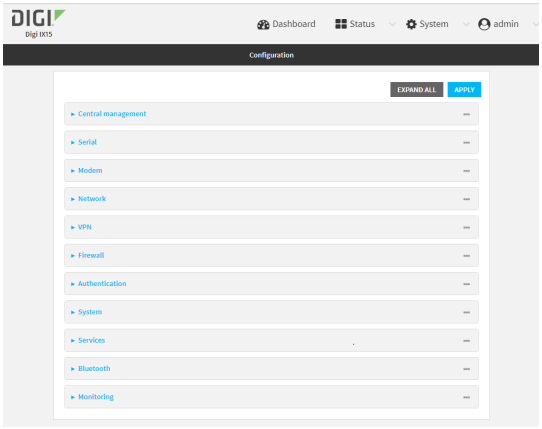
To map static IP addresses:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



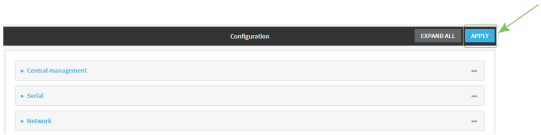
The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. Click to expand an existing LAN, or create a new LAN. See Configure a LAN.
5. Click to expand **IPv4** > **DHCP server** > **Advanced settings** > **Static leases**.
6. For **Add Static lease**, click ✚.
7. Type the **MAC address** of the device associated with this static lease.
8. Type the **IP address** for the static lease.

Note The IP address here should be outside of the DHCP server's configured lease range. See Configure a DHCP server for further information about the lease range.

9. (Optional) For **Hostname**, type a label for the static lease. This does not have to be the device's actual hostname.
10. Repeat for each additional DHCP static lease.
11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a static lease to the DHCP server configuration for an existing LAN. For example, to add static lease to a LAN named **my_lan**:

```
(config)> add network interface my_lan ipv4 dhcp_server advanced static_
lease end
(config network interface my_lan ipv4 dhcp_server advanced static_lease
0)>
```

See Configure a LAN for information about creating a LAN.

4. Set the MAC address of the device associated with this static lease, using the colon-separated format:

```
(config network interface my_lan ipv4 dhcp_server advanced static_lease
0)> mac 00:40:D0:13:35:36
(config network interface my_lan ipv4 dhcp_server advanced static_lease
0)>
```

5. Set the IP address for the static lease:

```
(config network interface my_lan ipv4 dhcp_server advanced static_lease
0)> ip 10.01.01.10
(network interface my_lan ipv4 dhcp_server advanced static_lease 0)>
```

Note The IP address here should be outside of the DHCP server's configured lease range. See Configure a DHCP server for further information about the lease range.

6. (Optional) Set a label for this static lease:

```
(config network interface my_lan ipv4 dhcp_server advanced static_lease
0)> name label
(config network interface my_lan ipv4 dhcp_server advanced static_lease
0)>
```

7. Save the configuration and apply the change:

```
(config network interface my_lan ipv4 dhcp_server advanced static_lease
0)> save
Configuration saved.
>
```

8. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show current static IP mapping

To view your current static IP mapping:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **Status**
3. Under **Networking**, click **DHCP Leases**.

Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Show the static lease configuration. For example, to show the static leases for a lan named **my_lan**:

```
(config)> show network interface my_lan ipv4 dhcp_server advanced static_
lease
0
    ip 192.168.2.10
    mac BF:C3:46:24:0E:D9
    no name
1
    ip 192.168.2.11
    mac E3:C1:1F:65:C3:0E
    no name
(config)>
```

4. Type **cancel** to exit configuration mode:

```
(config)> cancel
>
```

5. Type exit to exit the Admin CLI.

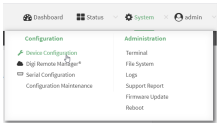
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Delete static IP mapping entries

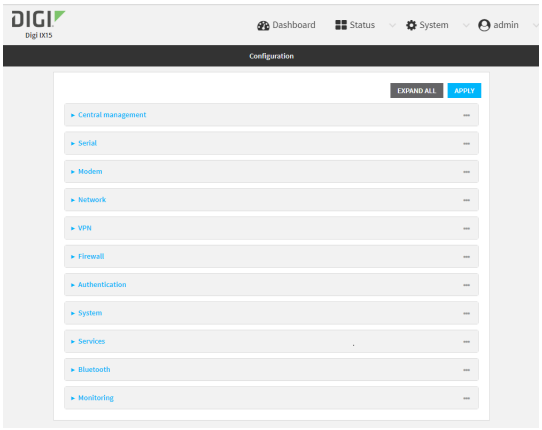
To delete a static IP entry:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. Click to expand an existing LAN.
5. Click to expand **IPv4** > **DHCP server** > **Advanced settings** > **Static leases**.
6. Click the menu icon (**...**) next to the name of the static lease to be deleted and select **Delete**.



7. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Show the static lease configuration. For example, to show the static leases for a lan named **my_lan**:

```
(config)> show network interface my_lan ipv4 dhcp_server advanced static_
lease
0
    ip 192.168.2.10
    mac BF:C3:46:24:0E:D9
    no name
1
```

```
    ip 192.168.2.11
    mac E3:C1:1F:65:C3:0E
    no name
(config)>
```

4. Use the **del *index_number*** command to delete a static lease. For example, to delete the static lease for the device listed in the above output with a mac address of BF:C3:46:24:0E:D9 (index number **0**):

```
(config)> del network interface lan1 ipv4 dhcp_server advanced static_
lease 0
(config)>
```

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure DHCP options

You can configure DHCP servers running on your Digi IX15 Gateway device to send certain specified DHCP options to DHCP clients. You can also set the user class, which enables you to specify which specific DHCP clients will receive the option. You can also force the command to be sent to the clients.

DHCP options can be set on a per-LAN basis, or can be set for all LANs. A total of 32 DHCP options can be configured.

Required configuration items

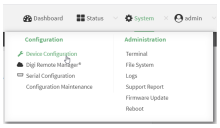
- DHCP option number.
- Value for the DHCP option.

Additional configuration items

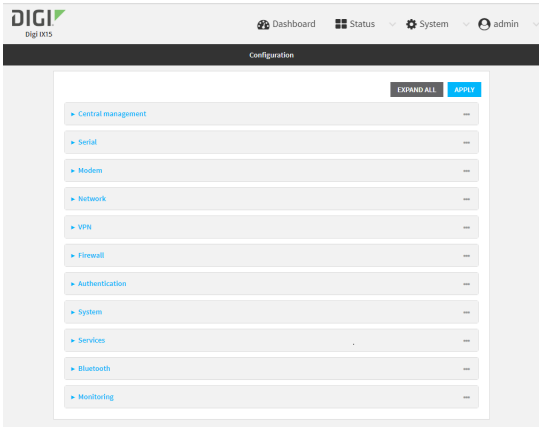
- The data type of the value.
- Force the option to be sent to the DHCP clients.
- A label for the custom option.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. Click to expand an existing LAN, or create a new LAN. See Configure a LAN.
5. Click to expand **IPv4** > **DHCP server** > **Advanced settings** > **Custom DHCP option**.
6. For **Add Custom option**, click ✚.

   Custom options are enabled by default. To disable, uncheck **Enable**.
7. For **Option number**, type the DHCP option number.
8. For **Value**, type the value of the DHCP option.
9. (Optional) For **Label**, type a label for the custom option.
10. (Optional) If **Forced send** is enabled, the DHCP option will always be sent to the client, even if the client does not ask for it.
11. (Optional) For **Data type**, select the data type that the option uses. If the incorrect data type is selected, the device will send the value as a string.
12. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a custom DHCP option to the DHCP server configuration for an existing LAN. For example, to add static lease to a LAN named **my_lan**:

```
(config)> add network interface my_lan ipv4 dhcp_server advanced custom_
option end
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)>
```

   See Configure a LAN for information about creating a LAN.
4. Custom options are enabled by default. To disable:

```
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)> enable false
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)>
```

5. Set the option number for the DHCP option:

```
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)> option 210
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)>
```

6. Set the value for the DHCP option:

```
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)> value_str value
(network interface my_lan ipv4 dhcp_server advanced custom_option 0)>
```

7. (Optional) Set a label for this custom option:

```
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)> name label
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)>
```

8. (Optional) To force the DHCP option to always be sent to the client, even if the client does not ask for it:

```
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)> force true
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)>
```

9. (Optional) Set the data type that the option uses.

If the incorrect data type is selected, the device will send the value as a string.

```
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)> datatype value
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)>
```

where *value* is one of:

- **1byte**
- **2byte**
- **4byte**
- **hex**
- **ipv4**
- **str**

The default is **str**.

10. Save the configuration and apply the change:

```
(config network interface my_lan ipv4 dhcp_server advanced custom_option
0)> save
Configuration saved.
>
```

11. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure DHCP relay

DHCP relay allows a router to forward DHCP requests from one LAN to a separate DHCP server, typically connected to a different LAN.

For the IX15 device, DHCP relay is configured by providing the IP address of a DHCP relay server, rather than an IP address range. If both the DHCP relay server and an IP address range are specified, DHCP relay is used, and the specified IP address range is ignored.

Multiple DHCP relay servers can be provided for each LAN. If multiple relay servers are provided, DHCP requests are forwarded to all servers without waiting for a response. Clients will typically use the IP address from the first DHCP response received.

Configuring DHCP relay involves the following items:

Required configuration items

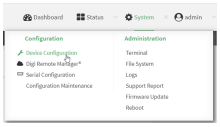
- Disable the DHCP server, if it is enabled.
- IP address of the primary DHCP relay server, to define the relay server that will respond to DHCP requests.

Additional configuration items

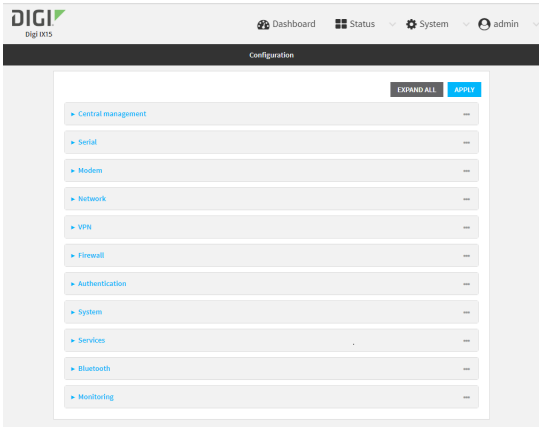
- IP address of additional DHCP relay servers.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. Click to expand an existing LAN, or create a new LAN. See Configure a LAN.
5. Disable the DHCP server, if it is enabled:
   a. Click to expand **IPv4** > **DHCP server**.
   b. Click **Enable** to toggle off the DHCP server.
6. Click to expand **DHCP relay**.
7. For **Add DHCP Server:**, click ✚.
8. For **DHCP server address**, type the IP address of the relay server.
9. Repeat for each additional DHCP relay server.
10. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a DHCP relay server to an existing LAN. For example, to add a server to a LAN named **my_lan**:

```
(config)> add network interface my_lan ipv4 dhcp_relay end
(config network interface lan1 my_lan dhcp_relay 0)>
```

See Configure a LAN for information about creating a LAN.

4. Set the IP address of the DHCP relay server:

```
(config network interface my_lan ipv4 dhcp_relay 0)> address 10.10.10.10
(config network interface my_lan ipv4 dhcp_relay 0)>
```

5. (Optional) Add additional DHCP relay servers:
   a. Move back one step in the configuration schema by typing two periods (**..**):

```
(config network interface my_lan ipv4 dhcp_relay 0)> ..
(config network interface my_lan ipv4 dhcp_relay)>
```

   b. Add the next server:

```
(config network interface lan1 ipv4 dhcp_relay)> add end
(config network interface lan1 ipv4 dhcp_relay 1)>
```

   c. Set the IP address of the DHCP relay server:

```
(config network interface my_lan ipv4 dhcp_relay 1)> address
10.10.10.11
(config network interface my_lan ipv4 dhcp_relay 1)>
```

   d. Repeat for each additional relay server.

1. Disable the DHCP server, if it is enabled:

```
(config network interface my_lan ipv4 dhcp_relay 1)> .. .. dhcp_server
enable false
(config network interface my_lan ipv4 dhcp_relay 1)>
```

6. Save the configuration and apply the change:

```
(config network interface lan1 ipv4 dhcp_relay 1)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show DHCP server status and settings

View DHCP status to monitor which devices have been given IP configuration by the Digi IX15 Gateway device and to diagnose DHCP issues.

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **Status**
3. Under **Networking**, click **DHCP Leases**.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Enter the show dhcp-lease command at the Admn CLI prompt:

```
> show dhcp-lease

 IP Address     Hostname         Expires
 -------------  ---------------  -------
 192.168.2.194  MTK-ENG-USER1
 192.168.2.195  MTK-ENG-USER2

>
```

3. Additional information can be returned by using the show dhcp-lease verbose command:

```
> show dhcp-lease verbose

 IP Address     Hostname      Expires                   Type     Active
MAC Address
 -------------  --------      ------------------------  -------  ------
-----------------
 192.168.2.194  MTK-ENG-USER1 May 19 08:25:11 UTC 2021  Dynamic  Yes
ba:ba:2c:13:8c:71
 192.168.2.195  MTK-ENG-USER2 May 20 11:32:12 UTC 2021  Dynamic  Yes
09:eb:10:f0:bc:16

>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Create a Virtual LAN (VLAN) route

Virtual LANs (VLANs) allow splitting a single physical LAN into separate Virtual LANs. This is useful for security reasons, and also helps to reduce broadcast traffic on the LAN.

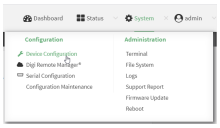
Required configuration items

- Device to be assigned to the VLAN.
- The VLAN ID. The TCP header uses the VLAN ID to identify the destination VLAN for the packet.

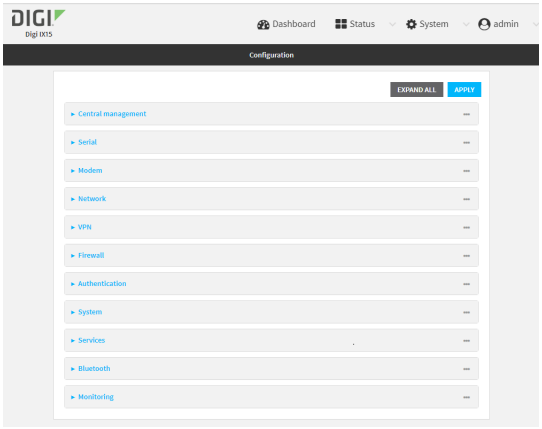
To create a VLAN:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Virtual LAN**.
4. Type a name for the VLAN and click ✚.
5. Select the **Device**.
6. Type or select a unique numeric **ID** for the VLAN ID.
7. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add the VLAN:

```
(config)> add network vlan name
(config)>
```

4. Set the device to be used by the VLAN:
   a. View a list of available devices:

```
(config network vlan vlan1)> device ?

Device: The Ethernet device to use for this virtual LAN
Format:

  /network/device/loopback
  /network/vlan/vlan1
Current value:

(config network vlan vlan1)>
```

   b. Add the device:

```
(config network vlan vlan1)> device /network/device/
(config network vlan vlan1)>
```

5. Set the VLAN ID:

```
(config network vlan vlan1)> id value
```

   where *value* is an integer between **1** and **4095**.
6. Save the configuration and apply the change:

```
(config network vlan vlan1)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show Surelink status and statistics

You can show Surelink status for all interfaces, or for an individual interface. You can also show Surelink status for ipsec tunnels and OpenVPN clients.

Surelink status is only available from the Admin CLI.

Command line

Show Surelink status for all interfaces

To show the Surelink status all interfaces, use the show surelink interface all command:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the Admin CLI prompt, type :

```
> show surelink interface all

 Interface  Test                          Proto  Last Response  Status
 ---------  ----------------------------  -----  -------------  -------
 eth1       Interface is up               IPv4   32 seconds     Passing
 eth1       Interface's DNS servers (DNS) IPv4   28 seconds     Passing
 eth       Interface is up               IPv4   21 seconds     Passing
 eth       Interface's DNS servers (DNS) IPv4   20 seconds     Passing
 modem      Interface is up               IPv4   115 seconds    Passing
 modem      Interface's DNS servers (DNS) IPv4   114 seconds    Passing

>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show Surelink status for a specific interface

To show the Surelink status a specific interface, use the show surelink interface name *name* command:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Use the show surelink interface name *name* command to show the Surelink status of a specific interface, for example:

```
> show surelink interface name eth1

 Interface  Test                          Proto  Last Response  Status
 ---------  ----------------------------  -----  -------------  -------
 eth1       Interface is up               IPv4   32 seconds    Passing
```

```
 eth1       Interface's DNS servers (DNS)  IPv4   28 seconds
Passingsing

>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show Surelink status for all IPsec tunnels

To show the Surelink status all IPsec tunnels, use the show surelink ipsec all command:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the Admin CLI prompt, type :

```
> show surelink ipsec all

 IPsec   Test                 Last Response  Status
 ------  -------------------  -------------  ------------------
 test    194.43.79.74 (Ping)  29 seconds     Passed
 test    194.43.79.75 (Ping)  5 seconds      Passed
 test1   194.43.79.74 (Ping)  21 seconds     Failed
 test2   194.43.79.75 (Ping)  21 seconds     Waiting for result

>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show Surelink status for a specific IPsec tunnel

To show the Surelink status a specific IPsec tunnel, use the show surelink ipsec tunnel *name* command:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Use the show surelink ipsec tunnel *name* command to show the Surelink status of a specific tunnel, for example:

```
> show surelink ipsec tunnel test

 IPsec   Test                 Last Response  Status
 ------  -------------------  -------------  ------------------
 test    194.43.79.74 (Ping)  29 seconds     Passed
```

```
 test    194.43.79.75 (Ping)   5 seconds      Passed

>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show Surelink status for all OpenVPN clients

To show the Surelink status all OpenVPN clients, use the show surelink openvpn client all command:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the Admin CLI prompt, type :

```
> show surelink openvpn all

 OpenVPN Client   Test                 Last Response  Status
 --------------  -------------------  -------------  ------------------
 test_client1    194.43.79.74 (Ping)   29 seconds     Passed
 test_client1    194.43.79.75 (Ping)   5 seconds      Passed
 test_client2    194.43.79.74 (Ping)   21 seconds     Failed
 test_client2    194.43.79.75 (Ping)   21 seconds     Waiting for result

>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show Surelink status for a specific OpenVPN client

To show the Surelink status a specific OpenVPN client, use the show surelink openvpn client *name* command:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Use the show surelink openvpn client *name* command to show the Surelink status of a specific OpenVPN client, for example:

```
> show surelink openvpn client test_client1

 OpenVPN Client   Test                 Last Response  Status
 --------------  -------------------  -------------  ------------------
 test_client1    194.43.79.74 (Ping)   29 seconds     Passed
 test_client1    194.43.79.75 (Ping)   5 seconds      Passed
```

```
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Serial port

IX15 devices have a single serial port that provides access to the command-line interface.
The default serial port configuration is:

Enabled

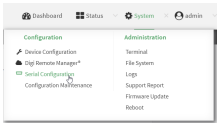
- **Serial mode**: Remote
- **Label**: None
- **Baud rate**: 9600
- **Data bits**: 8
- **Parity**: None
- **Stop bits**: 1
- **Flow control**: None

Configure Login mode

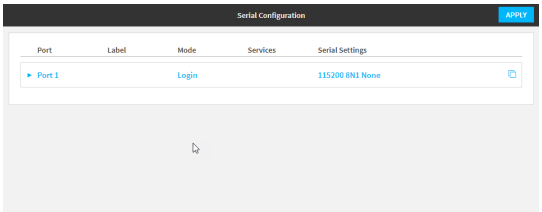
Login mode allows the user to log into the device through the serial port.
To change the configuration to match the serial configuration of the device to which you want to connect:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Configuration**, click **Serial Configuration**.

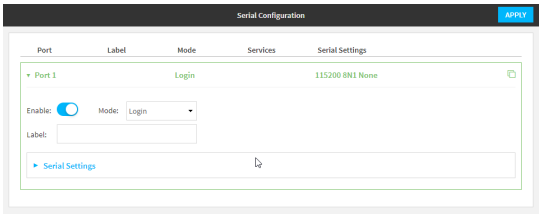


The **Serial Configuration** page is displayed.



Note You can also configure the serial port by using **Device Configuration** > **Serial**. Changes made by using either **Device Configuration** or **Serial Configuration** will be reflected in both.

3. Click the name of the port that you want to configure.



The serial port is enabled by default. To disable, toggle off **Enable**.

4. For **Mode**, select **Login**. This is the default.
5. (Optional) For **Label**, enter a label that will be used when referring to this port.
6. Expand **Serial Settings**.

   The entries in the following fields must match the information for the power controller. Refer to your power controller manual for the correct entries.

   a. **Baud rate**: For **Baud rate**, select the baud rate used by the device to which you want to connect. The default is **9600**.

   b. **Data bits**: For **Data bits**, select the number of data bits used by the device to which you want to connect. The default is **8**.

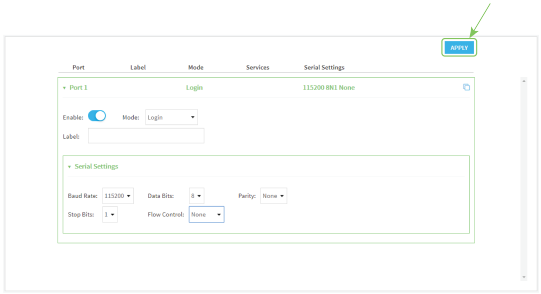
   c. **Parity**: For **Parity**, select the type of parity used by the device to which you want to connect. The default is **None**.

   d. **Stop bits**: For **Stop bits**, select the number of stop bits used by the device to which you want to connect. The default is **1**.

   e. **Flow control**: For **Flow control**, select the type of flow control used by the device to which you want to connect. The default is **None**.

7. Click **Apply** to save the configuration and apply the change.

   The **Apply** button is located at the top of the WebUI page. You may need to scroll to the top of the page to locate it.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

   ```
   > config
   (config)>
   ```
3. The serial port is enabled by default. To disable:

   ```
   (config)> serial port1 enable false
   (config)>
   ```
4. Set the mode:

   ```
   (config)> serial port1 mode login
   (config)>
   ```
5. (Optional) Set a label that will be used when referring to this port.

   ```
   (config)>path-paramlabel label
   (config)>
   ```
6. Set the baud rate used by the device to which you want to connect:

   ```
   (config)>path-parambaudrate rate
   (config)>
   ```
7. Set the number of data bits used by the device to which you want to connect:

   ```
   (config)>path-paramdatabits bits
   (config)>
   ```
8. Set the type of parity used by the device to which you want to connect:

   ```
   (config)>path-paramparity parity
   (config)>
   ```

   Allowed values are:

   - **even**
   - **odd**
   - **none**

   The default is **none**.
9. Set the stop bits used by the device to which you want to connect:

   ```
   (config)>path-paramstopbits bits
   (config)>
   ```

10. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

11. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

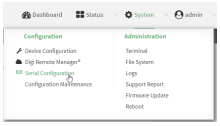
Configure Remote Access mode

Remote Access mode allows for remote access to another device that is connected to the serial port.

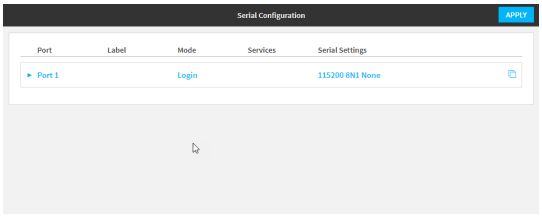
To change the configuration to match the serial configuration of the device to which you want to connect:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Configuration**, click **Serial Configuration**.

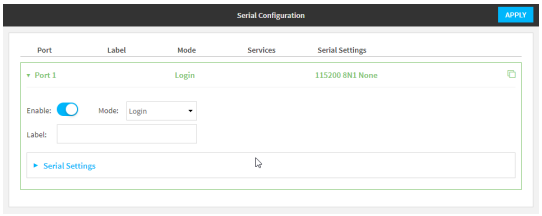


The **Serial Configuration** page is displayed.



Note You can also configure the serial port by using **Device Configuration** > **Serial**. Changes made by using either **Device Configuration** or **Serial Configuration** will be reflected in both.

3. Click the name of the port that you want to configure.



The serial port is enabled by default. To disable, toggle off **Enable**.

4. For **Mode**, select **Remote Access**. The default is **Remote**.
5. (Optional) For **Label**, enter a label that will be used when referring to this port.

6. Expand **Serial Settings**.

   The entries in the following fields must match the information for the power controller. Refer to your power controller manual for the correct entries.

   a. **Baud rate**: For **Baud rate**, select the baud rate used by the device to which you want to connect. The default is **9600**.

   b. **Data bits**: For **Data bits**, select the number of data bits used by the device to which you want to connect. The default is **8**.

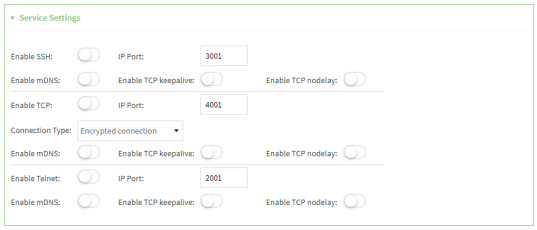
   c. **Parity**: For **Parity**, select the type of parity used by the device to which you want to connect. The default is **None**.

   d. **Stop bits**: For **Stop bits**, select the number of stop bits used by the device to which you want to connect. The default is **1**.

   e. **Flow control**: For **Flow control**, select the type of flow control used by the device to which you want to connect. The default is **None**.

7. Click **Enable** to enable the data framing feature.
8. For **Maximum Frame Count**, enter the maximum size of the packet. The default is **1024**.
9. For **Idle Time**, enter the length of time the device should wait before sending the packet.
10. For **End Pattern**, enter the end pattern. The packet is sent when this pattern is received from the serial port.
11. Click **Strip End Pattern** if you want to remove the end pattern from the packet before it is sent.
12. Expand **Service Settings**.

    All service settings are disabled by default. Click available options to toggle them to enabled, and set the IP ports as appropriate.



13. Expand **Autoconnect Settings**.

    a. Click **Enable** to enable the autoconnect feature.

    b. For **Connection Trigger**, select the option that describes the type of event that should trigger the connection.

    c. For **Outbound Connection Type**, select the option that describes the method used to initiate the connection.

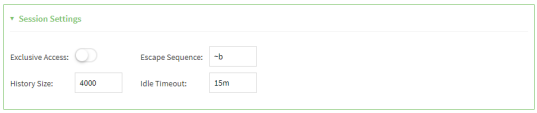
    d. For **Destination**, enter the host name or IP address of the remote server. When using SSH, this should be prefixed with the user name and followed by @.

    e. For **IP Port**, enter the TCP port of the remote server (1-65535).

    f. Click **Enable TCP keepalive** to enable TCP keepalive on the connection.

    g. Click **Enable TCP nodelay** to enable TCP nodealy on the connection.

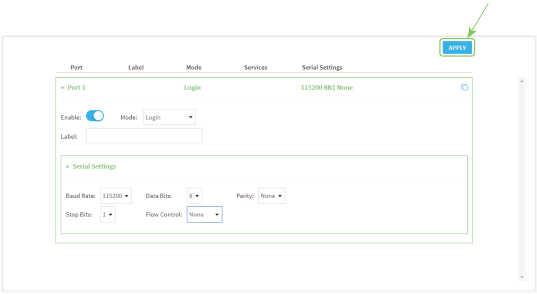
14. Expand **Session Settings**.



a. Enable **Exclusive access** to limit access to the serial port to a single active session.
b. For **Escape sequence**, type the characters used to start an escape sequence. If no characters are defined, the escape sequence is disabled. The default is **~b**.
c. For **History size**, type or select the number of bytes of output from the serial port that are written to buffer. These bytes are redisplayed when a user connects to the serial port. The default is **4000** bytes.
d. For **Idle timeout**, type the amount of time to wait before disconnecting due to user inactivity.

15. Expand **Monitor Settings**.
    a. Enable **CTS** to monitor CTS (Clear to Send) changes on this port.
    b. Enable **DCD** to monitor DCD (Data Carrier Detect) changes on this port.
16. Click **Apply** to save the configuration and apply the change.

    The **Apply** button is located at the top of the WebUI page. You may need to scroll to the top of the page to locate it.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. The serial port is enabled by default. To disable:

```
(config)> serial port1 enable false
(config)>
```

4. Set the mode:

```
(config)> serial port1 mode remoteaccess
(config)>
```

5. (Optional) Set a label that will be used when referring to this port.

```
(config)>path-paramlabel label
(config)>
```

6. Set the baud rate used by the device to which you want to connect:

```
(config)>path-parambaudrate rate
(config)>
```

7. Set the number of data bits used by the device to which you want to connect:

```
(config)>path-paramdatabits bits
(config)>
```

8. Set the type of parity used by the device to which you want to connect:

```
(config)>path-paramparity parity
(config)>
```

   Allowed values are:

   - **even**
   - **odd**
   - **none**

   The default is **none**.

9. Set the stop bits used by the device to which you want to connect:

```
(config)>path-paramstopbits bits
(config)>
```

10. Configure the session settings.
    a. Set the characters used to start an escape sequence:

```
(config)>path-paramescape string
(config)
```

       If no characters are defined, the escape sequence is disabled. The default is **~b**.

    b. Limit access to the serial port to a single active session:

```
(config)>path-paramexclusive true
(config)
```

    c. Set the number of bytes of output from the serial port that are written to buffer. These bytes are redisplayed when a user connects to the serial port.

```
(config)>path-paramhistory bytes
(config)
```

       The default is **4000** bytes.

d. Set the amount of time to wait before disconnecting due to user inactivity:

```
(config)>path-paramidle_timeout value
(config)
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **idle_timeout** to ten minutes, enter either **10m** or **600s**:

```
(config)>path-paramidle_timeout 600s
(config)
```

The default is **15m**.

11. Configure monitor settings.
    a. (Optional) Enable monitoring of CTS (Clear to Send) changes on this port:

```
(config)>path-parammonitor cts true
(config)
```

    b. (Optional) Enable monitoring of DCD (Data Carrier Detect) changes on this port:

```
(config)>path-parammonitor dcd true
(config)
```

12. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

13. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

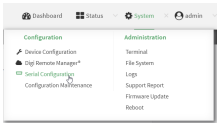
Configure Application mode

Application mode provides access to the serial device from Python applications.

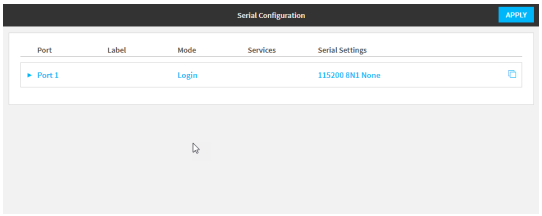
To change the configuration to match the serial configuration of the device to which you want to connect:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Configuration**, click **Serial Configuration**.

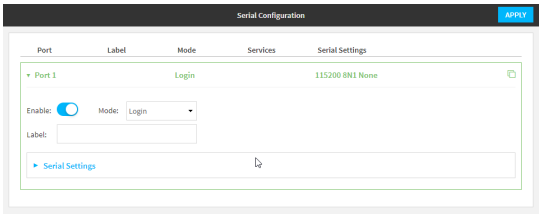


The **Serial Configuration** page is displayed.



Note You can also configure the serial port by using **Device Configuration** > **Serial**. Changes made by using either **Device Configuration** or **Serial Configuration** will be reflected in both.

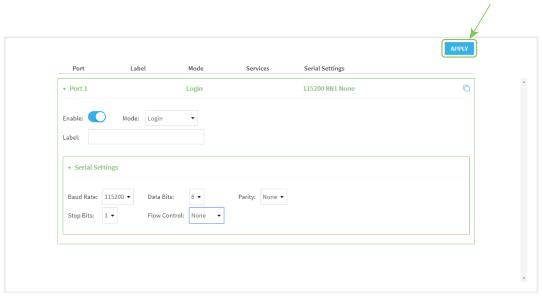
3. Click the name of the port that you want to configure.



The serial port is enabled by default. To disable, toggle off **Enable**.

4. For **Mode**, select **Application**. The default is **Remote**.
5. (Optional) For **Label**, enter a label that will be used when referring to this port.
6. Click **Apply** to save the configuration and apply the change.

   The **Apply** button is located at the top of the WebUI page. You may need to scroll to the top of the page to locate it.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. The serial port is enabled by default. To disable:

```
(config)> serial port1 enable false
(config)>
```

4. Set the mode:

```
(config)> serial port1 mode application
(config)>
```

5. (Optional) Set a label that will be used when referring to this port.

```
(config)>path-paramlabel label
(config)>
```

6. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

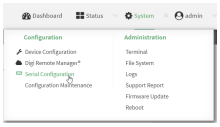
Configure UDP serial mode

The UDP serial mode option in the serial port configuration provides access to the serial port using UDP.

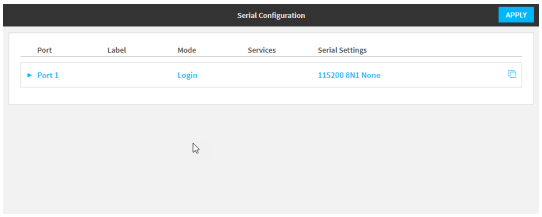
To change the configuration to match the serial configuration of the device to which you want to connect:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Configuration**, click **Serial Configuration**.

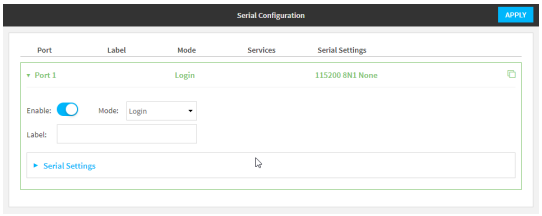


The **Serial Configuration** page is displayed.



Note You can also configure the serial port by using **Device Configuration** > **Serial**. Changes made by using either **Device Configuration** or **Serial Configuration** will be reflected in both.

3. Click to expand the port that you want to configure for UDP serial mode.



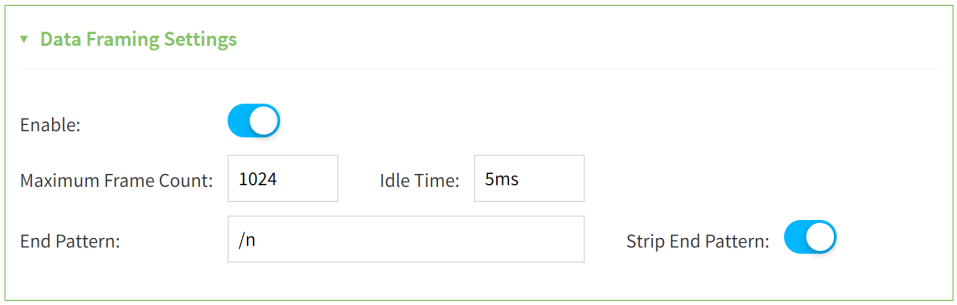
The serial port is enabled by default. To disable, toggle off **Enable**.

4. For **Mode**, select **UDP serial**.
   The default is **Remote**.
5. (Optional) For **Label**, enter a label that will be used when referring to this port.
6. Expand **Serial Settings**.



   a. For **Baud rate**, select the baud rate used by the device to which you want to connect.
   b. For **Data bits**, select the number of data bits used by the device to which you want to connect.
   c. For **Parity**, select the type of parity used by the device to which you want to connect.
   d. For **Stop bits**, select the number of stop bits used by the device to which you want to connect.
   e. For **Flow control**, select the type of flow control used by the device to which you want to connect.

7. Expand **Data Framing Settings**.

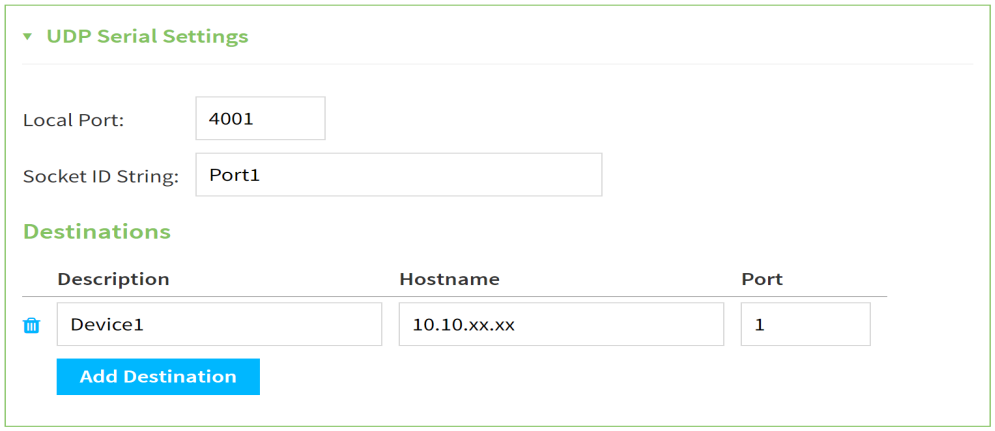


   a. Click **Enable** to enable the data framing feature.
   b. For **Maximum Frame Count**, enter the maximum size of the packet. The default is **1024**.
   c. For **Idle Time**, enter the length of time the device should wait before sending the packet.

d. For **End Pattern**, enter the end pattern. The packet is sent when this pattern is received from the serial port.

e. Click **Strip End Pattern** if you want to remove the end pattern from the packet before it is sent.

8. Expand **UDP Serial Settings**.



a. For **Local port**, enter the UDP port. The default is 4001 or serial port 1, 4002 for serial port 2, etc.

b. (Optional) For **Socket String ID**, enter a string that should be added at the beginning of each packet.

c. For **Destinations**, you can configure the remote sites to which you want to send data. If you do not specify any destinations, the IX15 send new data to the last hostname and port from which data was received. To add a destination:

i. Click **Add Destination**. A destination row is added.

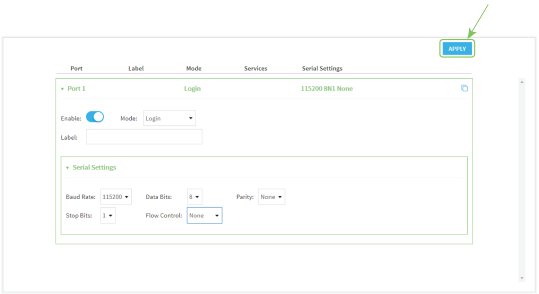
ii. (Optional) For **Description**, enter a description of the destination.

iii. For **Hostname**, enter the host name or IP address of the remote site to which data should be sent.

iv. For **Port**, enter the port number of the remote site to which data should be sent.

9. Click **Apply** to save the configuration and apply the change.

The **Apply** button is located at the top of the WebUI page. You may need to scroll to the top of the page to locate it.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. The serial port is enabled by default. To disable:

```
(config)> serial port1 enable false
(config)>
```

4. Set the mode:

```
(config)> serial port1 mode udp
(config)>
```

5. (Optional) Set a label that will be used when referring to this port.

```
(config)>serial port1 label label
(config)>
```

6. Set the baud rate used by the device to which you want to connect:

```
(config)>serial port1 label baudrate rate
(config)>
```

7. Set the number of data bits used by the device to which you want to connect:

```
(config)>serial port1 label databits bits
(config)>
```

8. Set the type of parity used by the device to which you want to connect:

```
(config)>serial port1 label parity parity
(config)>
```

   Allowed values are:

   - **even**
   - **odd**
   - **none**

   The default is **none**.
9. Set the stop bits used by the device to which you want to connect:

```
(config)>serial port1 label stopbits bits
(config)>
```

10. Set the type of flow control used by the device to which you want to connect:

```
(config)>serial port1 label flow type
(config)
```

Allowed values are:

- **none**
- **rts/cts**
- **xon/xoff**

The default is **none**.

11. Enable data framing:

```
(config)>serial port1 framing enable true
(config)
```

12. Set the maximum size of the packet:

```
(config)>serial port1 framing max_count int
(config)
```

The default is **1024**.

13. Set the length of time the device should wait before sending the packet:

```
(config)>serial port1 framing idle_time value
(config)
```

where *value* is in milliseconds (ms) or seconds (s). The maximum value is **60s**.

14. Set the end pattern. The packet is sent when this pattern is received from the serial port:

```
(config)>serial port1 framing end_pattern backslash-escaped-string
(config)
```

15. Set the strip end pattern if you want to remove the end pattern from the packet before it is sent:

```
(config)>serial port1 framing strip_pattern true
(config)
```

16. Set the UDP port:

```
(config)> serial port1 udp port port
(config)>
```

The default is **4001**.

17. (Optional) Enter a string that should be added at the beginning of each packet:

```
(config)> serial port1 udp socketid backslash-escaped-string
(config)>
```

18. Configure the remote sites to which you want to send data. If you do not specify any destinations, the IX15 send new data to the last hostname and port from which data was received. To add a destination:
    i. Add a destination:

```
(config)> add serial port1 upd destination end
(config serial port1 udp destination 0)>
```

    ii. (Optional) Enter a description of the destination:

```
(config serial port1 udp destination 0)> description string
(config serial port1 udp destination 0)>
```

    iii. Set the host name or IP address of the remote site to which data should be sent:

```
(config serial port1 udp destination 0)>hostname hostanme-or-IP-
address
(config serial port1 udp destination 0)>
```

    iv. Set the port number of the remote site to which data should be sent:

```
(config serial port1 udp destination 0)> port port
(config serial port1 udp destination 0)>
```

19. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

20. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

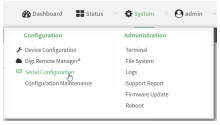
Configure Modbus mode

Modbus mode allows you to use the serial port for Modbus. See Modbus gateway.

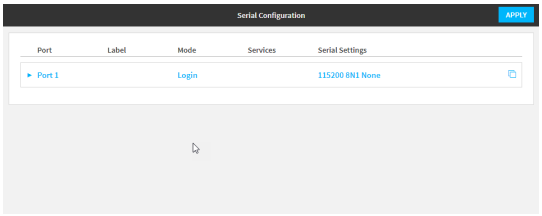
To change the configuration to match the serial configuration of the device to which you want to connect:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Configuration**, click **Serial Configuration**.

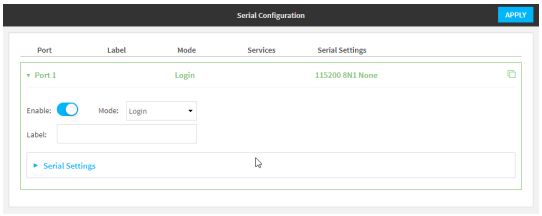


The **Serial Configuration** page is displayed.



Note You can also configure the serial port by using **Device Configuration** > **Serial**. Changes made by using either **Device Configuration** or **Serial Configuration** will be reflected in both.

3. Click the name of the port that you want to configure.



The serial port is enabled by default. To disable, toggle off **Enable**.

4. For **Mode**, select **Modbus**.
5. (Optional) For **Label**, enter a label that will be used when referring to this port.
6. Expand **Serial Settings**.

   The entries in the following fields must match the information for the power controller. Refer to your power controller manual for the correct entries.

   a. **Baud rate**: For **Baud rate**, select the baud rate used by the device to which you want to connect. The default is **9600**.

   b. **Data bits**: For **Data bits**, select the number of data bits used by the device to which you want to connect. The default is **8**.

   c. **Parity**: For **Parity**, select the type of parity used by the device to which you want to connect. The default is **None**.

   d. **Stop bits**: For **Stop bits**, select the number of stop bits used by the device to which you want to connect. The default is **1**.

   e. **Flow control**: For **Flow control**, select the type of flow control used by the device to which you want to connect. The default is **None**.

1. Set the baud rate used by the device to which you want to connect:

```
(config)>path-parambaudrate rate
(config)>
```

2. Set the number of data bits used by the device to which you want to connect:

```
(config)>path-paramdatabits bits
(config)>
```

3. Set the type of parity used by the device to which you want to connect:

```
(config)>path-paramparity parity
(config)>
```

Allowed values are:

- **even**
- **odd**
- **none**

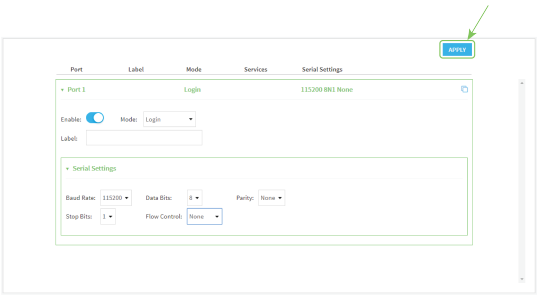
The default is **none**.

4. Set the stop bits used by the device to which you want to connect:

```
(config)>path-paramstopbits bits
(config)>
```

7. Click **Apply** to save the configuration and apply the change.

The **Apply** button is located at the top of the WebUI page. You may need to scroll to the top of the page to locate it.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. The serial port is enabled by default. To disable:

```
(config)> serial port1 enable false
(config)>
```

4. Set the mode:

```
(config)> serial port1 mode modbus
(config)>
```

5. (Optional) Set a label that will be used when referring to this port.

```
(config)>path-paramlabel label
(config)>
```

6. Expand **Serial Settings**.

   The entries in the following fields must match the information for the power controller. Refer to your power controller manual for the correct entries.

   a. **Baud rate**: For **Baud rate**, select the baud rate used by the device to which you want to connect. The default is **9600**.

   b. **Data bits**: For **Data bits**, select the number of data bits used by the device to which you want to connect. The default is **8**.

   c. **Parity**: For **Parity**, select the type of parity used by the device to which you want to connect. The default is **None**.

   d. **Stop bits**: For **Stop bits**, select the number of stop bits used by the device to which you want to connect. The default is **1**.

   e. **Flow control**: For **Flow control**, select the type of flow control used by the device to which you want to connect. The default is **None**.

1. Set the baud rate used by the device to which you want to connect:

```
(config)>path-parambaudrate rate
(config)>
```

2. Set the number of data bits used by the device to which you want to connect:

```
(config)>path-paramdatabits bits
(config)>
```

3. Set the type of parity used by the device to which you want to connect:

```
(config)>path-paramparity parity
(config)>
```

   Allowed values are:

   - **even**
   - **odd**
   - **none**

   The default is **none**.

4. Set the stop bits used by the device to which you want to connect:

```
(config)>path-paramstopbits bits
(config)>
```

7. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

8. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show serial status and statistics

To show the status and statistics for the serial port:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **Status**
3. Under **Connections**, click **Serial**.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Use the show serial command:

```
> show serial

Label     Port   Enable  Mode   Baudrate
--------  -----  ------  -----  --------
Serial 1  port1  true    login  9600
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Log serial port messages

To display and configure the serial port log:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **Status**
3. Under **Connections**, click **Serial**.
4. Click **Log**.

   The **Serial port log** window displays.
5. Click **Start** to start serial port logging.
6. Click **Stop** to stop serial port logging if it has been started.
7. Click **Refresh** to refresh the log display.
8. Click **Download** to download the serial port log.

9. (Optional) For **Log size**, configure the maximum allowed log size for the serial port log. The default is **65536**.

Routing

This chapter contains the following topics:

IP routing 233
Show the routing table 250
Dynamic DNS 251
Virtual Router Redundancy Protocol (VRRP) 257

IP routing

The IX15 device uses IP routes to decide where to send a packet it receives for a remote network. The process for deciding on a route to send the packet is as follows:

1. The device examines the destination IP address in the IP packet, and looks through the IP routing table to find a match for it.
2. If it finds a route for the destination, it forwards the IP packet to the configured IP gateway or interface.
3. If it cannot find a route for the destination, it uses a default route.
4. If there are two or more routes to a destination, the device uses the route with the longest mask.
5. If there are two or more routes to a destination with the same mask, the device uses the route with the lowest metric.

This section contains the following topics:

Configure a static route 234
Delete a static route 237
Policy-based routing 238
Configure a routing policy 239
Routing services 247
Configure routing services 247

Configure a static route

A static route is a manually configured routing entry. Information about the route is manually entered rather than obtained from dynamic routing traffic.

Required configuration items

- The destination address or network.
- The interface to use to reach the destination.

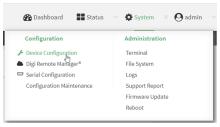
Additional configuration items

- A label used to identify this route.
- The IPv4 address of the gateway used to reach the destination.
- The metric for the route. When multiple routes are available to reach the same destination, the route with the lowest metric is used.
- The Maximum Transmission Units (MTU) of network packets using this route.

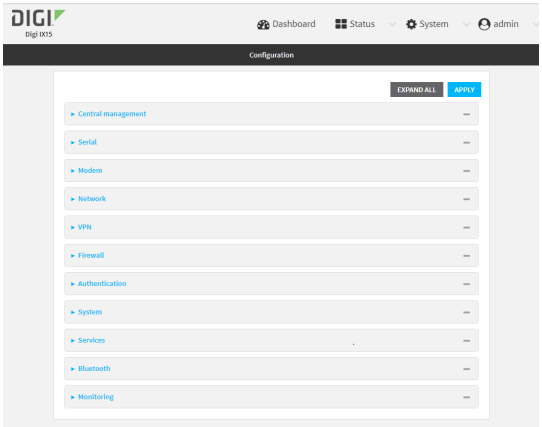
To configure a static route:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.

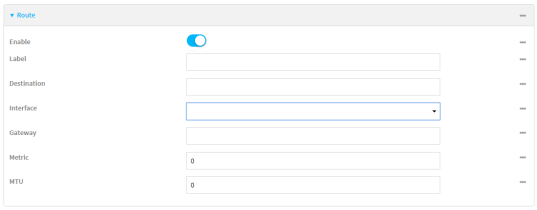


3. Click **Network** > **Routes** > **Static routes**.

4. Click the ✚ to add a new static route.



The new static route configuration page is displayed:



New static route configurations are enabled by default. To disable, click to toggle **Enable** to off.

5. (Optional) For **Label**, type a label that will be used to identify this route.
6. For **Destination**, type the IP address or network of the destination of this route.

   For example, to route traffic to the 192.168.47.0 network that uses a subnet mask of 255.255.255.0, type **192.168.47.0/24**. The **any** keyword can also be used to route packets to any destination with this static route.
7. For **Interface**, select the interface on the IX15 device that will be used with this static route.
8. (Optional) For **Gateway**, type the IPv4 address of the gateway used to reach the destination. Set to blank if the destination can be accessed without a gateway.
9. (Optional) For **Metric**, type the metric for the route. When multiple routes are available to reach the same destination, the route with the lowest metric is used.
10. (Optional) For **MTU**, type the Maximum Transmission Units (MTU) of network packets using this route.
11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a new static route:

```
(config)> add network route static end
(config network route static 0)>
```

   New static route instances are enabled by default. To disable:

```
(config network route static 0)> enable false
(config network route static 0)>
```

4. (Optional) set a label that will be used to identify this route. For example:

```
(config network route static 0)> label "route to accounting network"
(config network route static 0)>
```

5. Set the IP address or network of the destination of this route. For example:

```
(config network route static 0)> destination ip_address[/netmask]
(config network route static 0)>
```

   For example, to route traffic to the 192.168.47.0 network that uses a subnet mask of 255.255.255.0:

```
(config network route static 0)> dst 192.168.47.0/24
(config network route static 0)>
```

   The **any** keyword can also be used to route packets to any destination with this static route.

6. Set the interface on the IX15 device that will be used with this static route:
   a. Use the **?** to determine available interfaces:
   b. Set the interface. For example:

```
(config network route static 0)> interface /network/interface/eth1
(config network route static 0)>
```

7. (Optional) Set the IPv4 address of the gateway used to reach the destination. Set to blank if the destination can be accessed without a gateway.

```
(config network route static 0)> gateway IPv4_address
(config network route static 0)>
```

8. (Optional) Set the metric for the route. When multiple routes are available to reach the same destination, the route with the lowest metric is used.

```
(config network route static 0)> metric value
(config network route static 0)>
```

   where *value* is an interger between **0** and **65535**. The default is **0**.

9. (Optional) Set the Maximum Transmission Units (MTU) of network packets using this route:

```
(config network route static 0)> mtu integer
(config network route static 0)>
```

10. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

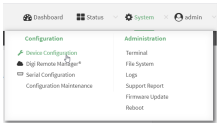
11. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

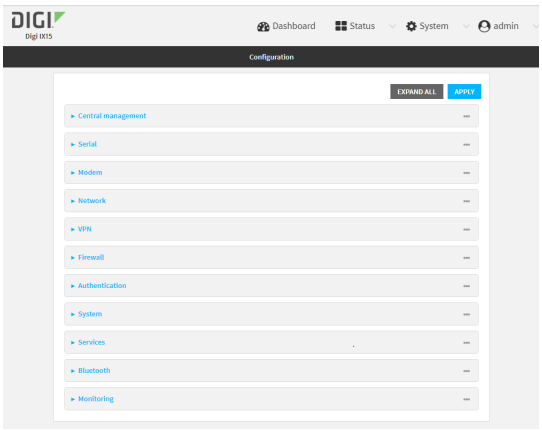
Delete a static route

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Routes** > **Static routes**.
4. Click the menu icon (**...**) for a static route and select **Delete**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Determine the index number of the static route to be deleted:

```
(config)> show network route static
0
    dst 10.0.0.1
    enable true
    no gateway
    interface /network/interface/lan1
    label new_static_route
    metric 0
    mtu 0
1
    dst 192.168.5.1
    enable true
    gateway 192.168.5.1
    interface /network/interface/lan2
    label new_static_route_1
    metric 0
    mtu 0
(config)>
```

4. Use the index number to delete the static route:

```
(config)> del network route static 0
(config)>
```

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Policy-based routing

Normally, a routing device determines how to route a network packet based on its destination address. However, you can use policy-based routing to forward the packet based on other criteria, such as the source of the packet. For example, you can configure the IX15 device so that high-priority traffic is routed through the cellular connection, while all other traffic is routed through an Ethernet (WAN) connection.

Policy-based routing for the IX15 device uses the following criteria to determine how to route traffic:

- Firewall zone (for example, internal/outbound traffic, external/inbound traffic, or IPSec tunnel traffic).
- Network interface (for example, the cellular connection, the WAN, or the LAN).
- IPv4 address.
- IPv6 address.
- MAC address.
- Domain.
- Protocol type (TCP, UDP, ICMP, or all).

The order of the policies is important. Routing policies are processed sequentially; as a result, if a packet matches an earlier policy, it will be routed using that policy's rules. It will not be processed by any subsequent rules.

Configure a routing policy

Required configuration items

- The packet matching parameters. It can any combination of the following:
  - Source interface.
  - Source address. This can be a firewall zone, an interface, a single IPv4/IPv6 address or network, or a MAC address.
  - Destination address. This can be a firewall zone, an interface, a single IPv4/IPv6 address or network, or a domain.
  - Protocol. This can be **any**, **tcp**, **udp** or **icmp**.
  - Source port. This is only used if the protocol is set to **tcp** or **udp**.
  - Destination port. This is only used if protocol is set to **tcp** or **udp**.
- The network interface used to reach the destination.

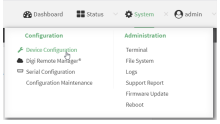
Additional configuration items

- A label for the routing policy.
- Whether packets that match this policy should be dropped when the gateway interface is disconnected, rather than forwarded through other interfaces.

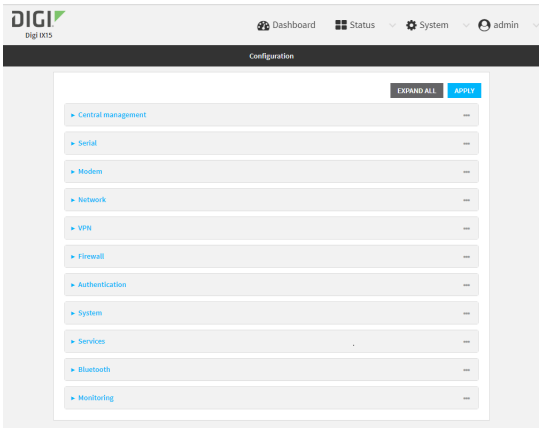
To configure a routing policy:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Routes** > **Policy-based routing**.
4. Click the ✚ to add a new route policy.



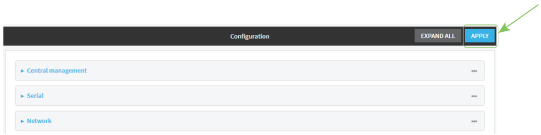
The new route policy page is displayed:

New route policies are enabled by default. To disable, click to toggle **Enable** to off.

5. (Optional) For **Label**, type a label that will be used to identify this route policy.
6. For **Interface**, select the interface on the IX15 device that will be used with this route policy.
7. (Optional) Enable **Exclusive** to configure the policy to drop packets that match the policy when the gateway interface is disconnected, rather than forwarded through other interfaces.
8. For **IP version**, select **Any**, **IPv4**, or **IPv6**.
9. For **Protocol**, select **Any**, **TCP**, **UDP**, or **ICMP**.
    - If **TCP** or **UDP** is selected for **Protocol**, type the port numbers of the **Source port** and **Destination port**, or set to **any** to match for any port.
    - If **ICMP** is selected for **Protocol**, type the ICMP type and optional code, or set to **any** to match for any ICMP type.
10. For **DSCP**, type the 6-bit hexadecimal Differentiated Services Code Point (DSCP) field match criteria. This will match packets based on the DHCP field within the ToS field of the IP header.
11. Configure source address information:
    a. Click to expand **Source address**.
    b. For **Type**, select one of the following:
        - **Zone**: Matches the source IP address to the selected firewall zone. See Firewall configuration for more information about firewall zones.
        - **Interface**: Matches the source IP address to the selected interface's network address.

- **IPv4 address**: Matches the source IP address to the specified IP address or network. Use the format ***IPv4_address***[**/*netmask***], or use **any** to match any IPv4 address.
- **IPv6 address**: Matches the source IP address to the specified IP address or network. Use the format ***IPv6_address*[/*prefix_length*]**, or use **any** to match any IPv6 address.
- **MAC address**: Matches the source MAC address to the specified MAC address.

12. Configure the destination address information:
    a. Click to expand **Destination address**.
    b. For **Type**, select one of the following:
        - **Zone**: Matches the destination IP address to the selected firewall zone. See Firewall configuration for more information about firewall zones.
        - **Interface**: Matches the destination IP address to the selected interface's network address.
        - **IPv4 address**: Matches the destination IP address to the specified IP address or network. Use the format *IPv4_address*/[*netmask*], or use **any** to match any IPv4 address.
        - **IPv6 address**: Matches the destination IP address to the specified IP address or network. Use the format *IPv6_address*/[*prefix_length*], or use **any** to match any IPv6 address.
        - **Domain**: Matches the destination IP address to the specified domain names. To specify domains:
            i. Click to expand **Domains**.
            ii. Click the ✚ to add a domain.
            iii. For **Domain**, type the domain name.
            iv. Repeat to add additional domains.
        - **Default route**: Matches packets destined for the default route, excluding routes for local networks.
13. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a new routing policy:

```
(config)> add network route policy end
(config network route policy 0)>
```

   New route policies are enabled by default. To disable:

```
(config network route policy 0)> enable false
(config network route policy 0)>
```

4. (Optional) Set the label that will be used to identify this route policy:

```
(config network route policy 0)> label "New route policy"
(config network route policy 0)>
```

5. Set the interface on the IX15 device that will be used with this route policy:
   a. Use the **?** to determine available interfaces:
   b. Set the interface. For example:

```
(config network route policy 0)> interface /network/interface/eth1
(config network route policy 0)>
```

6. (Optional) Enable **exclusive** to configure the policy to drop packets that match the policy when the gateway interface is disconnected, rather than forwarded through other interfaces:

```
(config network route policy 0)> exclusive true
(config network route policy 0)>
```

7. Select the IP version:

```
(config network route policy 0)> ip_version value
(config network route policy 0)>
```

   where *value* is one of **any**, **ipv4**, or **ipv6**.

8. Set the protocol:

```
(config network route policy 0)> protocol value
(config network route policy 0)>
```

   where *value* is one of:

   - **any**: All protocols are matched.
   - **tcp**: Source and destination ports are matched:
     a. Set the source port:

```
(config network route policy 0)> src_port value
(config network route policy 0)>
```

        where *value* is the port number, or the keyword **any** to match any port as the source port.

b. Set the destination port:

```
(config network route policy 0)> dst_port value
(config network route policy 0)>
```

where *value* is the port number, or the keyword **any** to match any port as the destination port.

- **upd**: Source and destination ports are matched:
  a. Set the source port:

```
(config network route policy 0)> src_port value
(config network route policy 0)>
```

where *value* is the port number, or the keyword **any** to match any port as the source port.

b. Set the destination port:

```
(config network route policy 0)> dst_port value
(config network route policy 0)>
```

where *value* is the port number, or the keyword **any** to match any port as the destination port.

- **icmp**: The ICMP protocol is matched. Identify the ICMP type:

```
(config network route policy 0)> icmp_type value
(config network route policy 0)>
```

where *value* is the ICMP type and optional code, or set to **any** to match for any ICMP type.

9. Set the source address type:

```
(config network route policy 0)> src type value
(config network route policy 0)>
```

where *value* is one of:

- **zone**: Matches the source IP address to the selected firewall zone. Set the zone:
  a. Use the **?** to determine available zones:

```
(config network route policy 0)> src zone ?

Zone: Match the IP address to the specified firewall zone.
Format:
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup
```

```
Default value: any
Current value: any

(config network route policy 0)> src zone
```

b. Set the zone. For example:

```
(config network route policy 0)> src zone external
(config network route policy 0)>
```

See Firewall configuration for more information about firewall zones.

- **interface**: Matches the source IP address to the selected interface's network address. Set the interface:
  a. Use the **?** to determine available interfaces:
  b. Set the interface. For example:

```
(config network route policy 0)> src interface
/network/interface/eth1
(config network route policy 0)>
```

- **address**: Matches the source IPv4 address to the specified IP address or network. Set the address that will be matched:

```
(config network route policy 0)> src address value
(config network route policy 0)>
```

where value uses the format ***IPv4_address***[**/*netmask***], or **any** to match any IPv4 address.

- **address6**: Matches the source IPv6 address to the specified IP address or network. Set the address that will be matched:

```
(config network route policy 0)> src address6 value
(config network route policy 0)>
```

where value uses the format ***IPv6_address***[**/*prefix_length***], or **any** to match any IPv6 address.

- **mac**: Matches the source MAC address to the specified MAC address. Set the MAC address to be matched:

```
(config network route policy 0)> src mac MAC_address
(config network route policy 0)>
```

10. Set the destination address type:

```
(config network route policy 0)> dst type value
(config network route policy 0)>
```

where *value* is one of:

- **zone**: Matches the destination IP address to the selected firewall zone. Set the zone:
  a. Use the **?** to determine available zones:

```
(config network route policy 0)> dst zone ?

Zone: Match the IP address to the specified firewall zone.
Format:
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

Default value: any
Current value: any

(config network route policy 0)> dst zone
```

  b. Set the zone. For example:

```
(config network route policy 0)> dst zone external
(config network route policy 0)>
```

  See Firewall configuration for more information about firewall zones.
- **interface**: Matches the destination IP address to the selected interface's network address. Set the interface:
  a. Use the **?** to determine available interfaces:
  b. Set the interface. For example:

```
(config network route policy 0)> dst interface
/network/interface/eth1
(config network route policy 0)>
```

- **address**: Matches the destination IPv4 address to the specified IP address or network. Set the address that will be matched:

```
(config network route policy 0)> dst address value
(config network route policy 0)>
```

  where value uses the format ***IPv4_address***[**/*netmask***], or **any** to match any IPv4 address.
- **address6**: Matches the destination IPv6 address to the specified IP address or network. Set the address that will be matched:

```
(config network route policy 0)> dst address6 value
(config network route policy 0)>
```

where value uses the format ***IPv6_address***[**/*prefix_length***], or **any** to match any IPv6 address.

- **mac**: Matches the destination MAC address to the specified MAC address. Set the MAC address to be matched:

```
(config network route policy 0)> dst mac MAC_address
(config network route policy 0)>
```

11. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

12. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Routing services

Your IX15 includes support for dynamic routing services and protocols. The following routing services are supported:

Service or protocol	Information
RIP	The IPv4 Routing Information Protocol (RIP) service supports RIPv2 (RFC2453) and RIPv1 (RFC1058).
RIPng	The IPv6 Routing Information Protocol (RIP) service supports RIPng (RFC2080).
OSPFv2	The IPv4 Open Shortest Path First (OSPF) service supports OSPFv2 (RFC2328).
OSPFv3	The IPv6 Open Shortest Path First (OSPF) service supports OSPFv3 (RFC2740).
BGP	The Border Gateway Protocol (BGP) service supports BGP-4 (RFC1771).
IS-IS	The IPv4 and IPv6 Intermediate System to Intermediate System (IS-IS) service.

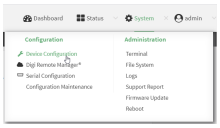
Configure routing services

Required configuration items

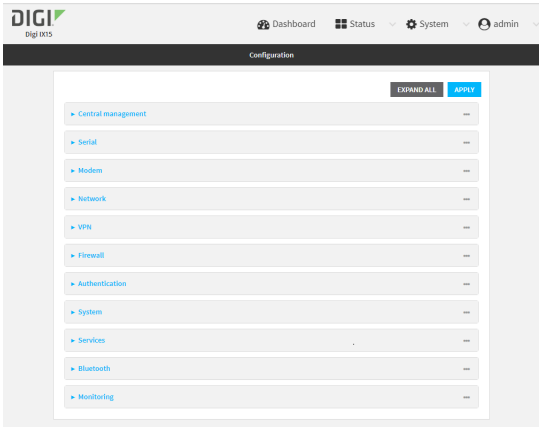
- Enable routing services.
- Enable and configure the types of routing services that will be used.

WebUI

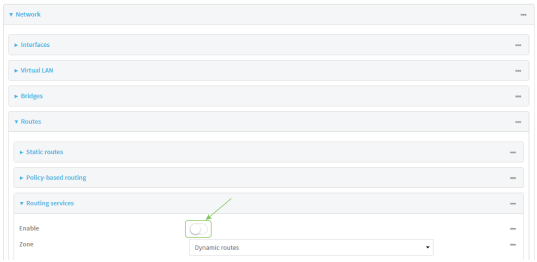
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Routes** > **Routing services**.
4. Click **Enable**.



The default firewall zone setting, **Dynamic routes**, is specifically designed to work with routing services and should be left as the default.

5. Configure the routing services that will be used:
   a. Click to expand a routing service.
   b. **Enable** the routing service.
   c. Complete the configuration of the routing service.
6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable routing services:

```
(config)> network route service enable true
(config)>
```

4. Configure routing services that will be used:
   a. Use the **?** to display available routing services:

```
(config)> network route service ?

Routing services: Settings for dynamic routing services and protocols.

 Parameters                Current Value
 ------------------------------------------------------------------------
----------
 enable                    true            Enable
 zone                      dynamic_routes  Zone

 Additional Configuration
 ------------------------------------------------------------------------
----------
 bgp                       BGP
 isis                      IS-IS
 ospfv2                    OSPFv2
 ospfv3                    OSPFv3
 rip                       RIP
 ripng                     RIPng

(config)>
```

   b. Enable a routing service that will be used. For example, to enable the RIP service:

```
(config)> network route service rip enable true
(config)>
```

   c. Complete the configuration of the routing service. For example, use the **?** to view the available parameters for the RIP service:

```
(config)> network route service rip ?

 Parameters                Current Value
```

```
 ------------------------------------------------------------------------
----------
 ecmp                        false          Allow ECMP
 enable                      true           Enable

 Additional Configuration
 ------------------------------------------------------------------------
----------
 interface                   Interfaces
 neighbour                   Neighbours
 redis                       Route redistribution
 timer                       Timers

(config)>
```

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

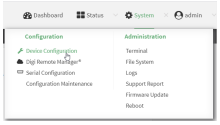
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show the routing table

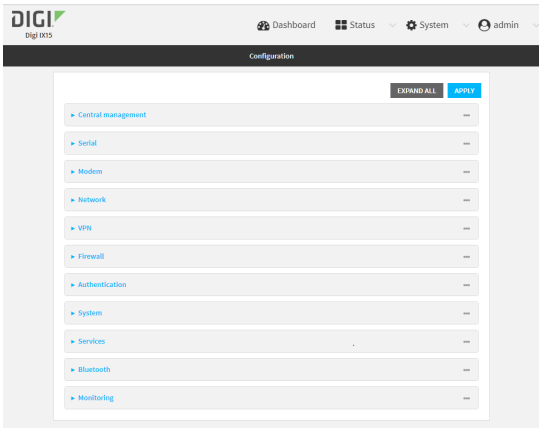
To display the routing table:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Status** > **Routes**.

   The **Network Routing** window is displayed.
4. Click **IPv4 Load Balance** to view IPv4 load balancing.
5. Click **IPv6 Load Balance** to view IPv6 load balancing.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the Admin CLI prompt, type show route:

   You can limit the display to only IPv4 entries by using **show route ipv4**, or to IPv6 entries by using **show route ipv6**. You can also display more information by adding the **verbose** option to the **show route** and **show route *ip_type*** commands.
3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Dynamic DNS

The Domain Name System (DNS) uses name servers to provide a mapping between computer-readable IP addresses and human-readable hostnames. This allows users to access websites and personal networks with easy-to-remember URLs. Unfortunately, IP addresses change frequently, invalidating these mappings when they do. Dynamic DNS has become the standard method of addressing this problem, allowing devices to update name servers with their new IP addresses.

By providing the IX15 device with the domain name and credentials obtained from a dynamic DNS provider, the router can automatically update the remote nameserver whenever your WAN or public IP address changes.

Your IX15 device supports a number of Dynamic DNS providers as well as the ability to provide a custom provider that is not included on the list of providers.

Configure dynamic DNS

This section describes how to cofigure dynamic DNS on a IX15 device.

Required configuration items

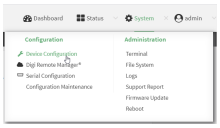
- Add a new Dynamic DNS service.
- The interface that has its IP address registered with the Dynamic DNS provider.
- The name of a Dynamic DNS provider.
- The domain name that is linked to the interface's IP address.
- The username and password to authenticate with the Dynamic DNS provider.

Additional configuration items

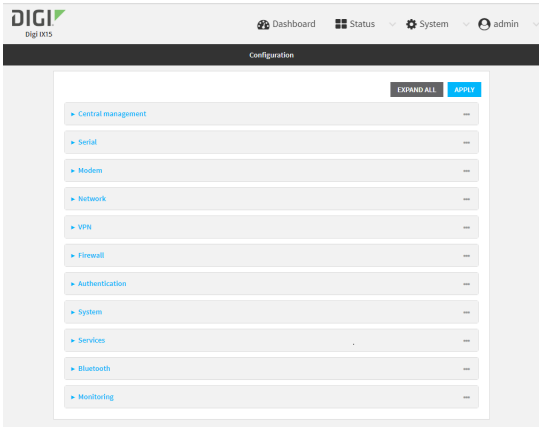
- If the Dynamic DNS service provider is set to **custom**, identify the URL that should be used to update the IP address with the Dynamic DNS provider.
- The amount of time to wait to check if the interface's IP address needs to be updated.
- The amount of time to wait to force an update of the interface's IP address.
- The amount of time to wait for an IP address update to succeed before retrying the update.
- The number of times to retry a failed IP address update.

WebUI

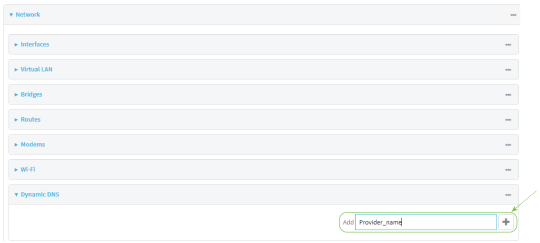
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



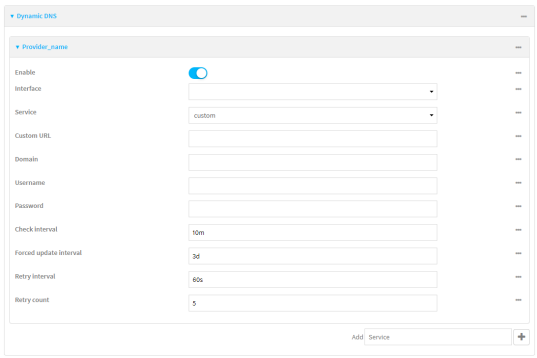
The **Configuration** window is displayed.



3. Click **Network** > **Dynamic DNS**.
4. Type a name for this Dynamic DNS instance in **Add Service** and click ✚.



The Dynamic DNS configuration page displays.



New Dynamic DNS configurations are enabled by default. To disable, click to toggle **Enable** to off.

5. For **Interface**, select the interface that has its IP address registered with the Dynamic DNS provider.
6. For **Service**, select the Dynamic DNS provider, or select **custom** to enter a custom URL for the Dynamic DNS provider.
7. If **custom** is selected for **Service**, type the **Custom URL** that should be used to update the IP address with the Dynamic DNS provider.
8. Type the **Domain** name that is linked to the interface's IP address.
9. Type the **Username** and **Password** used to authenticate with the Dynamic DNS provider.
10. (Optional) For **Check Interval**, type the amount of time to wait to check if the interface's IP address needs to be updated.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Check interval** to ten minutes, enter **10m** or **600s**.
11. (Optional) For **Forced update interval**, type the amount of time to wait to force an update of the interface's IP address.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Forced update interval** to ten minutes, enter **10m** or **600s**.

    The setting for **Forced update interval** must be larger than the setting for **Check Interval**.
12. (Optional) For **Retry interval**, type the amount of time to wait for an IP address update to succeed before retrying the update.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Retry interval** to ten minutes, enter **10m** or **600s**.
13. (Optional) For **Retry count**, type the number of times to retry a failed IP address update.
14. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a new Dynamic DNS instance. For example, to add an instance named **new_ddns_instance**:

```
(config)> add network ddns new_ddns_instance
(config network ddns new_ddns_instance)>
```

New Dynamic DNS instances are enabled by default. To disable:

```
(config network ddns new_ddns_instance)> enable false
(config network ddns new_ddns_instance)>
```

4. Set the interface for the Dynamic DNS instance:
   a. Use the **?** to determine available interfaces:
   b. Set the interface. For example:

```
(config network ddns new_ddns_instance)> interface eth1
(config network ddns new_ddns_instance)>
```

5. Set the Dynamic DNS provider service:
   a. Use the **?** to determine available services:

```
(config network ddns new_ddns_instance)> service ?

Service: The provider of the dynamic DNS service.
Format:
  custom
  3322.org
  changeip.com
  ddns.com.br
  dnsdynamic.org
  ...

Default value: custom
Current value: custom

(config network ddns new_ddns_instance)> service
```

   b. Set the service:

```
(config network ddns new_ddns_instance)> service service_name
(config network ddns new_ddns_instance)>
```

6. If **custom** is configured for **service**, set the custom URL that should be used to update the IP address with the Dynamic DNS provider:

```
(config network ddns new_ddns_instance)> custom url
(config network ddns new_ddns_instance)>
```

7. Set the domain name that is linked to the interface's IP address:

```
(config network ddns new_ddns_instance)> domain domain_name
(config network ddns new_ddns_instance)>
```

8. Set the username to authenticate with the Dynamic DNS provider:

```
(config network ddns new_ddns_instance)> username name
(config network ddns new_ddns_instance)>
```

9. Set the password to authenticate with the Dynamic DNS provider:

```
(config network ddns new_ddns_instance)> password pwd
(config network ddns new_ddns_instance)>
```

10. (Optional) Set the amount of time to wait to check if the interface's IP address needs to be updated:

```
(config network ddns new_ddns_instance)> check_interval value
(config network ddns new_ddns_instance)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **check_interval** to ten minutes, enter either **10m** or **600s**:

```
(config network ddns new_ddns_instance)> check_interval 600s
(config network ddns new_ddns_instance)>
```

The default is **10m**.

11. (Optional) Set the amount of time to wait to force an update of the interface's IP address:

```
(config network ddns new_ddns_instance)> force_interval value
(config network ddns new_ddns_instance)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **force_interval** to ten minutes, enter either **10m** or **600s**:

```
(config network ddns new_ddns_instance)> force_interval 600s
(config network ddns new_ddns_instance)>
```

The default is **3d**.

12. (Optional) Set the amount of time to wait for an IP address update to succeed before retrying the update:

```
(config network ddns new_ddns_instance)> retry_interval value
(config network ddns new_ddns_instance)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **retry_interval** to ten minutes, enter either **10m** or **600s**:

```
(config network ddns new_ddns_instance)> retry_interval 600s
(config network ddns new_ddns_instance)>
```

The default is **60s**.

13. (Optional) Set the number of times to retry a failed IP address update:

```
(config network ddns new_ddns_instance)> retry_count value
(config network ddns new_ddns_instance)>
```

where *value* is any interger. The default is **5**.

14. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

15. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Virtual Router Redundancy Protocol (VRRP)

Virtual Router Redundancy Protocol (VRRP) is a standard for gateway device redundancy and failover that creates a "virtual router" with a floating IP address. Devices connected to the LAN then use this virtual router as their default gateway. Responsibility for the virtual router is assigned to one of the VRRP-enabled devices on a LAN (the "master router"), and this responsibility transparently fails over to backup VRRP devices if the master router fails. This prevents the default gateway from being a single point of failure, without requiring configuration of dynamic routing or router discovery protocols on every host.

Multiple IX15 devices can be configured as VRRP devices and assigned a priority. The router with the highest priority will be used as the master router. If the master router fails, then the IP address of the virtual router is mapped to the backup device with the next highest priority. Each VRRP router is configured with a unique LAN IP address, and the same shared VRRP address.

VRRP+

VRRP+ is an extension to the VRRP standard that uses network probing to monitor connections through VRRP-enabled devices and can dynamically change the priority of the devices, including changing devices from master to backup, and from backup to master, even if the device has not failed. For example, if a host becomes unreachable on the far end of a network link, then the physical default gateway can be changed by adjusting the VRRP priority of the Digi IX15 Gateway device connected to the failing link. This provides failover capabilities based on the status of connections behind the router, in addition to the basic VRRP device failover. For IX15 devices, SureLink is used to probe network connections.

VRRP+ can be configured to probe a specified IP address by either sending an ICMP echo request (ping) or attempting to open a TCP socket to the IP address.

Configure VRRP

This section describes how to configure VRRP on a IX15 device.

Required configuration items

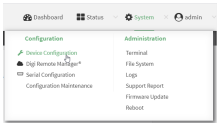
- Enable VRRP.
- The interface used by VRRP.

- The Router ID that identifies the virtual router instance. The Router ID must be the same on all VRRP devices that participate in the same VRRP device pool.
- The VRRP priority of this device.
- The shared virtual IP address for the VRRP virtual router. Devices connected to the LAN will use this virtual IP address as their default gateway.

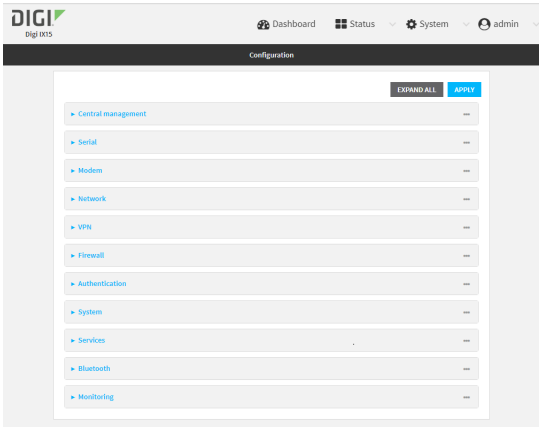
See Configure VRRP+ for information about configuring VRRP+, an extension to VRRP that uses network probing to monitor connections through VRRP-enabled devices and dynamically change the VRRP priorty of devices based on the status of their network connectivity.

WebUI

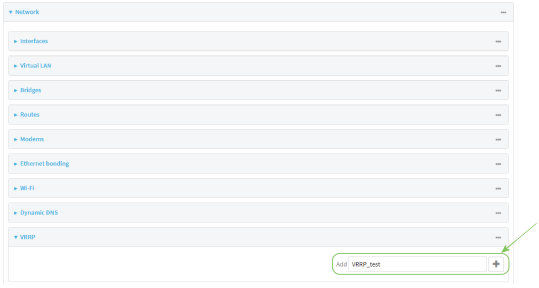
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



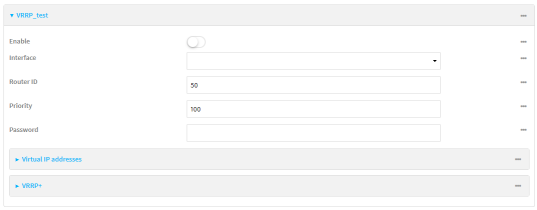
The **Configuration** window is displayed.



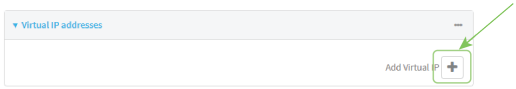
3. Click **Network** > **VRRP**.
4. For **Add VRRP instance**, type a name for the VRRP instance and click ✚.



The new VRRP instance configuration is displayed.



5. Click **Enable**.
6. For **Interface**, select the interface on which this VRRP instance should run.
7. For **Router ID** field, type the ID of the virtual router instance. The Router ID must be the same on all VRRP devices that participate in the same VRRP device pool. Allowed values are from **1** and **255**, and it is configured to **50** by default.
8. For **Priority**, type the priority for this router in the group. The router with the highest priority will be used as the master router. If the master router fails, then the IP address of the virtual router is mapped to the backup device with the next highest priority. If this device's actual IP address is being used as the virtual IP address of the VRRP pool, then the priority of this device should be set to **255** . Allowed values are from **1** and **255**, and it is configured to **100** by default.
9. (Optional) For **Password**, type a password that will be used to authenticate this VRRP router with VRRP peers. If the password length exceeds 8 characters, it will be truncated to 8 characters.
10. Configure the virtual IP addresses associated with this VRRP instance:
    a. Click to expand **Virtual IP addresses**.
    b. Click ✚ to add a virtual IP address.



    c. For **Virtual IP**, type the IPv4 or IPv6 address for a virtual IP of this VRRP instance.
    d. (Optional) Repeat to add additional virtual IPs.
11. See Configure VRRP+ for information about configuring VRRP+.
12. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a VRRP instance. For example:

```
(config)> add network vrrp VRRP_test
(config network vrrp VRRP_test)>
```

4. Enable the VRRP instance:

```
(config network vrrp VRRP_test)> enable true
(config network vrrp VRRP_test)>
```

5. Set the interface on which this VRRP instance should run:
   a. Use the **?** to determine available interfaces:
   b. Set the interface, for example:

```
(config network vrrp VRRP_test)> interface /network/interface/eth
(config network vrrp VRRP_test)>
```

   c. Repeat for additional interfaces.
6. Set the router ID. The Router ID must be the same on all VRRP devices that participate in the same VRRP device pool. Allowed values are from **1** and **255**, and it is configured to **50** by default.

```
(config network vrrp VRRP_test)> router_id int
(config network vrrp VRRP_test)>
```

7. Set the priority for this router in the group. The router with the highest priority will be used as the master router. If the master router fails, then the IP address of the virtual router is mapped to the backup device with the next highest priority. If this device's actual IP address is being used as the virtual IP address of the VRRP pool, then the priority of this device should be set to **255** . Allowed values are from **1** and **255**, and it is configured to **100** by default.

```
(config network vrrp VRRP_test)> priority int
(config network vrrp VRRP_test)>
```

8. (Optional) Set a password that will be used to authenticate this VRRP router with VRRP peers. If the password length exceeds 8 characters, it will be truncated to 8 characters.

```
(config network vrrp VRRP_test)> password pwd
(config network vrrp VRRP_test)>
```

9. Add a virtual IP address associated with this VRRP instance. This can be an IPv4 or IPv6 address.

```
(config network vrrp VRRP_test)> add virtual_address end ip_address
(config network vrrp VRRP_test)>
```

   Additional virtual IP addresses can be added by repeating this step with different values for *ip_address*.

10. Save the configuration and apply the change:

```
(config network vrrp new_vrrp_instance)> save
Configuration saved.
>
```

11. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure VRRP+

VRRP+ is an extension to the VRRP standard that uses SureLink network probing to monitor connections through VRRP-enabled devices and adjust devices' VRRP priority based on the status of the SureLink tests.

This section describes how to configure VRRP+ on a IX15 device.

Required configuration items

- Both master and backup devices:
  - A configured and enabled instance of VRRP. See Configure VRRP for information.
  - Enable VRRP+.
  - WAN interfaces to be monitored by using VRRP+.

    Note SureLink is enabled by default on all WAN interfaces, and should not be disabled on the WAN interfaces that are being monitored by VRRP+.

    If multiple WAN interfaces are being monitored on the same device, the VRRP priority will be adjusted only if all WAN interfaces fail SureLink tests.

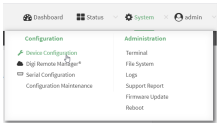
  - The amount that the VRRP priority will be modified when SureLink determines that the VRRP interface is not functioning correctly.
  - Configure the VRRP interface's DHCP server to use a custom gateway that corresponds to one of the VRRP virtual IP addresses.
- Backup devices only:
  - Enable and configure SureLink on the VRRP interface.
  - Set the IP gateway to the IP address of the VRRP interface on the master device.

Additional configuration items

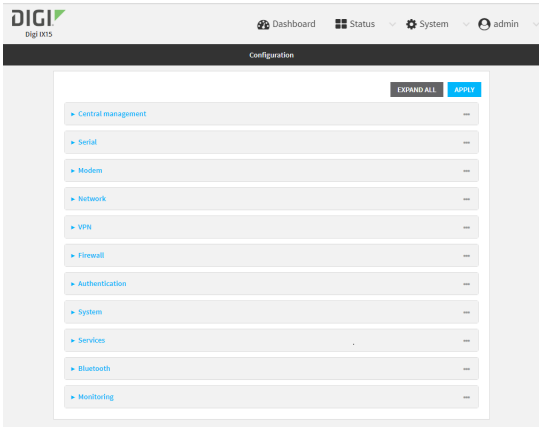
- For backup VRRP devices, enable the ability to monitor the VRRP master, so that a backup device can increase its priority when the master device fails SureLink tests.

WebUI

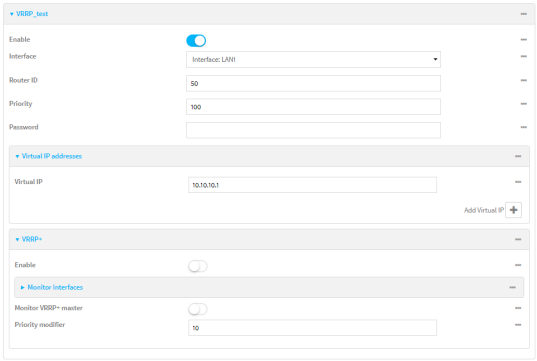
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



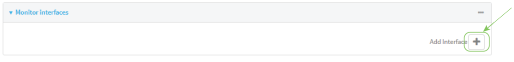
The **Configuration** window is displayed.



3. Click **Network** > **VRRP**.
4. Create a new VRRP instance, or click to expand an existing VRRP instance.
   See Configure VRRP for information about creating a new VRRP instance.
5. Click to expand **VRRP+**.



6. Click **Enable**.
7. Add interfaces to monitor:
   a. Click to expand **Monitor interfaces**.
   b. Click ✚ to add an interface for monitoring.



   c. For **Interface**, select the local interface to monitor. Generally, this will be a cellular or WAN

interface.

d. (Optional) Click ✚ again to add additional interfaces.

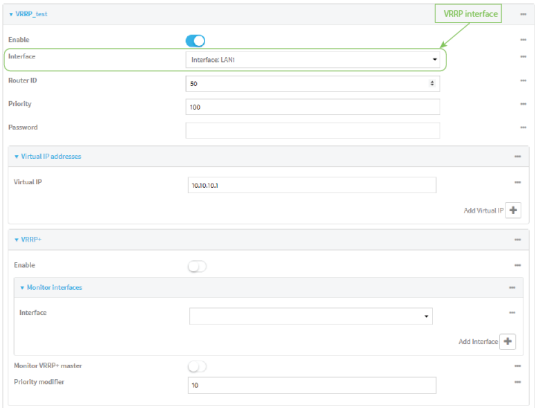
8. (Optional) For backup devices, click to enable **Monitor VRRP+ master**.

   This parameter allows a backup VRRP device to monitor the master device, and increase its priority when the master device is failing SureLink tests. This can allow a device functioning as a backup device to promote itself to master.

9. For **Priority modifier**, type or select the amount that the device's priority should be decreased due to SureLink connectivity failure, and increased when SureLink succeeds again.

   Along with the priority settings for devices in this VRRP pool, the amount entered here should be large enough to automatically demote a master device when SureLink connectivity fails. For example, if the VRRP master device has a priority of **100** and the backup device has a priority of **80**, then the **Priority modifier** should be set to an amount greater than **20** so that if SureLink fails on the master, it will lower its priority to below **80**, and the backup device will assume the master role.

10. Configure the VRRP interface. The VRRP interface is defined in the **Interface** parameter of the VRRP configuration, and generally should be a LAN interface:

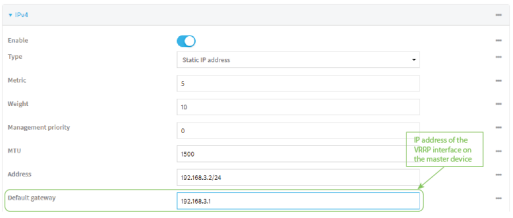


   To configure the VRRP interface:

   a. Click to expand **Network** > **Interfaces**.

   b. Click to expand the appropriate VRRP interface (for example, **LAN1**).

   c. For backup devices, for **Default Gateway**, type the IP address of the VRRP interface on the master device.

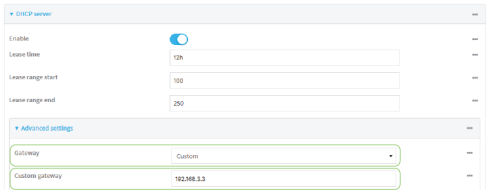


   d. Configure the VRRP interface's DHCP server to use a custom gateway that corresponds to one of the VRRP virtual IP addresses:

      i. Click to expand **DHCP Server** > **Advanced settings**.

      ii. For **Gateway**, select **Custom**.

iii. For **Custom gateway**, enter the IP address of one of the virtual IPs used by this VRRP instance.



e. For backup devices, enable and configure SureLink on the VRRP interface. Generally, this should be a LAN interface; VRRP+ will then monitor the LAN using SureLink to determine if the interface has network connectivity and promote a backup to master if SureLink fails.

i. Click to expand **IPv4** > **SureLink**.

ii. Click **Enable**.

iii. For **Interval**, type a the amount of time to wait between connectivity tests. To guarantee seamless internet access for VRRP+ purposes, SureLink tests should occur more often than the default of 15 minutes.

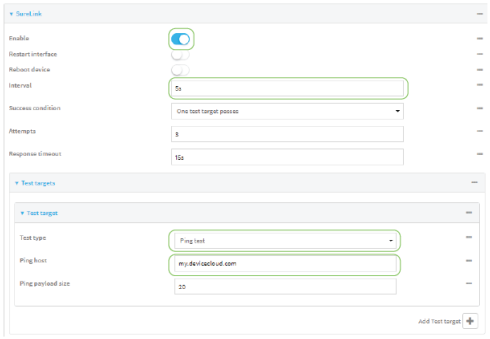
Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w**|**d**|**h**|**m**|**s**}. For example, to set **Interval** to five seconds, enter **5s**.

iv. Click to expand **Test targets** > **Test target**.

v. Configure the test target. For example, to configure SureLink to verify internet connectivity on the LAN by pinging my.devicecloud.com:

i. For **Test Type**, select **Ping test**.

ii. For **Ping host**, type **my.devicecloud.com**.



11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Create a new VRRP instance, or edit an existing one. See Configure VRRP for information about creating a new VRRP instance.
4. Enable VRRP+:

```
(config)> network vrrp VRRP_test vrrp_plus enable true
(config)>
```

5. Add interfaces to monitor. Generally, this will be a cellular or WAN interface.
   a. Use the **?** to determine available interfaces:
   b. Set the interface, for example:

```
(config)> add network vrrp VRRP_test vrrp_plus monitor_interface end
/network/interface/modem
(config)>
```

   c. (Optional) Repeat for additional interfaces.
6. Set the amount that the device's priority should be decreased or increased due to SureLink connectivity failure or success:

```
(config)> network vrrp VRRP_test vrrp_plus weight value
(config)>
```

   where *value* is an integer between **1** and **254**. The default is **10**.

   Along with the priority settings for devices in this VRRP pool, the amount entered here should be large enough to automatically demote a master device when SureLink connectivity fails. For example, if the VRRP master device has a priority of **100** and the backup device has a priority of **80**, then **weight** should be set to an amount greater than **20** so that if SureLink fails on the master, it will lower its priority to below **80**, and the backup device will assume the master role.
7. (Optional) For backup devices, enable the ability for the device to monitor the master device. This allows a backup VRRP device to monitor the master device, and increase its priority when the master device is failing SureLink tests. This can allow a device functioning as a backup device to promote itself to master.

```
(config)> network vrrp VRRP_test vrrp_plus monitor_master true
(config)>
```

8. Configure the VRRP interface:
   a. Configure the VRRP interface's DHCP server to use a custom gateway that corresponds to one of the VRRP virtual IP addresses:
      i. Set the DHCP server gateway type to custom:

```
(config)> network interface eth ipv4 dhcp_server advanced gateway
custom
(config)>
```

ii. Determine the VRRP virtual IP addresses:

```
(config)> show network vrrp VRRP_test virtual_address
0 192.168.3.3
1 10.10.10.1

(config)>
```

iii. Set the custom gateway to one of the VRRP virtual IP addresses. For example:

```
(config)> network interface eth ipv4 dhcp_server advanced gateway_
custom 192.168.3.3
(config)>
```

b. For backup devices, set the default gateway to the IP address of the VRRP interface on the master device. For example:

```
(config)> network interface eth ipv4 gateway 192.168.3.1
(config)>
```

c. For backup devices, enable and configure SureLink on the VRRP interface.

i. Determine the VRRP interface. Generally, this should be a LAN interface; VRRP+ will then monitor the LAN using SureLink to determine if the interface has network connectivity and promote a backup to master if SureLink fails.

```
(config)> show network vrrp VRRP_test interface
/network/interface/eth
(config)>
```

ii. Enable SureLink on the interface:

```
(config)> network interface eth ipv4 surelink enable true
(config)>
```

iii. Set the amount of time to wait between connectivity tests:

```
(config)> network interface eth ipv4 surelink interval value
(config)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **interval** to ten minutes, enter **5s**:

```
(config)> network interface eth ipv4 surelink interval 5s
(config)>
```

iv. Create a SureLink test target:

```
(config)> add network interface eth ipv4 surelink target end
(config network interface eth ipv4 surelink target 0)>
```

v. Configure the type of test for the test target:

```
(config network interface eth ipv4 surelink target 0)> test value
(config network interface eth ipv4 surelink target 0)>
```

where *value* is one of:

- **ping**: Tests connectivity by sending an ICMP echo request to a specified hostname or IP address.
  - Specify the hostname or IP address:

    ```
    (config network interface eth ipv4 surelink target 0)>
    ping_host host
    (config network interface eth ipv4 surelink target 0)>
    ```

  - (Optional) Set the size, in bytes, of the ping packet:

    ```
    (config network interface eth ipv4 surelink target 0)>
    ping_size [num]
    (config network interface eth ipv4 surelink target 0)>
    ```

- **dns**: Tests connectivity by sending a DNS query to the specified DNS server.
  - Specify the DNS server. Allowed value is the IP address of the DNS server.

    ```
    (config network interface eth ipv4 surelinktarget 0)> dns_
    server ip_address
    (config network interface eth ipv4 surelinktarget 0)>
    ```

- **dns_configured**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
- **http**: Tests connectivity by sending an HTTP or HTTPS GET request to the specified URL.
  - Specify the url:

    ```
    (config network interface eth ipv4 surelink target 0)>
    http_url value
    (config network interface eth ipv4 surelink target 0)>
    ```

    where *value* uses the format **http[s]://*hostname*/[*path*]**
- **interface_up**: The interface is considered to be down based on the interfaces down time, and the amount of time an initial connection to the interface takes before this test is considered to have failed.
  - (Optional) Set the amount of time that the interface can be down before this test is considered to have failed:

```
(config network interface eth ipv4 surelink target 0)>
interface_down_time value
(config network interface eth ipv4 surelink target 0)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **interface_down_time** to ten minutes, enter either **10m** or **600s**:

```
(config network interface eth ipv4 surelink target 0)>
interface_down_time 600s
(config network interface eth ipv4 surelink target 0)>
```

The default is 60 seconds.

- (Optional) Set the amount of time to wait for an initial connection to the interface before this test is considered to have failed:

```
(config network interface eth ipv4 surelink target 0)>
interface_timeout value
(config network interface eth ipv4 surelink target 0)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **interface_timeout** to ten minutes, enter either **10m** or **600s**:

```
(config network interface eth ipv4 surelink target 0)>
interface_timeout 600s
(config network interface eth ipv4 surelink target 0)>
```

The default is 60 seconds.

9. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

10. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Example: VRRP/VRRP+ configuration

This example configuration creates a VRRP pool containing two IX15 devices:

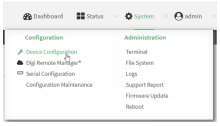


Configure device one (master device)

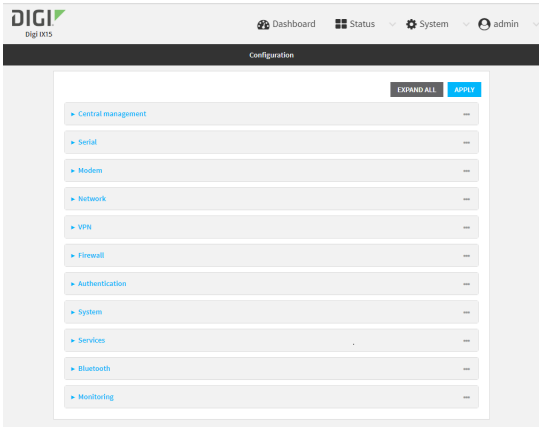
WebUI

Task 1: Configure VRRP on device one

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

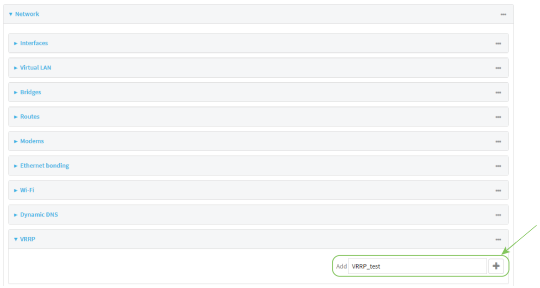


The **Configuration** window is displayed.

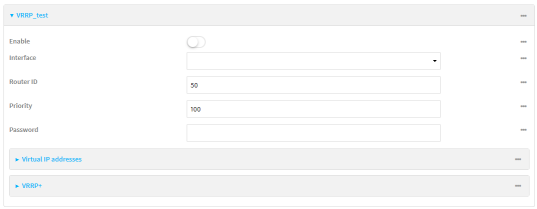


3. Click **Network** > **VRRP**.

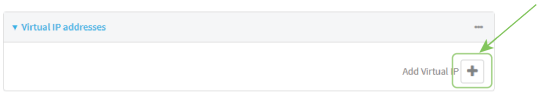
4. For **Add VRRP instance**, type a name for the VRRP instance and click ✚.



The new VRRP instance configuration is displayed.



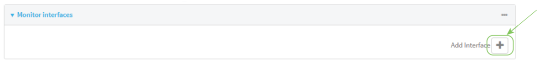
5. Click **Enable**.
6. For **Interface**, select **Interface: ETH**.
7. For **Router ID**, leave at the default setting of **50**.
8. For **Priority**, leave at the default setting of **100**.
9. Click to expand **Virtual IP addresses**.
10. Click ✚ to add a virtual IP address.



11. For **Virtual IP**, type **192.168.3.3**.

Task 2: Configure VRRP+ on device one

1. Click to expand **VRRP+**.
2. Click **Enable**.
3. Click to expand **Monitor interfaces**.
4. Click ✚ to add an interface for monitoring.



5. Select **Interface: Modem**.
6. For **Priority modifier**, type **30**.

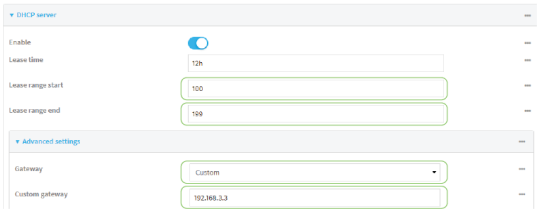
Task 3: Configure the IP address for the VRRP interface, ETH, on device one

1. Click **Network** > **Interfaces** > **ETH** > **IPv4**
2. For **Address**, type **192.168.3.1/24**.



Task 4: Configure the DHCP server for ETH on device one

1. Click to expand **Network** > **Interfaces** > **ETH** > **IPv4** > **DHCP Server**
2. For **Lease range start**, leave at the default of **100**.
3. For **Lease range end**, type **199**.
4. Click to expand **Advanced settings**.
5. For **Gateway**, select **Custom**.
6. For **Custom gateway**, enter **192.168.3.3**.



7. Click **Apply** to save the configuration and apply the change.



Command line

Task 1: Configure VRRP on device one

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Create the VRRP instance:

```
(config)> add network vrrp VRRP_test
(config network vrrp VRRP_test)>
```

4. Enable the VRRP instance:

```
(config network vrrp VRRP_test)> enable true
(config network vrrp VRRP_test)>
```

5. Set the VRRP interface to ETH:

```
(config network vrrp VRRP_test)> interface /network/interface/eth
(config network vrrp VRRP_test)>
```

6. Add the virtual IP address associated with this VRRP instance.

```
(config network vrrp VRRP_test)> add virtual_address end 192.168.3.3
(config network vrrp VRRP_test)>
```

Task 2: Configure VRRP+ on device one

1. Enable VRRP+:

```
(config network vrrp VRRP_test)> vrrp_plus enable true
(config network vrrp VRRP_test )>
```

2. Add the interface to monitor:

```
(config network vrrp VRRP_test)> add vrrp_plus monitor_interface end
/network/interface/modem
(config network vrrp VRRP_test)>
```

3. Set the amount that the device's priority should be decreased or increased due to SureLink connectivity failure or success to **30**:

```
(config network vrrp VRRP_test )> network vrrp VRRP_test vrrp_plus weight
30
(config network vrrp VRRP_test )>
```

Task 3: Configure the IP address for the VRRP interface, ETH, on device one

1. Type **...** to return to the root of the config prompt:

```
(config network vrrp VRRP_test )> ...
(config)>
```

2. Set the IP address for ETH:

```
(config)> network interface eth ipv4 address 192.168.3.1/24
(config)>
```

Task 4: Configure the DHCP server for ETH on device one

1. Set the start and end addresses of the DHCP pool to use to assign DHCP addresses to clients:
   a. Set the start address to **100**:

```
(config)> network interface eth ipv4 dhcp_server lease_start 100
(config)>
```

   b. Set the end address to **199**:

```
(config)> network interface eth ipv4 dhcp_server lease_end 199
(config)>
```

2. Set the DHCP server gateway type to custom:

```
(config)> network interface eth ipv4 dhcp_server advanced gateway custom
(config)>
```

3. Set the custom gateway to **192.168.3.3**:

```
(config)> network interface eth ipv4 dhcp_server advanced gateway_custom
192.168.3.3
(config)>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

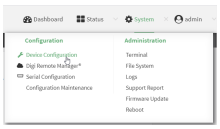
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure device two (backup device)

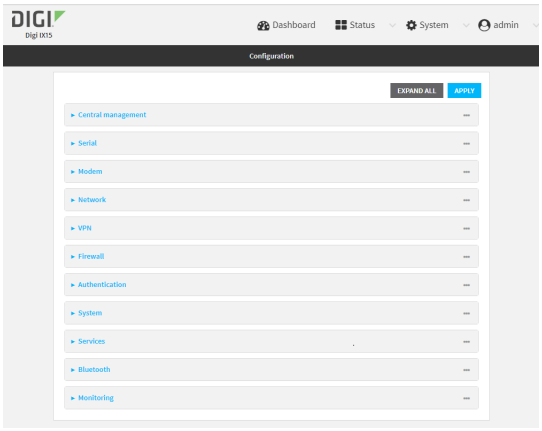
WebUI

Task 1: Configure VRRP on device two

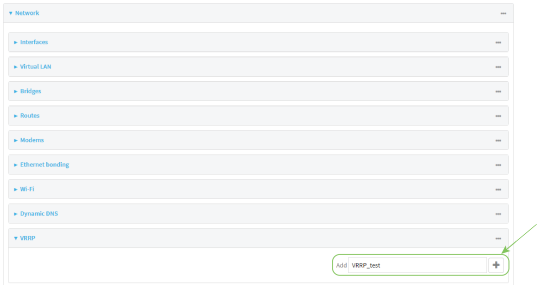
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



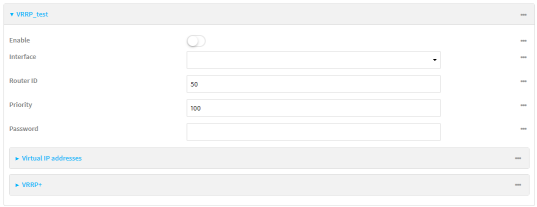
   The **Configuration** window is displayed.



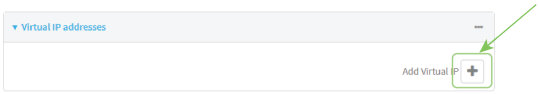
3. Click **Network** > **VRRP**.
4. For **Add VRRP instance**, type a name for the VRRP instance and click ✚.



The new VRRP instance configuration is displayed.



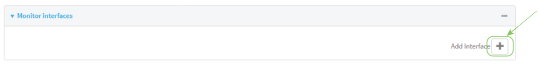
5. Click **Enable**.
6. For **Interface**, select **Interface: ETH**.
7. For **Router ID**, leave at the default setting of **50**.
8. For **Priority**, type **80**.
9. Click to expand **Virtual IP addresses**.
10. Click ✚ to add a virtual IP address.



11. For **Virtual IP**, type **192.168.3.3**.

Task 2: Configure VRRP+ on device two

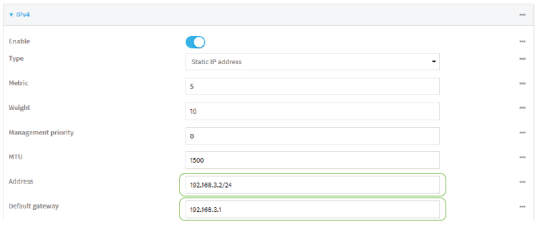
1. Click to expand **VRRP+**.
2. Click **Enable**.
3. Click to expand **Monitor interfaces**.
4. Click ✚ to add an interface for monitoring.



5. Select **Interface: Modem**.
6. Click to enable **Monitor VRRP+ master**.
7. For **Priority modifier**, type **30**.

Task 3: Configure the IP address for the VRRP interface, ETH, on device two

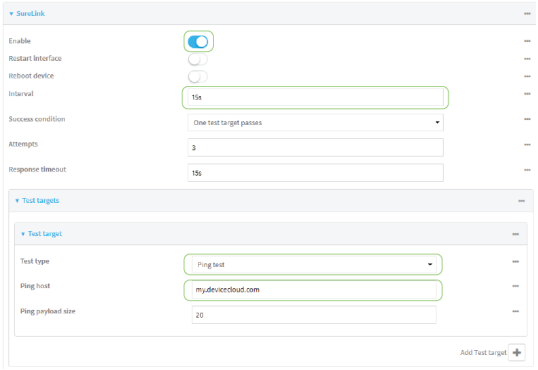
1. Click **Network** > **Interfaces** > **ETH** > **IPv4**
2. For **Address**, type **192.168.3.2/24**.
3. For **Default gateway**, type the IP address of the VRRP interface on the master device, configured above in Task 3, step 2 (**192.168.3.1**).



Task 4: Configure SureLink for ETH on device two

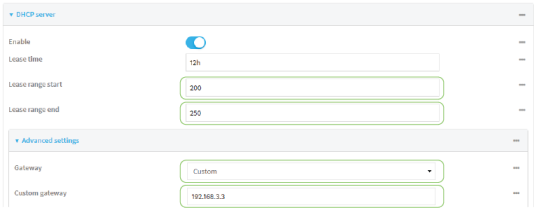
1. Click **Network** > **Interfaces** > **ETH** > **IPv4** > **SureLink**.
2. Click **Enable**.
3. For **Interval**, type **15s**.
4. Click to expand **Test targets** > **Test target**.
5. For **Test Type**, select **Ping test**.

6. For **Ping host**, type **my.devicecloud.com**.



Task 5: Configure the DHCP server for ETH on device two

1. Click to expand **Network** > **Interfaces** > **ETH** > **IPv4** > **DHCP Server**
2. For **Lease range start**, type **200**.
3. For **Lease range end**, type **250**.
4. Click **Advanced settings**.
5. For **Gateway**, select **Custom**.
6. For **Custom gateway**, enter **192.168.3.3**.



7. Click **Apply** to save the configuration and apply the change.



Command line

Task 1: Configure VRRP on device two

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Create the VRRP instance:

```
(config)> add network vrrp VRRP_test
(config network vrrp VRRP_test)>
```

4. Enable the VRRP instance:

```
(config network vrrp VRRP_test)> enable true
(config network vrrp VRRP_test)>
```

5. Set the VRRP interface to ETH:

```
(config network vrrp VRRP_test)> interface /network/interface/eth
(config network vrrp VRRP_test)>
```

6. Add the virtual IP address associated with this VRRP instance.

```
(config network vrrp VRRP_test)> add virtual_address end 192.168.3.3
(config network vrrp VRRP_test)>
```

Task 2: Configure VRRP+ on device two

1. Enable VRRP+:

```
(config network vrrp VRRP_test)> vrrp_plus enable true
(config network vrrp VRRP_test )>
```

2. Add the interface to monitor:

```
(config network vrrp VRRP_test)> add vrrp_plus monitor_interface end
/network/interface/modem
(config network vrrp VRRP_test)>
```

3. Enable the ability to monitor the master device:

```
(config network vrrp VRRP_test)> vrrp_plus monitor_master true
(config network vrrp VRRP_test)>
```

4. Set the amount that the device's priority should be decreased or increased due to SureLink connectivity failure or success to **30**:

```
(config network vrrp VRRP_test )> network vrrp VRRP_test vrrp_plus weight
30
(config network vrrp VRRP_test )>
```

Task 3: Configure the IP address for the VRRP interface, ETH, on device two

1. Type **...** to return to the root of the config prompt:

```
(config network vrrp VRRP_test )> ...
(config)>
```

2. Set the IP address for ETH:

```
(config)> network interface eth ipv4 address 192.168.3.2
(config)>
```

3. Set the default gateway to the IP address of the VRRP interface on the master device, configured above in Task 3, step 2 (**192.168.3.1**).

```
(config)> network interface eth ipv4 gateway 192.168.3.1
(config)>
```

Task 4: Configure SureLink for ETH on device two

1. Enable SureLink on the ETH interface:

```
(config)> network interface eth ipv4 surelink enable true
(config)>
```

2. Create a SureLink test target:

```
(config)> add network interface eth ipv4 surelink target end
(config network interface eth ipv4 surelink target 0)>
```

3. Set the type of test to ping:

```
(config network interface eth ipv4 surelink target 0)> test ping
(config network interface eth ipv4 surelink target 0)>
```

4. Set **my.devicecloud.com** as the hostname to ping:

```
(config network interface eth ipv4 surelink target 0)> ping_host
my.devicecloud.com
(config network interface eth ipv4 surelink target 0)>
```

Task 5: Configure the DHCP server for ETH on device two

1. Type **...** to return to the root of the configuration prompt:

```
(config network interface eth ipv4 surelink target 0)> ...
(config)>
```

2. Set the start and end addresses of the DHCP pool to use to assign DHCP addresses to clients:
   a. Set the start address to **200**:

```
(config)> network interface eth ipv4 dhcp_server lease_start 200
(config)>
```

   b. Set the end address to **250**:

```
(config)> network interface eth ipv4 dhcp_server lease_end 250
(config)>
```

3. Set the DHCP server gateway type to custom:

```
(config)> network interface eth ipv4 dhcp_server advanced gateway custom
(config)>
```

4. Set the custom gateway to **192.168.3.3**:

```
(config)> network interface eth ipv4 dhcp_server advanced gateway_custom
192.168.3.3
(config)>
```

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

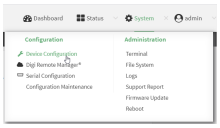
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show VRRP status and statistics

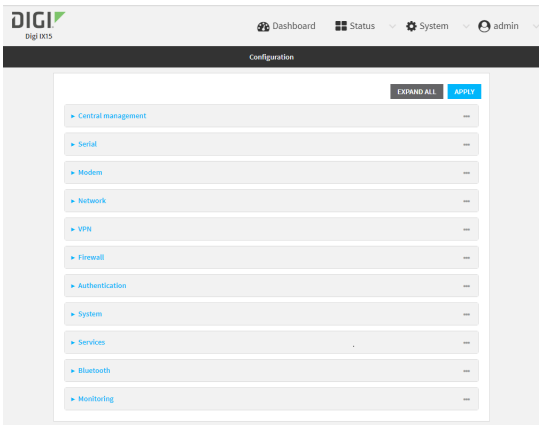
This section describes how to display VRRP status and statistics for a Digi IX15 Gateway device. VRRP status is available from the Web UI only.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

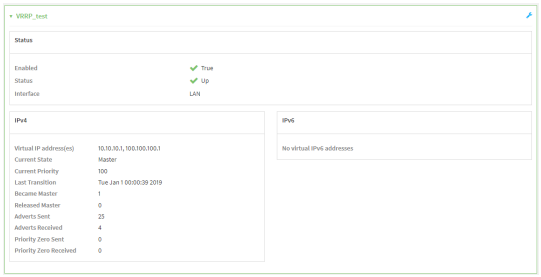


The **Configuration** window is displayed.



3. Click **Status** > **VRRP**.

   The **Virtual Router Redundancy Protocol** window is displayed.



Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the Admin CLI prompt, type show vrrp:

```
> show vrrp

 VRRP       Status  Proto  State   Virtual IP
 ----       ------  -----  ------  -------------
 VRRP_test  Up      IPv4   Backup  10.10.10.1
 VRRP_test  Up      IPv4   Backup  100.100.100.1
>
```

3. To display additional information about a specific VRRP instance, at the Admin CLI prompt, type show vrrp name *name*:

```
> show vrrp name VRRP_test

 VRRP_test VRRP Status
 ---------------------
 Enabled                  : True
 Status                   : Up
 Interface                : lan

 IPv4
   ----
   Virtual IP address(es) : 10.10.10.1, 100.100.100.1
   Current State          : Master
   Current Priority       : 100
   Last Transition        : Tue Jan  1 00:00:39 2019
   Became Master          : 1
   Released Master        : 0
   Adverts Sent           : 71
   Adverts Received       : 4
   Priority Zero Sent     : 0
   Priority zero Received : 0
```

```
>
```

Virtual Private Networks (VPN)

Virtual Private Networks (VPNs) are used to securely connect two private networks together so that devices can connect from one network to the other using secure channels.

This chapter contains the following topics:

IPsec 283
OpenVPN 336
Generic Routing Encapsulation (GRE) 368
L2TP 389
L2TPv3 Ethernet 408
NEMO 414

IPsec

IPsec is a suite of protocols for creating a secure communication link—an IPsec tunnel—between a host and a remote IP network or between two IP networks across a public network such as the Internet.

IPsec data protection

IPsec protects the data being sent across a public network by providing the following:

Data origin authentication
Authentication of data to validate the origin of data when it is received.

Data integrity
Authentication of data to ensure it has not been modified during transmission.

Data confidentiality
Encryption of data sent across the IPsec tunnel to ensure that an unauthorized device cannot read the data.

Anti-Replay
Authentication of data to ensure an unauthorized device has not injected it into the IPsec tunnel.

IPsec mode

The IX15 supports the **Tunnel** mode. With the **Tunnel** mode, the entire IP packet is encrypted and/or authenticated and then encapsulated as the payload in a new IP packet. **Transport** mode is not currently supported.

IPsec modes

IPsec can run in two different modes: **Tunnel** and **Transport**.

Tunnel
The entire IP packet is encrypted and/or authenticated and then encapsulated as the payload in a new IP packet.

Transport
Only the payload of the IP packet is encrypted and/or authenticated. The IP header is left untouched. This mode has limitations when using an authentication header, because the IP addresses in the IP header cannot be translated (for example, with Network Address Translation (NAT), as it would invalidate the authentication hash value.

Internet Key Exchange (IKE) settings

IKE is a key management protocol that allows IPsec to negotiate the security associations (SAs) that are used to create the secure IPsec tunnel. Both IKEv1 and IKEv2 are supported.

SA negotiations are performed in two phases, known as **phase 1** and **phase 2**.

Phase 1

In phase 1, IKE creates a secure authenticated communication channel between the device and the peer (the remote device which is at the other end of the IPsec tunnel) using the configured pre-shared key and the Diffie-Hellman key exchange. This creates the IKE SAs that are used to encrypt further IKE communications.

For IKEv1, there are two modes for the phase 1 negotiation: **Main mode** and **Aggressive mode**. IKEv2 does not use these modes.

Main mode
Main mode is the default mode. It is slower than aggressive mode, but more secure, in that all sensitive information sent between the device and its peer is encrypted.

Aggressive mode
Aggressive mode is faster than main mode, but is not as secure as main mode, because the device and its peer exchange their IDs and hash information in clear text instead of being encrypted. Aggressive mode is usually used when one or both of the devices have a dynamic external IP address.

Phase 2

In phase 2, IKE negotiates the SAs for IPsec. This creates two unidirectional SAs, one for each direction. Once the phase 2 negotiation is complete, the IPsec tunnel should be fully functional.

IPsec and IKE renegotiation

To reduce the chances of an IPsec tunnel being compromised, the IPsec SAs and IKE SA are renegotiated at a regular interval. This results in different encryption keys being used in the IPsec tunnel.

Authentication

Client authenticaton

XAUTH (extended authentication) pre-shared key authentication mode provides additional security by using client authentication credentials in addition to the standard pre-shared key. The IX15 device can be configured to authenticate with the remote peer as an XAUTH client.

RSA Signatures

With RSA signatures authentication, the IX15 device uses a private RSA key to authenticate with a remote peer that is using a corresponding public key.

Certificate-based Authentication

X.509 certificate-based authentication makes use of private keys on both the server and client which are secured and never shared. Both the server and client have a certificate which is generated with their respective private key and signed by a Certificate Authority (CA).

The IX15 implementation of IPsec can be configured to use X.509 certificate-based authentication using the private keys and certificates, along with a root CA certificate from the signing authority and, if available, a Certificate Revocation List (CRL).

Configure an IPsec tunnel

Configuring an IPsec tunnel with a remote device involves configuring the following items:

Required configuration items

- **IPsec tunnel configuration items:**
  - The mode: either tunnel or transport.
  - Enable the IPsec tunnel.

    The IPsec tunnel is enabled by default.

- The firewall zone of the IPsec tunnel.
- The routing metric for routes associated with this IPsec tunnel.
- The authentication type and pre-shared key or other applicable keys and certificates.

  If SCEP certificates will be selected as the Authentication type, create the SCEP client prior to configuring the IPsec tunnel. See Configure a Simple Certificate Enrollment Protocol client for instructions.
- The local endpoint type and ID values, and the remote endpoint host and ID values.

- **IKE configuration items**
  - The IKE version, either IKEv1 or IKEv2.
  - Whether to initiate a key exchange or wait for an incoming request.
  - The IKE mode, either main aggressive.
  - The IKE authentication protocol to use for the IPsec tunnel negotiation during phase 1 and phase 2.
  - The IKE encryption protocol to use for the IPsec tunnel negotiation during phase 1 and phase 2.
  - The IKE Diffie-Hellman group to use for the IPsec tunnel negotiation during phase 1 and phase 2.
- Enable dead peer detection and configure the delay and timeout.
- Destination networks that require source NAT.
- Active recovery configuration. See Configure SureLink active recovery for IPsec for information about IPsec active recovery.

Additional configuration items

The following additional configuration settings are not typically configured to get an IPsec tunnel working, but can be configured as needed:

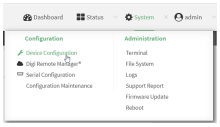
- Determine whether the device should use UDP encapsulation even when it does not detect that NAT is being used.
- If using IPsec failover, identify the primary tunnel during configuration of the backup tunnel.
- The Network Address Translation (NAT) keep alive time.
- The protocol, either Encapsulating Security Payload (ESP) or Authentication Header (AH).
- The management priority for the IPsec tunnel interface. The active interface with the highest management priority will have its address reported as the preferred contact address for central management and direct device access.
- Enable XAUTH client authentication, and the username and password to be used to authenticate with the remote peer.
- Enable Mode-configuration (MODECFG) to receive configuration information, such as the private IP address, from the remote peer.
- Disable the padding of IKE packets. This should normally not be done except for compatibility purposes.
- Destination networks that require source NAT.
- Depending on your network and firewall configuration, you may need to add a packet filtering rule to allow incoming IPsec traffic.

- **Tunnel and key renegotiating**
  - The lifetime of the IPsec tunnel before it is renegotiated.
  - The amount of time before the IKE phase 1 lifetime expires.
  - The amount of time before the IKE phase 2 lifetime expires
  - The lifetime margin, a randomizing amount of time before the IPsec tunnel is renegotiated.

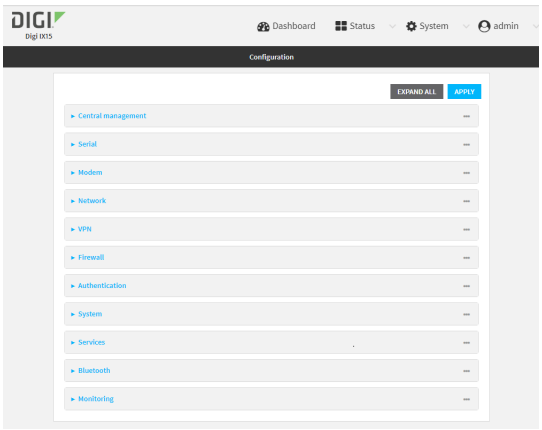
Note if the remote networks for an IPsec tunnel overlap with the networks for a WAN internet connection (wired, cellular, or otherwise), you must configure a static route to direct the traffic either through the IPsec tunnel, or through the WAN (outside of the IPsec tunnel). See Configure a static route for information about configuring a static route.

WebUI

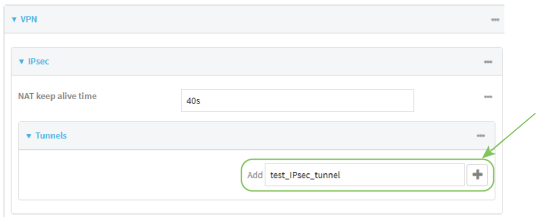
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



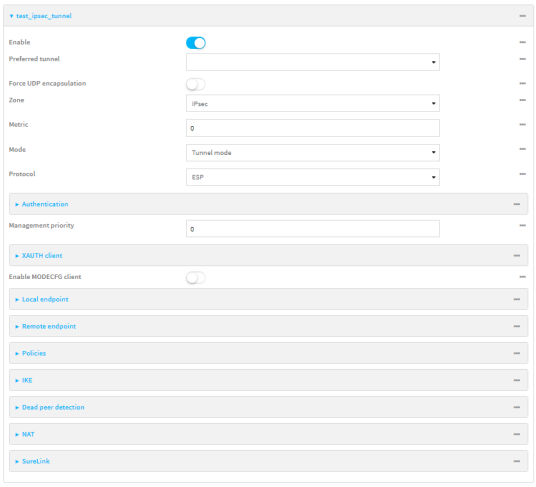
The **Configuration** window is displayed.



3. Click **VPN** > **IPsec**.
4. Click to expand **Tunnels**.
5. For **Add IPsec tunnel**, type a name for the tunnel and click **+**.



The new IPsec tunnel configuration is displayed.

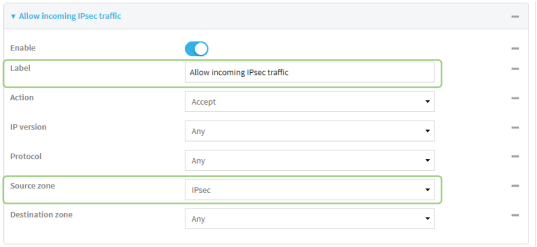


6. The IPsec tunnel is enabled by default. To disable, click **Enable**.
7. (Optional) **Preferred tunnel** provides an optional mechanism for IPsec failover behavior. See Configure IPsec failover for more information.
8. (Optional) Enable **Force UDP encapsulation** to force the tunnel to use UDP encapsulation even when it does not detect that NAT is being used.
9. For **Zone**, select the firewall zone for the IPsec tunnel. Generally this should be left at the default of **IPsec**.

   Note Depending on your network configuration, you may need to add a packet filtering rule to allow incoming traffic. For example, for the **IPsec** zone:

   a. Click to expand **Firewall** > **Packet filtering**.
   b. For **Add packet filter**, click ✚.
   c. For **Label**, type **Allow incoming IPsec traffic**.
   d. For **Source zone**, select **IPsec**.

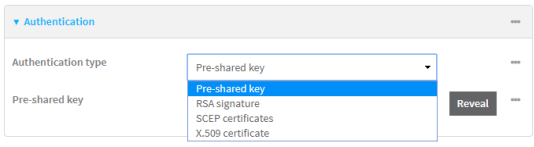
   Leave all other fields at their default settings.

10. For **Metric**, enter or select the priority of routes associated with this IPsec tunnel. When more than one active route matches a destination, the route with the lowest metric is used.

    The metric can also be used in tandem with SureLink to configure IPsec failover behavior. See Configure IPsec failover for more information.
11. For **Mode**, select **Tunnel mode**. **Transport mode** is not currently supported.

12. Select the Mode, either:
    - **Tunnel mode**: The entire IP packet is encrypted and/or authenticated and then encapsulated as the payload in a new IP packet.
    - **Transport mode**: Only the payload of the IP packet is encrypted and/or authenticated. The IP header is unencrypted.
13. Select the **Protocol**, either:
    - **ESP** (Encapsulating Security Payload): Provides encryption as well as authentication and integrity.
    - **AH** (Authentication Header): Provides authentication and integrity only.
14. Click to expand **Authentication**.



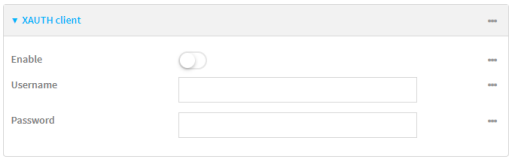
    a. For **Authentication type**, select one of the following:
        - **Pre-shared key**: Uses a pre-shared key (PSK) to authenticate with the remote peer.
            i. Type the **Pre-shared key**.
        - **Asymmetric pre-shared keys**: Uses asymmetric pre-shared keys to authenticate with the remote peer.
            i. For **Local key**, type the local pre-shared key. This must be the same as the remote key on the remote host.
            ii. For **Remote key**, type the remote pre-shared key. This must be the same as the local key on the remote host.
        - **RSA signature**: Uses a private RSA key to authenticate with the remote peer.
            i. For **Private key**, paste the device's private RSA key in PEM format.
            ii. Type the **Private key passphrase** that is used to decrypt the private key. Leave blank if the private key is not encrypted.
            iii. For **Peer public key**, paste the peer's public RSA key in PEM format.
        - **SCEP certificates**: Uses Simple Certificate Enrollment Protocol (SCEP) to download a private key, certificates, and an optional Certificate Revocation List (CRL) to the IX15 device from a SCEP server.

          You must create the SCEP client prior to configuring the IPsec tunnel. See Configure a Simple Certificate Enrollment Protocol client for instructions.
            i. For **SCEP Client**, select the SCEP client.
        - **X.509 certificate**: Uses private key and X.509 certificates to authenticate with the remote peer.
            i. For **Private key**, paste the device's private RSA key in PEM format.
            ii. Type the **Private key passphrase** that is used to decrypt the private key. Leave blank if the private key is not encrypted.
            iii. For **Certificate**, paste the local X.509 certificate in PEM format.

iv. For Peer verification, select either:

- **Peer certificate**: For **Peer certificate**, paste the peer's X.509 certificate in PEM format.
- **Certificate Authority**: For **Certificate Authority chain**, paste the Certificate Authority (CA) certificates. These must include all peer certificates in the chain up to the root CA certificate, in PEM format.

15. (Optional) For **Management Priority**, set the management priority for this IPsec tunnel. A tunnel that is up and has the highest priority will be used for central management and direct device access.
16. (Optional) To configure the device to connect to its remote peer as an XAUTH client:
    a. Click to expand **XAUTH client**.



    b. Click **Enable**.
    c. Type the **Username** and **Password** that the device will use to authenticate as an XAUTH client with the peer.
17. (Optional) Click **Enable MODECFG client** to receive configuration information, such as the private IP address, from the remote peer.
18. Click to expand **Local endpoint**.
    a. For **Type**, select either:
        - **Default route**: Uses the same network interface as the default route.
        - **Interface**: Select the **Interface** to be used as the local endpoint.
    b. Click to expand **ID**.
        i. Select the ID type:
            - **Auto**: The ID will be automatically determined from the value of the tunnels endpoints.
            - **Raw**: Enter an ID and have it passed unmodified to the underlying IPsec stack.

              For **Raw ID value**, type the ID that will be passed.
            - **Any**: Any ID will be accepted.
            - **IPv4**: The ID will be interpreted as an IP address and sent as an ID_IPV4_ADDR IKE identity.

              For **IPv4 ID value**, type an IPv4 formatted ID. This can be a fully-qualified domain name or an IPv4 address.
            - **IPv6**: The ID will be interpreted as an IP address and sent as an ID_IPV6_ADDR IKE identity.

              For **IPv6 ID value**, type an IPv6 formatted ID. This can be a fully-qualified domain name or an IPv6 address.
            - **RFC822/Email**: The ID will be interpreted as an RFC822 (email address).

              For **RFC822 ID value**, type the ID in internet email address format.

- **FQDN**: The ID will be interpreted as FQDN (Fully Qualified Domain Name) and sent as an ID_FQDN IKE identity.

  For **FQDN ID value**, type the ID as an FQDN.
- **KeyID**: The ID will be interpreted as a Key ID and sent as an ID_KEY_ID IKE identity.

  For **KEYID ID value**, type the key ID.
- **MAC address**: The device's primary MAC address will be used as the ID and sent as a ID_KEY_ID IKE identity.
- **Serial number**: The device's serial number will be used as the ID and sent as a ID_KEY_ID IKE identity.

19. Click to expand **Remote endpoint**.
    a. For **IP version**, select either **IPv4** or **IPv6**.
    b. For **Hostname list selection**, select one of the following:
       - **Round robin**: Attempts to connect to hostnames sequentially based on the list order.
       - **Random**: Randomly selects an IPsec peer to connect to from the hostname list.
       - **Priority ordered**: Selects the first hostname in the list that is resolvable.
    c. Click to expand **Hostname**.
       i. Click ✚ next to **Add Hostname**.
       ii. For Hostname, type a hostname or IPv4 address. If your device is not configured to initiate the IPsec connection (see **IKE** > **Initiate connection**), you can also use the keyword **any**, which means that the hostname is dynamic or unknown.
       iii. Click ✚ again to add additional hostnames.
    d. Click to expand **ID**.
       i. Select the ID type:
          - **Auto**: The ID will be automatically determined from the value of the tunnels endpoints.
          - **Raw**: Enter an ID and have it passed unmodified to the underlying IPsec stack.

            For **Raw ID value**, type the ID that will be passed.
          - **Any**: Any ID will be accepted.
          - **IPv4**: The ID will be interpreted as an IPv4 address and sent as an ID_IPV4_ADDR IKE identity.

            For **IPv4 ID value**, type an IPv4 formatted ID. This can be a fully-qualified domain name or an IPv4 address.
          - **IPv6**: The ID will be interpreted as an IPv6 address and sent as an ID_IPV6_ADDR IKE identity.

            For **IPv6 ID value**, type an IPv6 formatted ID. This can be a fully-qualified domain name or an IPv6 address.
          - **RFC822/Email**: The ID will be interpreted as an RFC822 (email address).

            For **RFC822 ID value**, type the ID in internet email address format.
          - **FQDN**: The ID will be interpreted as FQDN (Fully Qualified Domain Name) and sent as an ID_FQDN IKE identity.

For **FQDN ID value**, type the ID as an FQDN.

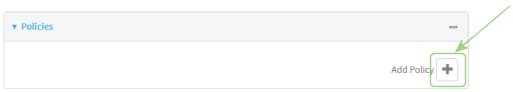
- **KeyID**: The ID will be interpreted as a Key ID and sent as an ID_KEY_ID IKE identity.

  For **KEYID ID value**, type the key ID.
- **MAC address**: The device's primary MAC address will be used as the ID and sent as a ID_KEY_ID IKE identity.
- **Serial number**: The device's serial number will be used as the ID and sent as a ID_KEY_ID IKE identity.

20. Click to expand **Policies**.

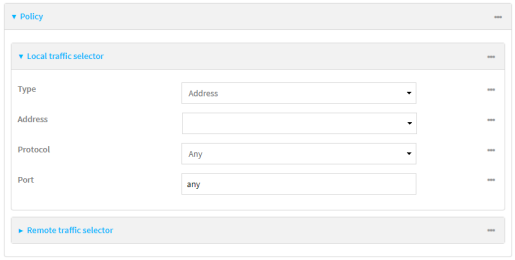
    Policies define the network traffic that will be encapsulated by this tunnel.

    a. Click ✚ to create a new policy.



    The new policy configuration is displayed.

    b. Click to expand **Local traffic selector**.



    c. For **Type**, select one of the following:

    - **Address**: The address of a local network interface.

      For **Address**, select the appropriate interface.
    - **Network**: The subnet of a local network interface.

      For **Address**, select the appropriate interface.
    - **Custom network**: A user-defined network.

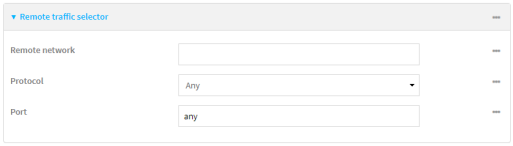
      For **Custom network**, enter the IPv4 address and optional netmask.
    - **Request a network**: Requests a network from the remote peer.
    - **Dynamic**: Uses the address of the local endpoint.

    d. For **Protocol**, select one of the following:

    - **Any**: Matches any protocol.
    - **TCP**: Matches TCP protocol only.
    - **UDP**: Matches UDP protocol only.
    - **ICMP**: Matches ICMP requests only.
    - **Other protocol**: Matches an unlisted protocol.

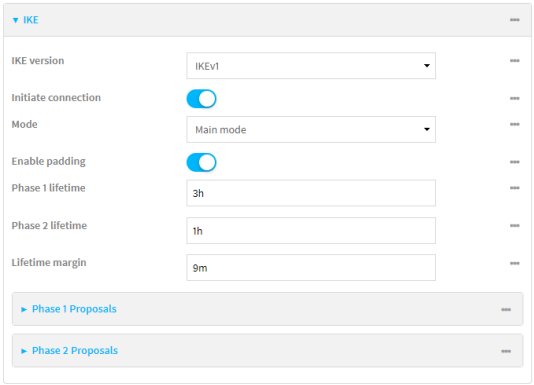
      If **Other protocol** is selected, type the number of the protocol.

e. For **Port**, type the port matching criteria.
   Allowed values are a port number, a range of port numbers, or **any**.
f. (Optional) Click to expand **Remote traffic selector**.



g. For **Remote network**, enter the IP address and optional netmask of the remote network.
h. For **Protocol**, select one of the following:
   - **Any**: Matches any protocol.
   - **TCP**: Matches TCP protocol only.
   - **UDP**: Matches UDP protocol only.
   - **ICMP**: Matches ICMP requests only.
   - **Other protocol**: Matches an unlisted protocol.
     If **Other protocol** is selected, type the number of the protocol.
i. For **Port**, type the port matching criteria.
   Allowed values are a port number, a range of port numbers, or **any**.

21. Click to expand **IKE**.



a. For **IKE version**, select either IKEv1 or IKEv2. This setting must match the peer's IKE version.
b. **Initiate connection** instructs the device to initiate the key exchange, rather than waiting for an incoming request. This must be disabled if **Remote endpoint** > **Hostname** is set to **any**.
c. For **Mode**, select either **Main mode** or **Aggressive mode**.
d. For **IKE fragmentation**, select one of the following:
   - **If supported by the peer**: Send oversized IKE messages in fragments, if the peer supports receiving them.
   - **Always**: Always send IKEv1 messages in fragments. For IKEv2, this option is equivalent to **If supported by the peer**.
   - **Never**: Do not send oversized IKE messages in fragments.

- **Accept**: Do not send oversized IKE messages in fragments, but announce support for fragmentation to the peer.

The default is **Always**.

e. For **Enable padding**, click to disable the padding of IKE packets. This should normally not be disabled except for compatibility purposes.

f. For Phase 1 lifetime, enter the amount of time that the IKE security association expires after a successful negotiation and must be re-authenticated.

Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

For example, to set **Phase 1 lifetime** to ten minutes, enter **10m** or **600s**.

g. For Phase 2 lifetime, enter the amount of time that the IKE security association expires after a successful negotiation and must be rekeyed.

Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

For example, to set **Phase 2 lifetime** to ten minutes, enter **10m** or **600s**.

h. For Lifetime margin, enter a randomizing amount of time before the IPsec tunnel is renegotiated.

Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

For example, to set **Lifetime margin** to ten minutes, enter **10m** or **600s**.

i. Click to expand **Phase 1 Proposals**.

i. Click ✚ to create a new phase 1 proposal.

ii. For **Cipher**, select the type of encryption.

iii. For **Hash**, select the type of hash to use to verify communication integrity.

iv. For **Diffie-Hellman group**, select the type of Diffie-Hellman group to use for key exchange.

v. You can add additional Phase 1 proposals by clicking ✚ next to **Add Phase 1 Proposal**.

j. Click to expand **Phase 2 Proposals**.

i. Click ✚ to create a new phase 2 proposal.

ii. For **Cipher**, select the type of encryption.

iii. For **Hash**, select the type of hash to use to verify communication integrity.

iv. For **Diffie-Hellman group**, select the type of Diffie-Hellman group to use for key exchange.

v. You can add additional Phase 2 proposals by clicking ✚ next to **Add Phase 2 Proposal**.

22. (Optional) Click to expand **Dead peer detection**. Dead peer detection is enabled by default. Dead peer detection uses periodic IKE transmissions to the remote endpoint to detect whether tunnel communications have failed, allowing the tunnel to be automatically restarted when failure occurs.

a. To enable or disable dead peer detection, click **Enable**.

b. For **Delay**, type the number of seconds between transmissions of dead peer packets. Dead peer packets are only sent when the tunnel is idle.

c. For **Timeout**, type the number of seconds to wait for a response from a dead peer packet before assuming the tunnel has failed.

23. (Optional) Click to expand **NAT** to create a list of destination networks that require source NAT.

a. Click **+** next to **Add NAT destination**.

b. For **Destination network**, type the IPv4 address and optional netmask of a destination network that requires source NAT. You can also use **any**, meaning that any destination network connected to the tunnel will use source NAT.

24. See Configure SureLink active recovery for IPsec for information about IPsec **Active recovery**.

25. (Optional) Click **Advanced** to set various IPsec-related time out, keep alive, and related values.

26. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add an IPsec tunnel. For example, to add an IPsec tunnel named **ipsec_example**:

```
(config)> add vpn ipsec tunnel ipsec_example
(config vpn ipsec tunnel ipsec_example)>
```

   The IPsec tunnel is enabled by default. To disable:

```
(config vpn ipsec tunnel ipsec_example)> enable false
(config vpn ipsec tunnel ipsec_example)>
```

4. (Optional) Set the tunnel to use UDP encapsulation even when it does not detect that NAT is being used:

```
(config vpn ipsec tunnel ipsec_example)> force_udp_encap true
(config vpn ipsec tunnel ipsec_example)>
```

5. Set the firewall zone for the IPsec tunnel. Generally this should be left at the default of **ipsec**.

```
(config vpn ipsec tunnel ipsec_example)> zone zone
(config vpn ipsec tunnel ipsec_example)>
```

   To view a list of available zones:

```
(config vpn ipsec tunnel ipsec_example)> zone ?

Zone: The firewall zone assigned to this IPsec tunnel. This can be used
by packet filtering rules
and access control lists to restrict network traffic on this tunnel.
Format:
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup
Default value: ipsec
Current value: ipsec

(config vpn ipsec tunnel ipsec_example)>
```

Note Depending on your network configuration, you may need to add a packet filtering rule to allow incoming traffic. For example, for the **IPsec** zone:

a. Type **...** to move to the root of the configuration:

```
(config vpn ipsec tunnel ipsec_example)> ...
(config)>
```

b. Add a packet filter:

```
(config)> add firewall filter end
(config firewall filter 2)>
```

c. Set the label to **Allow incoming IPsec traffic**:

```
(config config firewall filter 2)> label "Allow incoming IPsec
traffic"
(config firewall filter 2)>
```

d. Set the source zone to **ipsec**:

```
(config config firewall filter 2)> src_zone ipsec
(config firewall filter 2)>
```

6. Set the metric for the IPsec tunnel. When more than one active route matches a destination, the route with the lowest metric is used. The metric can also be used in tandem with SureLink to configure IPsec failover behavior. See Configure IPsec failover for more information.

```
(config vpn ipsec tunnel ipsec_example)> metric value
(config vpn ipsec tunnel ipsec_example)>
```

where *value* is any integer between **0** and **65535**.

7. Set the mode:

```
(config vpn ipsec tunnel ipsec_example)> mode mode
(config vpn ipsec tunnel ipsec_example)>
```

where *mode* is either:

- **tunnel**: The entire IP packet is encrypted and/or authenticated and then encapsulated as the payload in a new IP packet.
- **transport**: Only the payload of the IP packet is encrypted and/or authenticated. The IP header is unencrypted.

The default is **tunnel**.

8. Set the protocol:

```
(config vpn ipsec tunnel ipsec_example)> type protocol
(config vpn ipsec tunnel ipsec_example)>
```

where *protocol* is either:

- **esp** (Encapsulating Security Payload): Provides encryption as well as authentication and integrity.
- **ah** (Authentication Header): Provides authentication and integrity only.

The default is **esp**.

9. (Optional) Set the management priority for this IPsec tunnel:

```
(config vpn ipsec tunnel ipsec_example)> mgmt value
(config vpn ipsec tunnel ipsec_example)>
```

where *value* is any interger between **0** and **1000**.

10. Set the authentication type:

```
(config vpn ipsec tunnel ipsec_example)> auth type value
(config vpn ipsec tunnel ipsec_example)>
```

where *value* is one of:

- **secret**: Uses a pre-shared key (PSK) to authenticate with the remote peer.
  a. Set the pre-shared key:

```
(config vpn ipsec tunnel ipsec_example)> auth secret key
(config vpn ipsec tunnel ipsec_example)>
```

- **asymmetric-secrets**: Uses asymmetric pre-shared keys to authenticate with the remote peer.
  a. Set the local pre-shared key. This must be the same as the remote key on the remote host.:

```
(config vpn ipsec tunnel ipsec_example)> auth local_secret key
(config vpn ipsec tunnel ipsec_example)>
```

  b. Set the remote pre-shared key. This must be the same as the local key on the remote host.:

```
(config vpn ipsec tunnel ipsec_example)> auth remote_secret key
(config vpn ipsec tunnel ipsec_example)>
```

- **rsasig**: Uses a private RSA key to authenticate with the remote peer.
  a. For the **private_key** parameter, paste the device's private RSA key in PEM format:

```
(config vpn ipsec tunnel ipsec_example)> auth private_key key
(config vpn ipsec tunnel ipsec_example)>
```

  b. Set the private key passphrase that is used to decrypt the private key. Leave blank if the private key is not encrypted.

```
(config vpn ipsec tunnel ipsec_example)> auth private_key_
passphrase passphrase
(config vpn ipsec tunnel ipsec_example)>
```

  c. For the **peer_public_key** parameter, paste the peer's public RSA key in PEM format:

```
(config vpn ipsec tunnel ipsec_example)> auth peer_public_key
key
(config vpn ipsec tunnel ipsec_example)>
```

- **x509**: Uses private key and X.509 certificates to authenticate with the remote peer.
  a. For the **private_key** parameter, paste the device's private RSA key in PEM format:

```
(config vpn ipsec tunnel ipsec_example)> auth private_key key
(config vpn ipsec tunnel ipsec_example)>
```

  b. Set the private key passphrase that is used to decrypt the private key. Leave blank if the private key is not encrypted.

```
(config vpn ipsec tunnel ipsec_example)> auth private_key_
passphrase passphrase
(config vpn ipsec tunnel ipsec_example)>
```

  c. For the **cert** parameter, paste the local X.509 certificate in PEM format:

```
(config vpn ipsec tunnel ipsec_example)> auth cert certificate
(config vpn ipsec tunnel ipsec_example)>
```

  d. Set the method for verifying the peer's X.509 certificate:

```
(config vpn ipsec tunnel ipsec_example)> auth peer_verify value
(config vpn ipsec tunnel ipsec_example)>
```

  where *value* is either:

  - **cert**: Uses the peer's X.509 certificate in PEM format for verification.
    - For the **peer_cert** parameter, paste the peer's X.509 certificate in PEM format:

```
(config vpn ipsec tunnel ipsec_example)> auth peer_cert
certificate
(config vpn ipsec tunnel ipsec_example)>
```

  - **ca**: Uses the Certificate Authority chain for verification.
    - For the **ca_cert** parameter, paste the Certificate Authority (CA) certificates. These must include all peer certificates in the chain up to the root CA certificate, in PEM format.

```
(config vpn ipsec tunnel ipsec_example)> auth ca_cert cert_
chain
(config vpn ipsec tunnel ipsec_example)>
```

11. (Optional) Configure the device to connect to its remote peer as an XAUTH client:
    a. Enable XAUTH client functionality:

```
(config vpn ipsec tunnel ipsec_example)> xauth_client enable true
(config vpn ipsec tunnel ipsec_example)>
```

b. Set the XAUTH client username:

```
(config vpn ipsec tunnel ipsec_example)> xauth_client username name
(config vpn ipsec tunnel ipsec_example)>
```

c. Set the XAUTH client password:

```
(config vpn ipsec tunnel ipsec_example)> xauth_client password pwd
(config vpn ipsec tunnel ipsec_example)>
```

12. (Optional) Enable MODECFG client functionality:

    MODECFG client functionality configures the device to receive configuration information, such as the private IP address, from the remote peer.

    a. Enable MODECFG client functionality:

```
(config vpn ipsec tunnel ipsec_example)> modecfg_client enable true
(config vpn ipsec tunnel ipsec_example)>
```

13. Configure the local endpoint:

    a. Set the method for determining the local network interface:

```
(config vpn ipsec tunnel ipsec_example)> local type value
(config vpn ipsec tunnel ipsec_example)>
```

    where *value* is either:

    - **defaultroute**: Uses the same network interface as the default route.
    - **interface**: Select the **Interface** to be used as the local endpoint.

    b. Set the ID type:

```
(config vpn ipsec tunnel ipsec_example)> local id type value
(config vpn ipsec tunnel ipsec_example)>
```

    where *value* is one of:

    - **auto**: The ID will be automatically determined from the value of the tunnels endpoints.
    - **raw**: Enter an ID and have it passed unmodified to the underlying IPsec stack.

      Set the unmodified ID that will be passed:

```
(config vpn ipsec tunnel ipsec_example)> local id type raw_id id
(config vpn ipsec tunnel ipsec_example)>
```

    - **any**: Any ID will be accepted.
    - **ipv4**: The ID will be interpreted as an IPv4 address and sent as an ID_IPV4_ADDR IKE identity.

      Set an IPv4 formatted ID. This can be a fully-qualified domain name or an IPv4 address.

```
(config vpn ipsec tunnel ipsec_example)> local id type ipv4_id
id
(config vpn ipsec tunnel ipsec_example)>
```

- **ipv6**: The ID will be interpreted as an IPv6 address and sent as an ID_IPV6_ADDR IKE identity.

  Set an IPv6 formatted ID. This can be a fully-qualified domain name or an IPv6 address.

```
(config vpn ipsec tunnel ipsec_example)> local id type ipv6_id
id
(config vpn ipsec tunnel ipsec_example)>
```

- **rfc822**: The ID will be interpreted as an RFC822 (email address).

  Set the ID in internet email address format:

```
(config vpn ipsec tunnel ipsec_example)> local id type rfc822_id
id
(config vpn ipsec tunnel ipsec_example)>
```

- **fqdn**: The ID will be interpreted as FQDN (Fully Qualified Domain Name) and sent as an ID_FQDN IKE identity.
- **keyid**: The ID will be interpreted as a Key ID and sent as an ID_KEY_ID IKE identity.

  Set the key ID:

```
(config vpn ipsec tunnel ipsec_example)> local id type keyid_id
id
(config vpn ipsec tunnel ipsec_example)>
```

- **mac_address**: The device's MAC address will be used for the Key ID and sent as an ID_KEY_ID IKE identity.
- **serial_number**: The ID device's serial number will be used for the Key ID and sent as an ID_KEY_ID IKE identity.

14. Configure the remote endpoint:
    a. Add a remote hostname:

```
(config vpn ipsec tunnel ipsec_example)> add remote hostname end value
(config vpn ipsec tunnel ipsec_example)>
```

    where *value* is the hostname or IPv4 address of the IPsec peer. If your device is not configured to initiate the IPsec connection (see ike initiate), you can also use the keyword **any**, which means that the hostname is dynamic or unknown.

    Repeat for additional hostnames.

    b. Set the hostname selection type:

```
(config vpn ipsec tunnel ipsec_example)> remote hostname_selection
value
(config vpn ipsec tunnel ipsec_example)>
```

    where *value* is one of:

- **round_robin**: Attempts to connect to hostnames sequentially based on the list order.
- **random**: Randomly selects an IPsec peer to connect to from the hostname list.
- **priority**: Selects the first hostname in the list that is resolvable.

c. Set the ID type:

```
(config vpn ipsec tunnel ipsec_example)> remote id type value
(config vpn ipsec tunnel ipsec_example)>
```

where *value* is one of:

- **auto**: The ID will be automatically determined from the value of the tunnels endpoints.
- **raw**: Enter an ID and have it passed unmodified to the underlying IPsec stack.

  Set the unmodified ID that will be passed:

  ```
  (config vpn ipsec tunnel ipsec_example)> remote id type raw_id
  id
  (config vpn ipsec tunnel ipsec_example)>
  ```

- **any**: Any ID will be accepted.
- **ipv4**: The ID will be interpreted as an IPv4 address and sent as an ID_IPV4_ADDR IKE identity.

  Set an IPv4 formatted ID. This can be a fully-qualified domain name or an IPv4 address.

  ```
  (config vpn ipsec tunnel ipsec_example)> remote id type ipv4_id
  id
  (config vpn ipsec tunnel ipsec_example)>
  ```

- **ipv6**: The ID will be interpreted as an IPv6 address and sent as an ID_IPV6_ADDR IKE identity.

  Set an IPv6 formatted ID. This can be a fully-qualified domain name or an IPv6 address.

  ```
  (config vpn ipsec tunnel ipsec_example)> remote id type ipv6_id
  id
  (config vpn ipsec tunnel ipsec_example)>
  ```

- **rfc822**: The ID will be interpreted as an RFC822 (email address).

  Set the ID in internet email address format:

  ```
  (config vpn ipsec tunnel ipsec_example)> remote id type rfc822_
  id id
  (config vpn ipsec tunnel ipsec_example)>
  ```

- **fqdn**: The ID will be interpreted as FQDN (Fully Qualified Domain Name) and sent as an ID_FQDN IKE identity.

- **keyid**: The ID will be interpreted as a Key ID and sent as an ID_KEY_ID IKE identity.

  Set the key ID:

  ```
  (config vpn ipsec tunnel ipsec_example)> remote id type keyid_id id
  (config vpn ipsec tunnel ipsec_example)>
  ```

- **mac_address**: The device's MAC address will be used for the Key ID and sent as an ID_KEY_ID IKE identity.
- **serial_number**: The ID device's serial number will be used for the Key ID and sent as an ID_KEY_ID IKE identity.

15. Configure IKE settings:
    a. Set the IKE version:

    ```
    (config vpn ipsec tunnel ipsec_example)> ike version value
    (config vpn ipsec tunnel ipsec_example)>
    ```

    where *value* is either **ikev1** or **ikev2**. This setting must match the peer's IKE version.

    b. Determine whether the device should initiate the key exchange, rather than waiting for an incoming request. By default, the device will initiate the key exchange. This must be disabled if remote hostname is set to **any**. To disable:

    ```
    (config vpn ipsec tunnel ipsec_example)> ike initiate false
    (config vpn ipsec tunnel ipsec_example)>
    ```

    c. Set the IKE phase 1 mode:

    ```
    (config vpn ipsec tunnel ipsec_example)> ike mode value
    (config vpn ipsec tunnel ipsec_example)>
    ```

    where *value* is either **aggressive** or **main**.

    d. Set the IKE fragmentation:

    ```
    (config vpn ipsec tunnel ipsec_example)> ike fragmentation value
    (config vpn ipsec tunnel ipsec_example)>
    ```

    where *value* is one of:

    - **if_supported**: Send oversized IKE messages in fragments, if the peer supports receiving them.
    - **always**: Always send IKEv1 messages in fragments. For IKEv2, this option is equivalent to **if supported**.
    - **never**: Do not send oversized IKE messages in fragments.
    - **accept**: Do not send oversized IKE messages in fragments, but announce support for fragmentation to the peer.

    The default is **always**.

    e. Padding of IKE packets is enabled by default and should normally not be disabled except for compatibility purposes. To disable:

```
(config vpn ipsec tunnel ipsec_example)> ike pad false
(config vpn ipsec tunnel ipsec_example)>
```

f. Set the amount of time that the IKE security association expires after a successful negotiation and must be re-authenticated:

```
(config vpn ipsec tunnel ipsec_example)> ike phase1_lifetime value
(config vpn ipsec tunnel ipsec_example)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **phase1_lifetime** to ten minutes, enter either **10m** or **600s**:

```
(config vpn ipsec tunnel ipsec_example)> ike phase1_lifetime 600s
(config vpn ipsec tunnel ipsec_example)>
```

The default is three hours.

g. Set the amount of time that the IKE security association expires after a successful negotiation and must be rekeyed.

```
(config vpn ipsec tunnel ipsec_example)> ike phase2_lifetime value
(config vpn ipsec tunnel ipsec_example)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **phase2_lifetime** to ten minutes, enter either **10m** or **600s**:

```
(config vpn ipsec tunnel ipsec_example)> ike phase2_lifetime 600s
(config vpn ipsec tunnel ipsec_example)>
```

The default is one hour.

h. Set a randomizing amount of time before the IPsec tunnel is renegotiated:

```
(config vpn ipsec tunnel ipsec_example)> ike lifetime_margin value
(config vpn ipsec tunnel ipsec_example)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **lifetime_margin** to ten minutes, enter either **10m** or **600s**:

```
(config vpn ipsec tunnel ipsec_example)> ike lifetime_margin 600s
(config vpn ipsec tunnel ipsec_example)>
```

The default is nine minutes.

i. Configure the types of encryption, hash, and Diffie-Hellman group to use during phase 1:

i. Add a phase 1 proposal:

```
(config vpn ipsec tunnel ipsec_example)> add ike phase1_proposal
end
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)>
```

ii. Set the type of encryption to use during phase 1:

```
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)>
cipher value
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)>
```

where *value* is one of **3des**, **aes128**, **aes192**, **aes256**, or **null**. The default is **3des**.

iii. Set the type of hash to use during phase 1 to verify communication integrity:

```
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)>
hash value
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)>
```

where *value* is one of **md5**, **sha1**, **sha256**, **sha384**, or **sha512**. The default is **sha1**.

iv. Set the type of Diffie-Hellman group to use for key exchange during phase 1:

i. Use the **?** to determine available Diffie-Hellman group types:

```
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)>
dh_group ?
curve25519
curve448
ecp192
ecp224
...
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)>
```

ii. Set the Diffie-Hellman group type:

```
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)>
dh_group value
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)>
```

The default is **modp2048**.

v. (Optional) Add additional phase 1 proposals:

i. Move back one level in the schema:

```
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)>
..
(config vpn ipsec tunnel ipsec_example ike phase1_proposal)>
```

ii. Add an additional proposal:

```
(config vpn ipsec tunnel ipsec_example ike phase1_proposal)>
add end
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 1)>
```

Repeat the above steps to set the type of encryption, hash, and Diffie-Hellman group for the additional proposal.

iii. Repeat to add more phase 1 proposals.

j. Configure the types of encryption, hash, and Diffie-Hellman group to use during phase 2:

i. Move back two levels in the schema:

```
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)> ..
..
(config vpn ipsec tunnel ipsec_example ike)>
```

ii. Add a phase 2 proposal:

```
(config vpn ipsec tunnel ipsec_example ike)> add ike phase2_
proposal end
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)>
```

iii. Set the type of encryption to use during phase 2:

```
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)>
cipher value
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)>
```

where *value* is one of **3des**, **aes128**, **aes192**, **aes256**, or **null**. The default is **3des**.

iv. Set the type of hash to use during phase 2 to verify communication integrity:

```
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)>
hash value
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)>
```

where *value* is one of **md5**, **sha1**, **sha256**, **sha384**, or **sha512**. The default is **sha1**.

v. Set the type of Diffie-Hellman group to use for key exchange during phase 2:

i. Use the **?** to determine available Diffie-Hellman group types:

```
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)>
dh_group ?
curve25519
curve448
ecp192
ecp224
...
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)>
```

ii. Set the Diffie-Hellman group type:

```
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)>
dh_group value
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)>
```

The default is **modp2048**.

vi. (Optional) Add additional phase 2 proposals:

i. Move back one level in the schema:

```
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)>
..
(config vpn ipsec tunnel ipsec_example ike phase2_proposal)>
```

ii. Add an additional proposal:

```
(config vpn ipsec tunnel ipsec_example ike phase2_proposal)>
add end
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 1)>
```

Repeat the above steps to set the type of encryption, hash, and Diffie-Hellman group for the additional proposal.

iii. Repeat to add more phase 2 proposals.

16. (Optional) Configure dead peer detection:

Dead peer detection is enabled by default. Dead peer detection uses periodic IKE transmissions to the remote endpoint to detect whether tunnel communications have failed, allowing the tunnel to be automatically restarted when failure occurs.

a. Change to the root of the configuration schema:

```
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)> ...
(config)>
```

b. To disable dead peer detection:

```
(config)> vpn ipsec tunnel ipsec_example dpd enable false
(config)>
```

c. Set the number of seconds between transmissions of dead peer packets. Dead peer packets are only sent when the tunnel is idle. The default is **60**.

```
(config)> vpn ipsec tunnel ipsec_example dpd delay value
(config)>
```

d. Set the number of seconds to wait for a response from a dead peer packet before assuming the tunnel has failed. The default is **90**.

```
(config)> vpn ipsec tunnel ipsec_example dpd timeout value
(config)>
```

17. (Optional) Create a list of destination networks that require source NAT:

a. Add a destination network:

```
(config)> add vpn ipsec tunnel ipsec_example nat end
(config vpn ipsec tunnel ipsec_example nat 0)>
```

b. Set the IPv4 address and optional netmask of a destination network that requires source NAT. You can also use **any**, meaning that any destination network connected to the tunnel will use source NAT.

```
(config vpn ipsec tunnel ipsec_example nat 0)> dst value
(config vpn ipsec tunnel ipsec_example nat 0)>
```

18. Configure policies that define the network traffic that will be encapsulated by this tunnel:
    a. Change to the root of the configuration schema:

```
(config vpn ipsec tunnel ipsec_example nat 0)> ...
(config)>
```

    b. Add a policy:

```
(config)> add vpn ipsec tunnel ipsec_example policy end
(config vpn ipsec tunnel ipsec_example policy 0)>
```

    c. Set the type of local traffic selector:

```
(config vpn ipsec tunnel ipsec_example policy 0)> local type value
(config vpn ipsec tunnel ipsec_example policy 0)>
```

    where *value* is one of:

    - **address**: The address of a local network interface.

      Set the address:

      i. Use the **?** to determine available interfaces:
      ii. Set the interface. For example:

```
(config vpn ipsec tunnel ipsec_example policy 0)> local
address eth1
(config vpn ipsec tunnel ipsec_example policy 0)>
```

    - **network**: The subnet of a local network interface.

      Set the network:

      i. Use the **?** to determine available interfaces:
      ii. Set the interface. For example:

```
(config vpn ipsec tunnel ipsec_example policy 0)> local
network eth1
(config vpn ipsec tunnel ipsec_example policy 0)>
```

    - **custom**: A user-defined network.

      Set the custom network:

```
(config vpn ipsec tunnel ipsec_example policy 0)> local custom
value
(config vpn ipsec tunnel ipsec_example policy 0)>
```

      where *value* is the IPv4 address and optional netmask. The keyword **any** can also be used.

    - **request**: Requests a network from the remote peer.
    - **dynamic**: Uses the address of the local endpoint.

d. Set the port matching criteria for the local traffic selector:

```
(config vpn ipsec tunnel ipsec_example policy 0)> local port value
(config vpn ipsec tunnel ipsec_example policy 0)>
```

where *value* is the port number, a range of port numbers, or the keyword **any**.

e. Set the protocol matching criteria for the local traffic selector:

```
(config vpn ipsec tunnel ipsec_example policy 0)> local protocol value
(config vpn ipsec tunnel ipsec_example policy 0)>
```

where *value* is one of:

- **any**: Matches any protocol.
- **tcp**: Matches TCP protocol only.
- **udp**: Matches UDP protocol only.
- **icmp**: Matches ICMP requests only.
- **other**: Matches an unlisted protocol.

  If **other** is used, set the number of the protocol:

  ```
  (config vpn ipsec tunnel ipsec_example policy 0)> local
  protocol_other int
  (config vpn ipsec tunnel ipsec_example policy 0)>
  ```

  Allowed values are an integer between **1** and **255**.

f. Set the IP address and optional netmask of the remote traffic selector:

```
(config vpn ipsec tunnel ipsec_example policy 0)> remote network value
(config vpn ipsec tunnel ipsec_example policy 0)>
```

g. Set the port matching criteria for the remote traffic selector:

```
(config vpn ipsec tunnel ipsec_example policy 0)> remote port value
(config vpn ipsec tunnel ipsec_example policy 0)>
```

where *value* is the port number, a range of port numbers, or the keyword **any**.

h. Set the protocol matching criteria for the remote traffic selector:

```
(config vpn ipsec tunnel ipsec_example policy 0)> remote protocol
value
(config vpn ipsec tunnel ipsec_example policy 0)>
```

where *value* is one of:

- **any**: Matches any protocol.
- **tcp**: Matches TCP protocol only.
- **udp**: Matches UDP protocol only.
- **icmp**: Matches ICMP requests only.

- **other**: Matches an unlisted protocol.

  If **other** is used, set the number of the protocol:

  ```
  (config vpn ipsec tunnel ipsec_example policy 0)> remote
  protocol_other int
  (config vpn ipsec tunnel ipsec_example policy 0)>
  ```

  Allowed values are an integer between **1** and **255**.

19. (Optional) You can also configure various IPsec related time out, keep alive, and related values:

    a. Change to the root of the configuration schema:

    ```
    (config vpn ipsec tunnel ipsec_example policy 0)> ...
    (config)>
    ```

    b. Use the **?** to determine available options:

    ```
    (config)> vpn ipsec advanced ?

    Advanced: Advanced configuration that applies to all IPsec tunnels.

     Parameters                Current Value
     ------------------------------------------------------------------------
    ---------
     debug                     none           Debug level
     ike_fragment_size         1280           Maximum IKE fragment size
     ike_retransmit_tries      5              IKE retransmit tries
     keep_alive                40s            NAT keep alive time

     Additional Configuration
     ------------------------------------------------------------------------
    ----------
     connection_retry_timeout  Connection retry timeout
     connection_try_interval   Connection try interval
     ike_timeout               IKE timeout

    (config)>
    ```

    Generally, the default settings for these should be sufficient.

    c. You can also enable debugging for IPsec:

    ```
    (config)> vpn ipsec advanced debug value
    (config)>
    ```

    where *value* is one of:

    - **none**
    - **basic_auditing**
    - **detailed_control**
    - **generic_control**
    - **raw_data**
    - **sensitive_data**

20. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

21. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure IPsec failover

There are two methods to configure the IX15 device to fail over from a primary IPsec tunnel to a backup tunnel:

- **SureLink** active recovery—You can use SureLink along with the IPsec tunnel's metric to configure two or more tunnels so that when the primary tunnel is determined to be inactive by SureLink, a secondary tunnel can begin serving traffic that the primary tunnel was serving.
- **Preferred tunnel**—When multiple IPsec tunnels are configured, one tunnel can be configured as a backup to another tunnel by defining a preferred tunnel for the backup device.

Required configuration items

- Two or more configured IPsec tunnels: The primary tunnel, and one or more backup tunnels.
- Either:
  - SureLink configured on the primary tunnel with **Restart Interface** enabled, and the metric for all tunnels set appropriately to determine which IPsec tunnel has priority. With this failover configuration, both tunnels are active simultaneously, and there is minimal downtime due to failover.
  - Identify the preferred tunnel during configuration of the backup tunnel. In this scenario, the backup tunnel is not active until the preferred tunnel fails.

IPsec failover using SureLink

With this configuration, when two IPsec tunnels are configured with the same local and remote endpoints but different metrics, traffic addressed to the remote endpoint will be routed through the IPsec tunnel with the lower metric.

If **SureLink** > **Restart Interface** is enabled for the tunnel with the lower metric, and SureLink determines that the tunnel is not functioning properly (for example, pings to a host at the other end of the tunnel are failing), then:

1. SureLink will shut down the tunnel and renegotiate its IPsec connection.
2. While the tunnel with the lower metric is down, traffic addressed to the remote endpoint will be routed through the tunnel with the higher metric.

For example:

- Tunnel_1:
  - **Metric**: 10
  - **Local endpoint** > **Interface**: ETH
  - **Remote endpoint** > **Hostname**: 192.168.10.1
  - **SureLink** configuration:
    - **Restart Interface** enabled
    - **Test target**:
      - **Test type**: Ping test
      - **Ping host**: 192.168.10.2
- Tunnel_2:

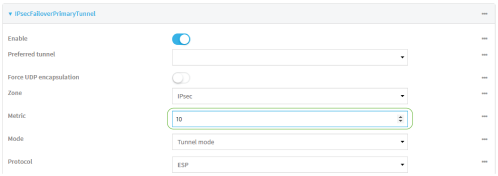
- **Metric**: 20
- **Local endpoint** > **Interface**: ETH
- **Remote endpoint** > **Hostname**: 192.168.10.1

In this configuration:

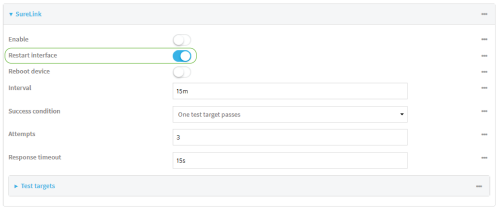
1. Tunnel_1 will normally be used for traffic destined for the 192.168.10.1 endpoint.
2. If pings to 192.168.10.2 fail, SureLink will shut down the tunnel and renegotiate its IPsec connection.
3. While Tunnel_1 is down, Tunnel_2 will be used for traffic destined for the 192.168.10.1 endpoint.

WebUI

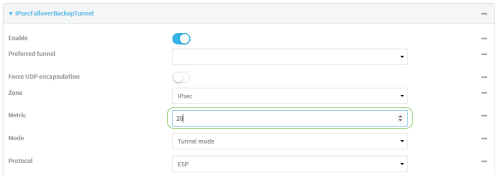
1. Configure the primary IPsec tunnel. See Configure an IPsec tunnel for instructions.
   - During configuration of the IPsec tunnel, set the metric to a low value (for example, **10**).



   - Configure SureLink for the primary IPsec tunnel and enable **Restart interface**. See Configure SureLink active recovery for IPsec for instructions.



2. Create a backup IPsec tunnel. Configure this tunnel to use the same local and remote endpoints as the primary tunnel. See Configure an IPsec tunnel for instructions.
   - During configuration of the IPsec tunnel, set the metric to a value that is higher than the metric of the primary tunnel (for example, **20**).



Command line

1. Configure the primary IPsec tunnel. See Configure an IPsec tunnel for instructions.
   - During configuration of the IPsec tunnel, set the metric to a low value (for example, **10**):

```
(config vpn ipsec tunnel IPsecFailoverPrimaryTunnel)> metric 10
(config vpn ipsec tunnel IPsecFailoverPrimaryTunnel)>
```

   - Configure SureLink for the primary IPsec tunnel and enable **Restart interface**. See Configure SureLink active recovery for IPsec for instructions.

```
(config vpn ipsec tunnel IPsecFailoverPrimaryTunnel)> surelink
restart true
(config vpn ipsec tunnel IPsecFailoverPrimaryTunnel)>
```

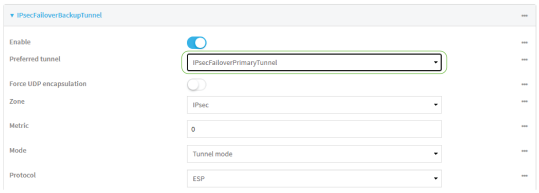
2. Create a backup IPsec tunnel. Configure this tunnel to use the same local and remote endpoints as the primary tunnel. See Configure an IPsec tunnel for instructions.
   - During configuration of the IPsec tunnel, set the metric to a value that is higher than the metric of the primary tunnel (for example, **20**):

```
(config vpn ipsec tunnel IPsecFailoverBackupTunnel)> metric 20
(config vpn ipsec tunnel IPsecFailoverBackupTunnel)>
```

IPsec failover using Preferred tunnel

WebUI

1. Configure the primary IPsec tunnel. See Configure an IPsec tunnel for instructions.
2. Create a backup IPsec tunnel. See Configure an IPsec tunnel for instructions.
3. During configuration of the backup IPsec tunnel, identify the primary IPsec tunnel in the **Preferred tunnel** parameter:



Command line

1. Configure the primary IPsec tunnel. See Configure an IPsec tunnel for instructions.
2. Create a backup IPsec tunnel. See Configure an IPsec tunnel for instructions.
3. During configuration of the backup IPsec tunnel, identify the primary IPsec tunnel:
   a. Use the **?** to view a list of available tunnels:

```
(config vpn ipsec tunnel backup_ipsec_tunnel)> ipsec_failover ?

Preferred tunnel: This tunnel will not start until the preferred
tunnel has failed. It will continue
to operate until the preferred tunnel returns to full operation
```

```
status.
Format:
 primary_ipsec_tunnel
 backup_ipsec_tunnel
Optional: yes
Current value:

(config vpn ipsec tunnel backup_ipsec_tunnel)> ipsec_failover
```

b. Set the primary IPsec tunnel:

```
(config vpn ipsec tunnel backup_ipsec_tunnel)> ipsec_failover primary_
ipsec_tunnel
(config vpn ipsec tunnel backup_ipsec_tunnel)>
```

Configure SureLink active recovery for IPsec

You can configure the IX15 device to regularly probe IPsec tunnels to determine if the connection has failed and take remedial action.

You can also configure the IPsec tunnel to fail over to a backup tunnel. See Configure IPsec failover for further information.

Required configuration items

- A valid IPsec configuration. See Configure an IPsec tunnel for configuration instructions.
- Enable IPsec active recovery.
- The behavior of the IX15 device upon IPsec failure: either
  - Restart the IPsec interface
  - Reboot the device.

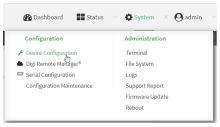
Additional configuration items

- The interval between connectivity tests.
- Whether the interface should be considered to have failed if one of the test targets fails, or all of the test targets fail.
- The number of probe attempts before the IPsec connection is considered to have failed.
- The amount of time that the device should wait for a response to a probe attempt before considering it to have failed.

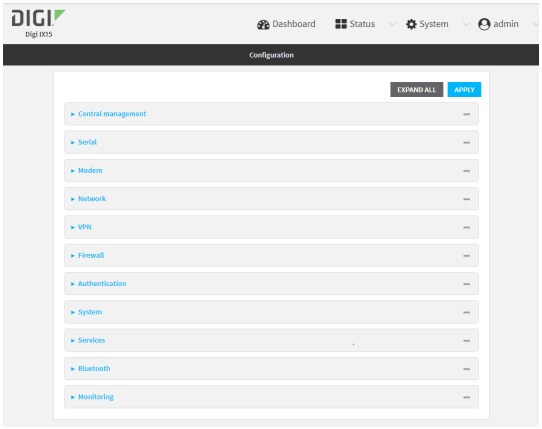
To configure the IX15 device to regularly probe the IPsec connection:

WebUI

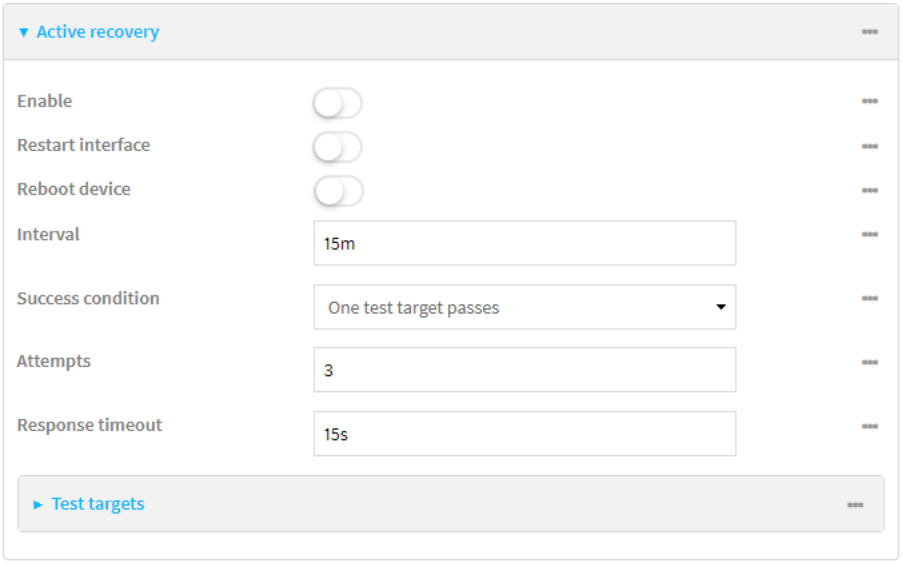
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **VPN** > **IPsec**.
4. Create a new IPsec tunnel or select an existing one:
   - To create a new IPsec tunnel, see Configure an IPsec tunnel.
   - To edit an existing IPsec tunnel, click to expand the appropriate tunnel.
5. After creating or selecting the IPsec tunnel, click **Active recovery**.



6. **Enable** active recovery.

7. For **Restart interface**, enable to configure the device to restart the interface when its connection is considered to have failed. This is useful for interfaces that may regain connectivity after restarting, such as a cellular modem.
8. For **Reboot device**, enable to instruct the device to reboot when the WAN connection is considered to have failed.
9. Change the **Interval** between connectivity tests.

   Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

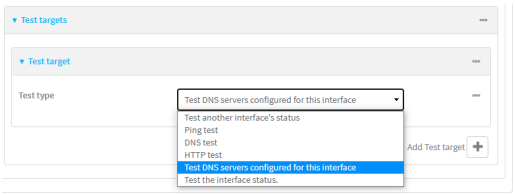
   For example, to set **Interval** to ten minutes, enter **10m** or **600s**.

   The default is 15 minutes.
10. For **Success condition**, determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets.
11. For **Attempts**, type the number of probe attempts before the WAN is considered to have failed.
12. For **Response timeout**, type the amount of time that the device should wait for a response to a probe attempt before considering it to have failed.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Response timeout** to ten minutes, enter **10m** or **600s**.

    The default is 15 seconds.
13. Add a test target:
    a. Click to expand **Test targets**.
    b. For **Add Test target**, click **+**.
    c. Select the **Test type**:



- **Test another interface's status**: Allows you to test another interface's status, to create a failover or coupled relationship between interfaces. If **Test another interface's status** is selected:
  - For **Test Interface**, select the alternate interface to be tested.
  - For **IP version**, select the alternate interface's IP version. This allows you to determine the alternate interface's status for a particular IP version.
  - For **Expected status**, select whether the expected status of the alternate interface is **Up** or **Down**. For example, if **Expected status** is set to **Down**, but the alternate interface is determined to be up, then this test will fail.
- **Ping test**: Tests connectivity by sending an ICMP echo request to the hostname or IP address specified in **Ping host**. You can also optionally change the number of bytes in the **Ping payload size**.
- **DNS test**: Tests connectivity by sending a DNS query to the specified **DNS server**.

- **HTTP test**: Tests connectivity by sending an HTTP or HTTPS GET request to the URL specified in **Web servers**. The URL should take the format of **http[s]://*hostname*/[*path*]**.
- **Test DNS servers configured for this interface**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
- **Test the interface status**: The interface is considered to be down based on:
  - **Down time**: The amount of time that the interface can be down before this test is considered to have failed.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Down time** to ten minutes, enter **10m** or **600s**.

    The default is 60 seconds.
  - **Initial connection time**: The amount of time to wait for an initial connection to the interface before this test is considered to have failed.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Initial connection time** to ten minutes, enter **10m** or **600s**.

    The default is 60 seconds.

14. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Create a new IPsec tunnel, or edit an existing one:
   - To create a new IPsec tunnel, see Configure an IPsec tunnel.
   - To edit an existing IPsec tunnel, change to the IPsec tunnel's node in the configuration schema. For example, for an IPsec tunnel named **ipsec_example**, change to the **ipsec_example** node in the configuration schema:

```
(config)> vpn ipsec tunnel ipsec_example
(config vpn ipsec tunnel ipsec_example)>
```

4. Enable active recovery:

```
(config vpn ipsec tunnel ipsec_example)> surelink enable true
(config vpn ipsec tunnel ipsec_example)>
```

5. To configure the device to restart the interface when its connection is considered to have failed:

```
(config vpn ipsec tunnel ipsec_example)> surelink restart true
(config vpn ipsec tunnel ipsec_example)>
```

   This is useful for interfaces that may regain connectivity after restarting, such as a cellular modem.

6. To configure the device to reboot when the interface is considered to have failed:

```
(config vpn ipsec tunnel ipsec_example)> surelink reboot enable
(config vpn ipsec tunnel ipsec_example)>
```

7. Set the **Interval** between connectivity tests:

```
(config vpn ipsec tunnel ipsec_example)> surelink interval value
(config vpn ipsec tunnel ipsec_example)>
```

   where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

   For example, to set **interval** to ten minutes, enter either **10m** or **600s**:

```
(config vpn ipsec tunnel ipsec_example)> surelink interval 600s
(config vpn ipsec tunnel ipsec_example)>
```

   The default is 15 minutes.

8. Determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets:

```
(config vpn ipsec tunnel ipsec_example)> surelink success_condition value
(config vpn ipsec tunnel ipsec_example)>
```

   Where *value* is either **one** or **all**.

9. Set the number of probe attempts before the WAN is considered to have failed:

```
(config vpn ipsec tunnel ipsec_example)> surelink  attempts num
(config vpn ipsec tunnel ipsec_example)>
```

   The default is **3**.

10. Set the amount of time that the device should wait for a response to a probe attempt before considering it to have failed:

```
(config vpn ipsec tunnel ipsec_example)> surelink timeout value
(config vpn ipsec tunnel ipsec_example)>
```

    where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **timeout** to ten minutes, enter either **10m** or **600s**:

```
(config vpn ipsec tunnel ipsec_example)> surelink timeout 600s
(config vpn ipsec tunnel ipsec_example)>
```

The default is 15 seconds.

11. Configure test targets:
    a. Add a test target:

    ```
    (config vpn ipsec tunnel ipsec_example)> add surelink target end
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    ```

    b. Set the test type:

    ```
    (config vpn ipsec tunnel ipsec_example surelink target 0)> test value
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    ```

    where *value* is one of:

    - **ping**: Tests connectivity by sending an ICMP echo request to a specified hostname or IP address.
      - Specify the hostname or IP address:

      ```
      (config vpn ipsec tunnel ipsec_example surelink target 0)>
      ping_host host
      (config vpn ipsec tunnel ipsec_example surelink target 0)>
      ```

      - (Optional) Set the size, in bytes, of the ping packet:

      ```
      (config vpn ipsec tunnel ipsec_example surelink target 0)>
      ping_size [num]
      (config vpn ipsec tunnel ipsec_example surelink target 0)>
      ```

    - **dns**: Tests connectivity by sending a DNS query to the specified DNS server.
      - Specify the DNS server. Allowed value is the IP address of the DNS server.

      ```
      (config vpn ipsec tunnel ipsec_example surelink target 0)>
      dns_server ip_address
      (config vpn ipsec tunnel ipsec_example surelink target 0)>
      ```

    - **dns_configured**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
    - **http**: Tests connectivity by sending an HTTP or HTTPS GET request to the specified URL.
      - Specify the url:

      ```
      (config vpn ipsec tunnel ipsec_example surelink target 0)>
      http_url value
      (config vpn ipsec tunnel ipsec_example surelink target 0)>
      ```

      where *value* uses the format **http[s]://*hostname*/[*path*]**

- **interface_up**: The interface is considered to be down based on the interfaces down time, and the amount of time an initial connection to the interface takes before this test is considered to have failed.
  - (Optional) Set the amount of time that the interface can be down before this test is considered to have failed:

    ```
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    interface_down_time value
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    ```

    where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

    For example, to set **timeout** to ten minutes, enter either **10m** or **600s**:

    ```
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    interface_down_time 600s
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    ```

    The default is 60 seconds.
  - (Optional) Set the amount of time to wait for an initial connection to the interface before this test is considered to have failed:

    ```
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    interface_timeout value
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    ```

    where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

    For example, to set **timeout** to ten minutes, enter either **10m** or **600s**:

    ```
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    interface_timeout 600s
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    ```

    The default is 60 seconds.
  - **other**: Allows you to test another interface's status, to create a failover or coupled relationship between interfaces:

    ```
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    other value
    (config vpn ipsec tunnel ipsec_example surelink target 0)>
    ```

    If **other** is set:

- Set the alternate interface to be tested:
  i. Use the **?** to determine available interfaces:
  ii. Set the interface. For example:

```
(config vpn ipsec tunnel ipsec_example surelink
target 0)> other_interface /network/interface/eth1
(config vpn ipsec tunnel ipsec_example surelink
target 0)>
```

- Set the alternate interface's IP version. This allows you to determine the alternate interface's status for a particular IP version.

```
(config vpn ipsec tunnel ipsec_example surelink target 0)>
other_ip_version value
(config vpn ipsec tunnel ipsec_example surelink target 0)>
```

  where *value* is one of: **any**, **both**, **ipv4**, or **ipv6**.
- Set the expected status of the alternate interface:

```
(config vpn ipsec tunnel ipsec_example surelink target 0)>
other_status value
(config vpn ipsec tunnel ipsec_example surelink target 0)>
```

  where *value* is either **up** or **down**. For example, if **other_status** is set to **down**, but the alternate interface is determined to be up, then this test will fail.

12. Save the configuration and apply the change:

```
(config vpn ipsec tunnel ipsec_example connection_monitor target 0)> save
Configuration saved.
>
```

13. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show IPsec status and statistics

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, select **Status** > **IPsec**.

   The **IPsec** page appears.
3. To view configuration details about an IPsec tunnel, click the (configuration) icon in the upper right of the tunnel's status pane.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. To display details about all configured IPsec tunnels, type the following at the prompt:

```
> show ipsec all

 Name    Enable  Status   Hostname
 ------  ------  -------  ---------------
 ipsec1  true    up       192.168.2.1
 vpn1    false   pending  192.168.3.1

>
```

3. To display details about a specific tunnel:

```
> show ipsec tunnel ipsec1

 Tunnel                  : ipsec1
 Enable                  : true
 Status                  : pending
 Hostname                : 192.168.2.1
 Zone                    : ipsec
 Mode                    : tunnel
 Type                    : esp

>
```

4. Type **exit** to exit the Admin CLI.

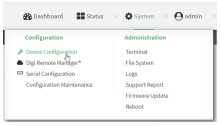
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Debug an IPsec configuration

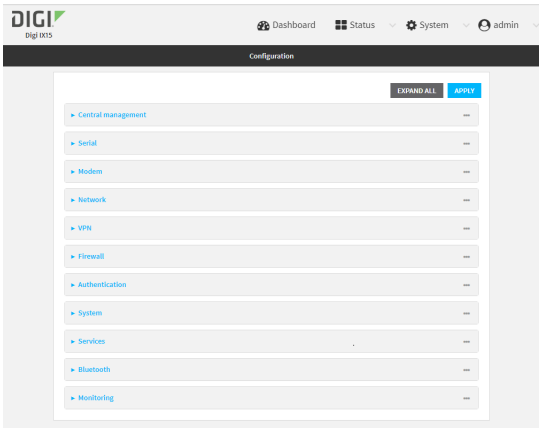
If you experience issues with an IPsec tunnel not being successfully negotiated with the remote end of the tunnel, you can enable IPsec debug messages to be written to the system log. See View system and event logs for more information about viewing the system log.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **VPN** > **IPsec**.
4. Click to expand **Advanced**.
5. For **Debug level**, select one of the following:
   - **Disable debug messages**.
   - **Basic auditing debug**: Logs basic auditing information, (for example, SA up/SA down).
   - **Generic control flow** : Select this for basic debugging information.
   - **Detailed control flow** : More detailed debugging control flow.
   - **Raw data**: Includes raw data dumps in hexadecimal format.
   - **Sensitive material**: Also includes sensitive material in dumps (for example, encryption keys).
6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Set the IPsec debug value:

```
config> vpn ipsec advanced debug value
config>
```

   where *value* is one of:

- **none**. (Default) No debug messages are written.
- **basic_auditing**: Logs basic auditing information, (for example, SA up/SA down).
- **generic_control**: Select this for basic debugging information.
- **detailed_control**: More detailed debugging control flow.
- **raw_data**: Includes raw data dumps in hexadecimal format.
- **sensitive_data**: Also includes sensitive material in dumps (for example, encryption keys).

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure a Simple Certificate Enrollment Protocol client

Simple Certificate Enrollment Protocol (SCEP) is a mechanism that allows for large-scale X.509 certificate deployment. You can configure IX15 device to function as a SCEP client that will connect to a SCEP server that is used to sign Certificate Signing Requests (CSRs), provide Certificate Revocation Lists (CRLs), and distribute valid certificates from a Certificate Authority (CA).

Required configuration

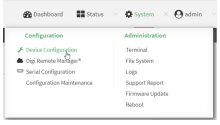
- Enable the SCEP client.
- The fully-qualified domain name of the SCEP server to be used for certificate requests.
- The challenge password provided by the SCEP server that the SCEP client will use when making SCEP requests.
- The distinguished name to be used for the CSR.
- The file name of the Certificate Revocation List (CRL) from the Certificate Authority (CA).

Additional configuration

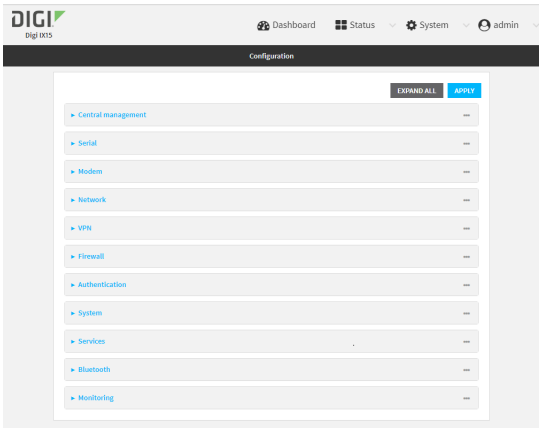
- The number of days that the certificate enrollment can be renewed, prior to the request expiring.

WebUI

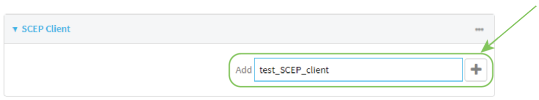
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



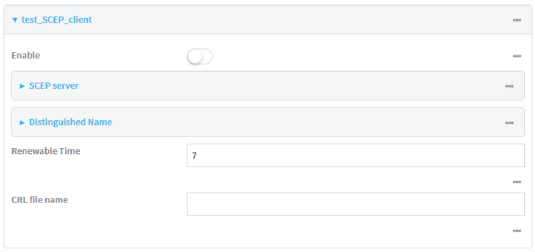
The **Configuration** window is displayed.



3. Click **Network** > **SCEP Client**.
4. For **Add clients**, enter a name for the SCEP client and click **+**.



The new SCEP client configuration is displayed.



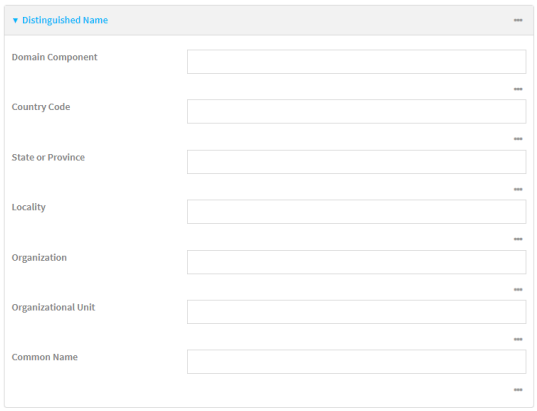
5. Click **Enable** to enable the SCEP client.
6. For **Renewable Time**, type the number of days that the certificate enrollment can be renewed, prior to the request expiring. This value is configured on the SCEP server, and is used by the IX15 device to determine when to start attempting to auto-renew an existing certificate. The default is **7**.
7. (Optional) For CRL file name, type the filename of the Certificate Revocation List (CRL) from the CA.

   The CRL is stored on the IX15 device in the /etc/config/scep_client/*client_name* directory.
8. Click to expand **SCEP server**.



9. For **FQDN**, type the fully qualified domain name or IP address of the SCEP server.

10. (Optiona) For **CA identity**, type a string that will be understood by the certificate authority. For example, it could be a domain name or a user name. If the certificate authority has multiple CA certificates, this field can be used to distinguish which is required.
11. For **Path**, Type the HTTP URL path required for accessing the certificate authority. You should leave this option at the default of **/cgi-bin/pkiclient.exe** unless directed by the CA to use another path.
12. For **Password**, type the challenge password as configured on the SCEP server.
13. Click to expand **Distinguished Name**.



14. Type the value for each appropriate Distinguished Name attribute.
15. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a new SCEP client:

```
(config)> add network scep_client scep_client_name
(config network scep_client scep_client_name
)>
```

4. Enable the SCEP client:

```
(config network scep_client scep_client_name)> enable true
(config network scep_client scep_client_name)>
```

5. Set the url parameter to the fully qualified domain name or IP address of the SCEP server:

```
(config network scep_client scep_client_name)> server url
https://scep.example.com
(config network scep_client scep_client_name)>
```

6. (Optiona) Set a CA idenity string that will be understood by the certificate authority. For example, it could be a domain name or a user name. If the certificate authority has multiple CA certificates, this field can be used to distinguish which is required.

```
(config network scep_client scep_client_name)> server ca_ident string
(config network scep_client scep_client_name)>
```

7. Set the HTTP URL path required for accessing the certificate authority. You should leave this option at the default of **/cgi-bin/pkiclient.exe** unless directed by the CA to use another path.

```
(config network scep_client scep_client_name)> server path path
(config network scep_client scep_client_name)>
```

8. Set the challenge password as configured on the SCEP server:

```
(config network scep_client scep_client_name)> server password challenge_
password
(config network scep_client scep_client_name)>
```

9. Set Distinguished Name attributes:
   a. Set the Domain Component:

```
(config network scep_client scep_client_name)> distinguished_name dc
value
(config network scep_client scep_client_name)>
```

   b. Set the two letter Country Code:

```
(config network scep_client scep_client_name)> distinguished_name c
value
(config network scep_client scep_client_name)>
```

   c. Set the State or Province:

```
(config network scep_client scep_client_name)> distinguished_name st
value
(config network scep_clientscep_client_name )>
```

   d. Set the Locality:

```
(config network scep_client scep_client_name)> distinguished_name l
value
(config network scep_client scep_client_name)>
```

e. Set the Organization:

```
(config network scep_client scep_client_name)> distinguished_name o
value
(config network scep_client scep_client_name)>
```

f. Set the Organizational Unit:

```
(config network scep_client scep_client_name)> distinguished_name ou
value
(config network scep_client scep_client_name)>
```

g. Set the Common Name:

```
(config network scep_client scep_client_name)> distinguished_name cn
value
(config network scep_client scep_client_name)>
```

10. Set the number of days that the certificate enrollment can be renewed, prior to the request expiring. This value is configured on the SCEP server, and is used by the IX15 device to determine when to start attempting to auto-renew an existing certificate. The default is **7**.

```
(config network scep_client scep_client_name)> renewable_time integer
(config network scep_client scep_client_name)>
```

11. (Optional) Set the filename of the Certificate Revocation List (CRL) from the CA.

    The CRL is stored on the IX15 device in the /etc/config/scep_client/*client_name* directory.

```
(config network scep_client scep_client_name)> crl_name name
(config network scep_client scep_client_name)>
```

12. Save the configuration and apply the change:

```
(config network scep_client scep_client_name)> save
Configuration saved.
>
```

13. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Example: SCEP client configuration with Fortinet SCEP server

In this example configuration, we will configure the IX15 device as a SCEP client that will connect to a Fortinet SCEP server.

Fortinet configuration

On the Fortinet server:

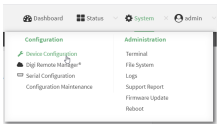
1. Enable ports for SCEP services:
   a. From the menu, select **Network** > **Interfaces**.
   b. Select the appopriate port and click **Edit**.
   c. For **Access Rights** > **Services**, enable the following services:
      - **HTTPS** > **SCEP**
      - **HTTPS** > **CRL Downloads**
      - **HTTP** > **SCEP**
      - **HTTP** > **CRL Downloads**
   d. The remaining fields can be left at their defaults or changed as appropriate.
   e. Click **OK**.
2. Create a Certificate Authority (CA):
   a. From the menu, click **Certificate Authorities** > **Local CAs**.
   b. Click **Create New**.
   c. Type a **Certificate ID** for the CA, for example, **fortinet_example_ca**.
   d. Complete the **Subject Information** fields.
   e. The remaining fields can be left at their defaults or changed as appropriate.
   f. Click **OK**.
3. Edit SCEP settings:
   a. From the menu, click **SCEP** > **General**.
   b. Click **Enable SCEP** if it is not enabled.
   c. For **Default enrollment password**, enter a password. The password entered here must correspond to the challenge password configured for the SCEP client on the IX15 device.
   d. The remaining fields can be left at their defaults or changed as appropriate.
   e. Click **OK**.
4. Create an **Enrollment Request**:
   a. From the menu, click **SCEP** > **Enrollment Requests**.
   b. Click **Create New**.
   c. For **Automatic request type**, select **Wildcard**.
   d. For **Certificate authority**, select the CA created in step 1, above.
   e. Complete the **Subject Information** fields. The Distinguished Name (DN) attributes entered here must correspond to the Distinguished Name attributes configured for the SCEP client on the IX15 device.
   f. For **Renewal** > **Allow renewal *x* days before the certified is expired**, type the number of days that the certificate enrollment can be renewed, prior to the request expiring. The **Renewable Time** setting on the IX15 device must match the setting of this parameter.
   g. The remaining fields can be left at their defaults or changed as appropriate.
   h. Click **OK**.

IX15 configuration

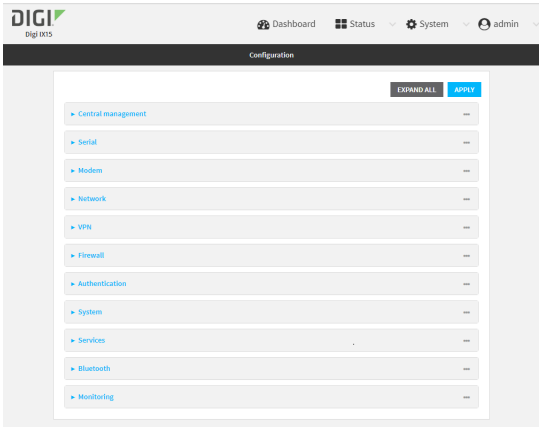
On the IX15 device:

WebUI

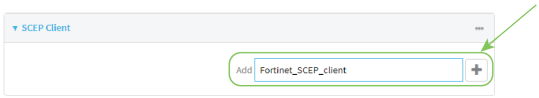
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



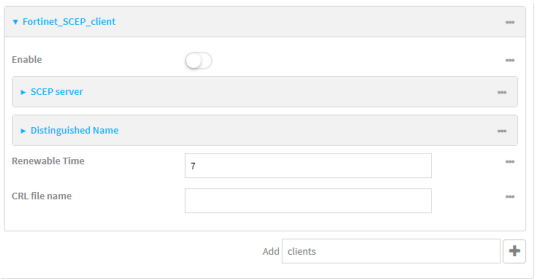
The **Configuration** window is displayed.



3. Click **Network** > **SCEP Client**.
4. For **Add clients**, enter a name for the SCEP client and click ✚.

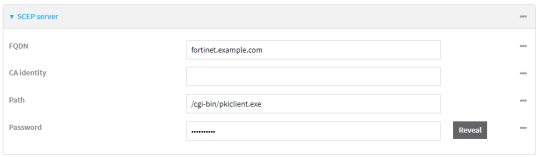


The new SCEP client configuration is displayed.

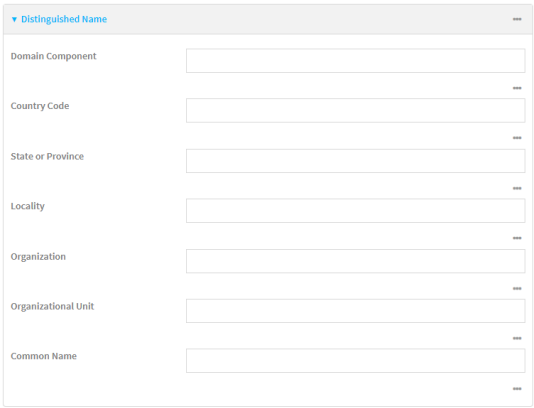


5. Click **Enable** to enable the SCEP client.
6. For **Renewable Time**, type the number of days that the certificate enrollment can be renewed, prior to the request expiring. This value must match the setting of the **Allow renewal *x* days before the certified is expired** option on the Fortinet server.
7. (Optional) For CRL file name, type the filename of the Certificate Revocation List (CRL) from the CA. The filename of the CRL corresponds to the Certificate ID of the CA created on the Fortinet server, for example, **fortinet_example_ca.crl**.

8. Click to expand **SCEP server**.



9. For **FQDN**, type the fully qualified domain name or IP address of the Fortinet server.
10. For **Password**, type the challenge password. This corresponds to the **Default enrollment password** on the Fortinet server.
11. Click to expand **Distinguished Name**.



12. Type the value for each appropriate Distinguished Name attribute. The values entered here must correspond to the DN attributes in the **Enrollment Request** on the Fortinet server.
13. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a new SCEP client, for example, **Fortinet_SCEP_client**:

```
(config)> add network scep_client Fortinet_SCEP_client
(config network scep_client Fortinet_SCEP_client
)>
```

4. Enable the SCEP client:

```
(config network scep_client Fortinet_SCEP_client)> enable true
(config network scep_client Fortinet_SCEP_client)>
```

5. Set the url parameter to the fully qualified domain name or IP address of the SCEP server:

```
(config network scep_client Fortinet_SCEP_client)> server url
https://fortinet.example.com
(config network scep_client Fortinet_SCEP_client)>
```

6. Set the challenge password as configured on the SCEP server. This corresponds to the **Default enrollment password** on the Fortinet server.

```
(config network scep_client Fortinet_SCEP_client)> server password
challenge_password
(config network scep_client Fortinet_SCEP_client)>
```

7. Set Distinguished Name attributes. The values entered here must correspond to the DN attributes in the **Enrollment Request** on the Fortinet server.
   a. Set the Domain Component:

```
(config network scep_client Fortinet_SCEP_client)> distinguished_name
dc value
(config network scep_client Fortinet_SCEP_client)>
```

   b. Set the two letter Country Code:

```
(config network scep_client Fortinet_SCEP_client)> distinguished_name
c value
(config network scep_client Fortinet_SCEP_client)>
```

   c. Set the State or Province:

```
(config network scep_client Fortinet_SCEP_client)> distinguished_name
st value
(config network scep_client  Fortinet_SCEP_client)>
```

d. Set the Locality:

```
(config network scep_client Fortinet_SCEP_client)> distinguished_name
l value
(config network scep_client Fortinet_SCEP_client)>
```

e. Set the Organization:

```
(config network scep_client Fortinet_SCEP_client)> distinguished_name
o value
(config network scep_client Fortinet_SCEP_client)>
```

f. Set the Organizational Unit:

```
(config network scep_client Fortinet_SCEP_client)> distinguished_name
ou value
(config network scep_client Fortinet_SCEP_client)>
```

g. Set the Common Name:

```
(config network scep_client Fortinet_SCEP_client)> distinguished_name
cn value
(config network scep_client Fortinet_SCEP_client)>
```

8. Set the number of days that the certificate enrollment can be renewed, prior to the request expiring. This value must match the setting of the **Allow renewal *x* days before the certified is expired** option on the Fortinet server.

```
(config network scep_client Fortinet_SCEP_client)> renewable_time integer
(config network scep_client Fortinet_SCEP_client)>
```

9. (Optional) Set the filename of the Certificate Revocation List (CRL) from the CA.

   The CRL is stored on the IX15 device in the /etc/config/scep_client/*client_name* directory.

```
(config network scep_client Fortinet_SCEP_client)> crl_name name
(config network scep_client Fortinet_SCEP_client)>
```

10. Save the configuration and apply the change:

```
(config network scep_client Fortinet_SCEP_client)> save
Configuration saved.
>
```

11. Type **exit** to exit the Admin CLI.

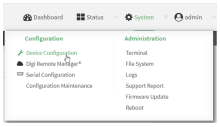
    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Disable hardware cryptographic acceleration

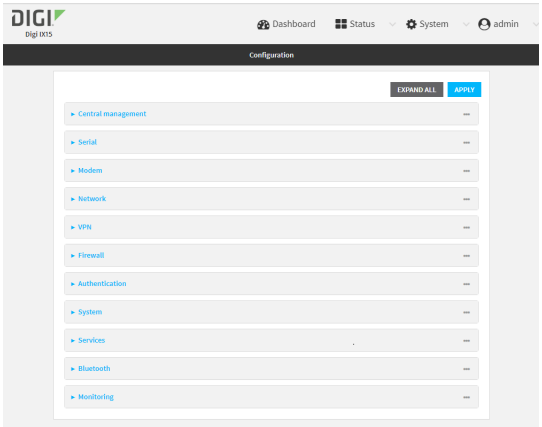
If you are experiencing problems when using IPSEC, such as the kernel crashing or unexpected package loss, disabling hardware cryptographic acceleration may correct the problem.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Select **System**.
4. Click to disable **HW crypto acceleration**.
5. Click **Apply** to save the configuration and apply the change.



The device must be rebooted for the change to take effect. See Reboot your IX15 device.

Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Disable hardware cryptographic acceleration:

```
(config)> system hycrypto false
>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

The device must be rebooted for the change to take effect. See Reboot your IX15 device.

OpenVPN

OpenVPN is an open-source Virtual Private Network (VPN) technology that creates secure point-to-point or site-to-site connections in routed or bridged configurations. OpenVPN uses a custom security protocol that is Secure Socket Layer (SSL) / Transport Layer Security (TLS) for key exchange. It uses standard encryption and authentication algorithms for data privacy and authentication over TCP or UDP.

The OpenVPN server can push the network configuration, such as the topology and IP routes, to OpenVPN clients. This makes OpenVPN simpler to configure as it reduces the chances of a configuration mismatch between the client and server. OpenVPN also supports cipher negotiation between the client and server. This means you can configure the OpenVPN server and clients with a range of different cipher options and the server will negotiate with the client on the cipher to use for the connection.

For more information on OpenVPN, see www.openvpn.net.

OpenVPN modes:

There are two modes for running OpenVPN:

- Routing mode, also known as TUN.
- Bridging mode, also known as TAP.

Routing (TUN) mode

In routing mode, each OpenVPN client is assigned a different IP subnet from the OpenVPN server and other OpenVPN clients. OpenVPN clients use Network Address Translation (NAT) to route traffic from devices connected on its LAN interfaces to the OpenVPN server.

The manner in which the IP subnets are defined depends on the OpenVPN topology in use. The IX15 device supports two types of OpenVPN topology:

OpenVPN Topology	Subnet definition method
net30	Each OpenVPN client is assigned a **/30** subnet within the IP subnet specified in the OpenVPN server configuration. With net30 topology, pushed routes are used, with the exception of the default route. Automatic route pushing (exec) is not allowed, because this would not inform the firewall and would be blocked.
subnet	Each OpenVPN client connected to the OpenVPN server is assigned an IP address within the IP subnet specified in the OpenVPN server configuration. For the IX15 device, pushed routes are not allowed; you will need to manually configure routes on the device.

For more information on OpenVPN topologies, see OpenVPN topology.

Bridging (TAP) mode

In bridging mode, a LAN interface on the OpenVPN server is assigned to OpenVPN. The LAN interfaces of the OpenVPN clients are on the same IP subnet as the OpenVPN server's LAN interface. This means that devices connected to the OpenVPN client's LAN interface are on the same IP subnet as devices. The IX15 device supports two mechanisms for configuring an OpenVPN server in TAP mode:

- OpenVPN managed—The IX15 device creates the interface and then uses its standard configuration to set up the connection (for example, its standard DHCP server configuration).
- Device only—IP addressing is controlled by the system, not by OpenVPN.

Additional OpenVPN information

For more information on OpenVPN, see these resources:

Bridging vs. routing

OpenVPN/Routing

Configure an OpenVPN server

Required configuration items

- Enable the OpenVPN server.

  The OpenVPN server is enabled by default.
- The mode used by the OpenVPN server, one of:
  - **TUN (OpenVPN managed)**—Also known as routing mode. Each OpenVPN client is assigned a different IP subnet from the OpenVPN server and other OpenVPN clients. OpenVPN clients use Network Address Translation (NAT) to route traffic from devices connected on its LAN interfaces to the OpenVPN server.
  - **TAP - OpenVPN managed**—Also know as bridging mode. A more advanced implementation of OpenVPN. The IX15 device creates an OpenVPN interface and uses standard interface configuration (for example, a standard DHCP server configuration).
  - **TAP - Device only**—An alternate form of OpenVPN bridging mode, in which the device, rather than OpenVPN, controls the interface configuration. If this method is is, the OpenVPN server must be included as a device in either an interface or a bridge.
- The firewall zone to be used by the OpenVPN server.
- The IP network and subnet mask of the OpenVPN server.
- The server's Certificate authority (CA) certificate, and public, private and Diffie-Hellman (DH) keys.
- An OpenVPN authentication group and an OpenVPN user.
- Determine the method of certificate management:
  - Certificates managed by the server.
  - Certificates created externally and added to the server.
- If certificates are created and added to the server, determine the level of authentication:
  - Certificate authentication only.
  - Username and password authentication only.
  - Certificate and username and password authentication.

  If username and password authentication is used, you must create an OpenVPN authentication group and user. See Configure an OpenVPN Authentication Group and User for instructions.
- Certificates and keys:
  - The **CA certificate** (usually in a ca.crt file).
  - The **Public key** (for example, server.crt)

- The **Private key** (for example, server.key).
- The **Diffie Hellman key** (usually in dh2048.pem).

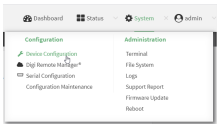
- Active recovery configuration. See Configure SureLink active recovery for OpenVPN for information about OpenVPN active recovery.

Additional configuration items

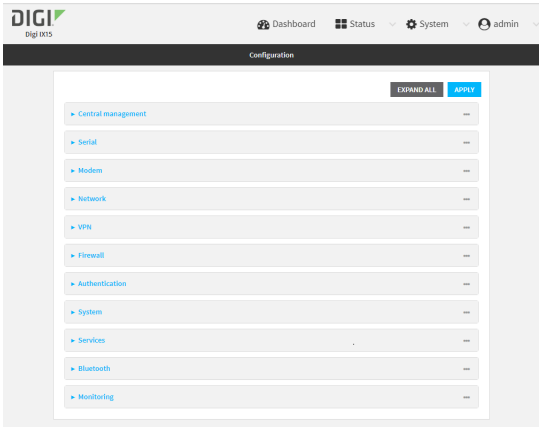
- The route metric for the OpenVPN server.
- The range of IP addresses that the OpenVPN server will provide to clients.
- The TCP/UDP port to use. By default, the IX15 device uses port **1194**.
- Access control list configuration to restrict access to the OpenVPN server through the firewall.
- Additional OpenVPN parameters.

WebUI

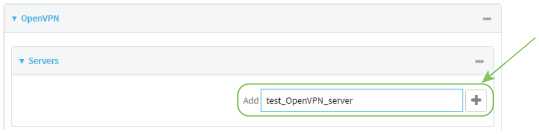
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



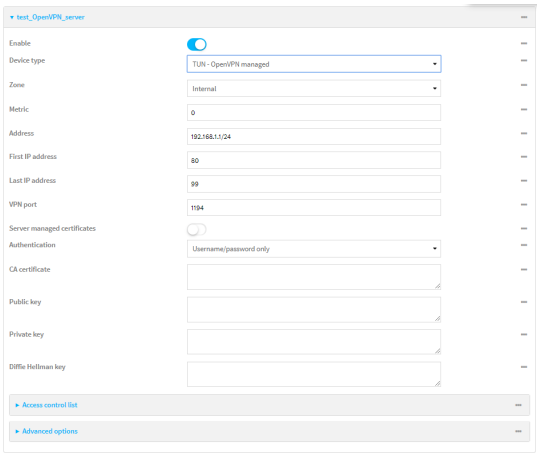
The **Configuration** window is displayed.



3. Click **VPN** > **OpenVPN** > **Servers**.
4. For **Add**, type a name for the OpenVPN server and click ✚.



The new OpenVPN server configuration is displayed.



The OpenVPN server is enabled by default. To disable, click **Enable**.

5. For **Device type**, select the mode used by the OpenVPN server, either:
   - **TUN (OpenVPN managed)**
   - **TAP - OpenVPN managed**
   - **TAP - Device only**

   See OpenVPN for information about OpenVPN server modes.
6. If **TUN (OpenVPN managed)** or **TAP - OpenVPN managed** is selected for **Device type**:
   a. For **Zone**, select the firewall zone for the OpenVPN server. For TUN device types, this should be set to **Internal** to treat clients as LAN devices.
   b. (Optional) Select the **Metric** for the OpenVPN server. If multiple active routes match a destination, the route with the lowest metric will be used. The default setting is **0**.
   c. For **Address**, type the IP address and subnet mask of the OpenVPN server.
   d. (Optional) For **First IP address** and **Last IP address**, set the range of IP addresses that the OpenVPN server will use when providing IP addresses to clients. The default is from **80** to **99**.
7. (Optional) Set the **VPN port** that the OpenVPN server will use. The default is **1194**.
8. For **Server managed certificates**, determine the method of certificate management. If enabled, the server will manage certificates. If not enabled, certificates must be created externally and added to the server.
9. If **Server managed certificates** is not enabled:
   a. Select the **Authentication** type:
      - **Certificate only**: Uses only certificates for client authentication. Each client requires a public and private key.
      - **Username/password only**: Uses a username and password for client authentication. You must create an OpenVPN authentication group and user. See Configure an OpenVPN Authentication Group and User for instructions.
      - **Certificate and username/password**: Uses both certificates and a username and password for client authentication. Each client requires a public and private key, and you must create an OpenVPN authentication group and user. See Configure an OpenVPN Authentication Group and User for instructions.

b. Paste the contents of the **CA certificate** (usually in a ca.crt file), the **Public key** (for example, server.crt), the **Private key** (for example, server.key), and the **Diffie Hellman key** (usually in dh2048.pem) into their respective fields. The contents will be hidden when the configuration is saved.

10. (Optional) Click to expand **Access control list** to restrict access to the OpenVPN server:
    - To limit access to specified IPv4 addresses and networks:
      a. Click **IPv4 Addresses**.
      b. For **Add Address**, click ✚.
      c. For **Address**, enter the IPv4 address or network that can access the device's service-type. Allowed values are:
         - A single IP address or host name.
         - A network designation in CIDR notation, for example, 192.168.1.0/24.
         - **any**: No limit to IPv4 addresses that can access the service-type.
      d. Click ✚ again to list additional IP addresses or networks.
    - To limit access to specified IPv6 addresses and networks:
      a. Click **IPv6 Addresses**.
      b. For **Add Address**, click ✚.
      c. For **Address**, enter the IPv6 address or network that can access the device's service-type. Allowed values are:
         - A single IP address or host name.
         - A network designation in CIDR notation, for example, 2001:db8::/48.
         - **any**: No limit to IPv6 addresses that can access the service-type.
      d. Click ✚ again to list additional IP addresses or networks.
    - To limit access to hosts connected through a specified interface on the IX15 device:
      a. Click **Interfaces**.
      b. For **Add Interface**, click ✚.
      c. For **Interface**, select the appropriate interface from the dropdown.
      d. Click ✚ again to allow access through additional interfaces.
    - To limit access based on firewall zones:
      a. Click **Zones**.
      b. For **Add Zone**, click ✚.
      c. For **Zone**, select the appropriate firewall zone from the dropdown.
         See Firewall configuration for information about firewall zones.
      d. Click ✚ again to allow access through additional firewall zones.
11. (Optional) Click to expand **Advanced Options** to manually set additional OpenVPN parameters.
    a. Click **Enable** to enable the use of additional OpenVPN parameters.
    b. Click **Override** if the additional OpenVPN parameters should override default options.
    c. For **OpenVPN parameters**, type the additional OpenVPN parameters.
12. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type:

```
(config)> add vpn openvpn server name
(config vpn openvpn server name)>
```

   where *name* is the name of the OpenVPN server.

   The OpenVPN server is enabled by default. To disable the server, type:

```
(config vpn openvpn server name)> enable false
(config vpn openvpn server name)>
```

4. Set the mode used by the OpenVPN server:

```
(config vpn openvpn server name)> device_type value
(config vpn openvpn server name)>
```

   where *value* is one of:

   - **TUN (OpenVPN managed)**—Also known as routing mode. Each OpenVPN client is assigned a different IP subnet from the OpenVPN server and other OpenVPN clients. OpenVPN clients use Network Address Translation (NAT) to route traffic from devices connected on its LAN interfaces to the OpenVPN server.
   - **TAP - OpenVPN managed**—Also know as bridging mode. A more advanced implementation of OpenVPN. The IX15 device creates an OpenVPN interface and uses standard interface configuration (for example, a standard DHCP server configuration).
   - **TAP - Device only**—An alternate form of OpenVPN bridging mode, in which the device, rather than OpenVPN, controls the interface configuration. If this method is is, the OpenVPN server must be included as a device in either an interface or a bridge.

   See OpenVPN for information about OpenVPN modes. The default is **tun**.

5. If **tap** or **tun** are set for **device_type**:

   a. Set the IP address and subnet mask of the OpenVPN server.

```
(config vpn openvpn server name)> address ip_address/netmask
(config vpn openvpn server name)>
```

b. Set the firewall zone for the OpenVPN server. For TUN device types, this should be set to **internal** to treat clients as LAN devices.

```
(config vpn openvpn server name)> zone value
(config vpn openvpn server name)>
```

To view a list of available zones:

```
(config vpn openvpn server name)> firewall zone ?

Zone: The zone for the local TUN interface. To treat clients as LAN
devices this would usually be
set to internal.
Format:
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup
Current value:

(config vpn openvpn server name)>
```

c. (Optional) Set the route metric for the OpenVPN server. If multiple active routes match a destination, the route with the lowest metric will be used.

```
(config vpn openvpn server name)> metric value
(config vpn openvpn server name)>
```

where *value* is an interger between **0** and **65535**. The default is **0**.

d. (Optional) Set the range of IP addresses that the OpenVPN server will use when providing IP addresses to clients:

i. Set the first address in the range limit:

```
(config vpn openvpn server name)> server_first_ip value
(config vpn openvpn server name)>
```

where *value* is a number between **1** and **255**. The number entered here will represent the first client IP address. For example, if **address** is set to **192.168.1.1/24** and **server_first_ip** is set to **80**, the first client IP address will be 192.168.1.80.

The default is from **80**.

ii. Set the last address in the range limit:

```
(config vpn openvpn server name)> server_last_ip value
(config vpn openvpn server name)>
```

where *value* is a number between **1** and **255**. The number entered here will represent the last client IP address. For example, if **address** is set to **192.168.1.1/24** and **server_last_ip** is set to **99**, the last client IP address will be 192.168.1.80.

The default is from **80**.

6. (Optional) Set the port that the OpenVPN server will use:

```
(config vpn openvpn server name)> port port
(config vpn openvpn server name)>
```

The default is **1194**.

7. Determine the method of certificate management:
   a. To allow the server to manage certificates:

```
(config vpn openvpn server name)> autogenerate true
(config vpn openvpn server name)>
```

   b. To create certificates externally and add them to the server

```
(config vpn openvpn server name)> autogenerate false
(config vpn openvpn server name)>
```

   The default setting is **false**.

   c. If **autogenerate** is set to false:
      i. Set the authentication type:

```
(config vpn openvpn server name)> authentication value
(config vpn openvpn server name)>
```

      where *value* is one of:

      - **cert**: Uses only certificates for client authentication. Each client requires a public and private key.
      - **passwd**: Uses a username and password for client authentication. You must create an OpenVPN authentication group and user. See Configure an OpenVPN Authentication Group and User for instructions.
      - **cert_passwd**: Uses both certificates and a username and password for client authentication. Each client requires a public and private key, and you must create an OpenVPN authentication group and user. See Configure an OpenVPN Authentication Group and User for instructions.

      ii. Paste the contents of the CA certificate (usually in a ca.crt file) into the value of the **cacert** parameter:

```
(config vpn openvpn server name)> cacert value
(config vpn openvpn server name)>
```

      iii. Paste the contents of the public key (for example, server.crt) into the value of the **server_cert** parameter:

```
(config vpn openvpn server name)> server_cert value
(config vpn openvpn server name)>
```

iv. Paste the contents of the private key (for example, server.key) into the value of the **server_key** parameter:

```
(config vpn openvpn server name)> server_key value
(config vpn openvpn server name)>
```

v. Paste the contents of the Diffie Hellman key (usually in dh2048.pem) into the value of the **diffie** parameter:

```
(config vpn openvpn server name)> diffie value
(config vpn openvpn server name)>
```

8. (Optional) Set the access control list to restrict access to the OpenVPN server:

- To limit access to specified IPv4 addresses and networks:

```
(config vpn openvpn server name)> add acl address end value
(config vpn openvpn server name)>
```

Where *value* can be:

- A single IP address or host name.
- A network designation in CIDR notation, for example, 192.168.1.0/24.
- **any**: No limit to IPv4 addresses that can access the service-type.

Repeat this step to list additional IP addresses or networks.

- To limit access to specified IPv6 addresses and networks:

```
(config vpn openvpn server name)> add acl address6 end value
(config vpn openvpn server name)>
```

Where *value* can be:

- A single IP address or host name.
- A network designation in CIDR notation, for example, 2001:db8::/48.
- **any**: No limit to IPv6 addresses that can access the service-type.

Repeat this step to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX15 device:

```
(config vpn openvpn server name)> add acl interface end value
(config vpn openvpn server name)>
```

Where *value* is an interface defined on your device.

Display a list of available interfaces:

Use **... network interface ?** to display interface information:

```
(config vpn openvpn server name)> ... network interface ?

Interfaces

Additional Configuration
------------------------------------------
```

```
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config vpn openvpn server name)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config vpn openvpn server name)> add acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config vpn openvpn server name)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.

 Additional Configuration
 ------------------------------------------------------------
-----------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

(config vpn openvpn server name)>
```

Repeat this step to list additional firewall zones.

9. (Optional) Set additional OpenVPN parameters.
   a. Enable the use of additional OpenVPN parameters:

```
(config vpn openvpn server name)> advanced_options enable true
(config vpn openvpn server name)>
```

   b. Configure whether the additional OpenVPN parameters should override default options:

```
(config vpn openvpn server name)> advanced_options override true
(config vpn openvpn server name)>
```

c. Set the additional OpenVPN parameters:

```
(config vpn openvpn server name)> extra parameters
(config vpn openvpn server name)>
```

10. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

11. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

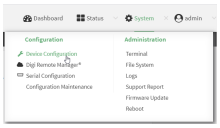
Configure an OpenVPN Authentication Group and User

If username and password authentication is used for the OpenVPN server, you must create an OpenVPN authentication group and user.

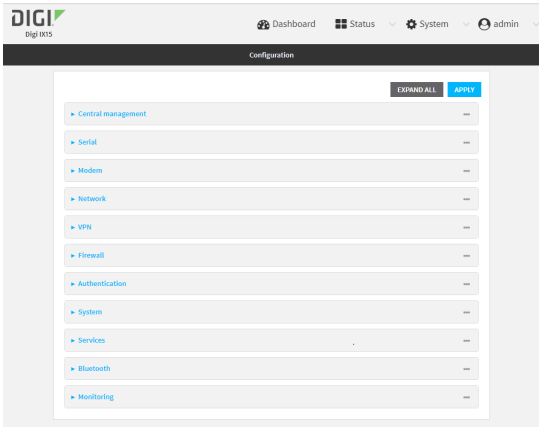
See Configure an OpenVPN server for information about configuring an OpenVPN server to use username and password authentication. See IX15 user authentication for more information about creating authentication groups and users.

WebUI

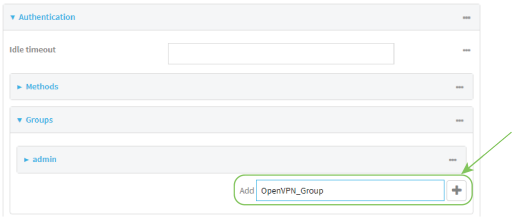
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



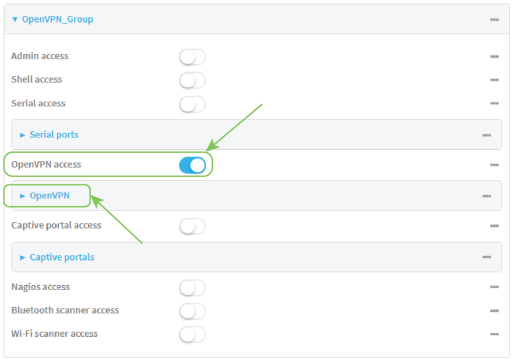
The **Configuration** window is displayed.



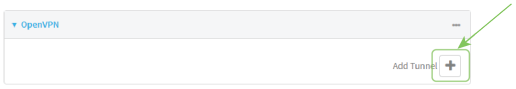
3. Add an OpenVPN authentication group:
   a. Click **Authentication** > **Groups**.
   b. For **Add Group**, type a name for the group (for example, **OpenVPN_Group**) and click ✚.



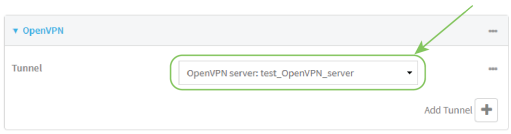
The new authentication group configuration is displayed.



   c. Click **OpenVPN access** to enable OpenVPN access rights for users of this group.
   d. Click to expand the **OpenVPN** node.
   e. Click ✚ to add a tunnel.

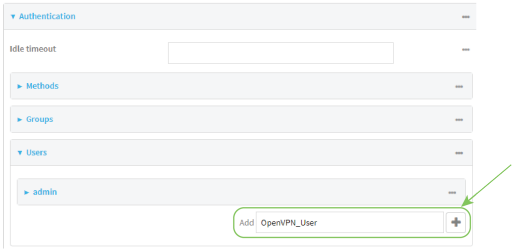


   f. For **Tunnel**, select an OpenVPN tunnel to which users of this group will have access.



   g. Repeat to add additional OpenVPN tunnels.

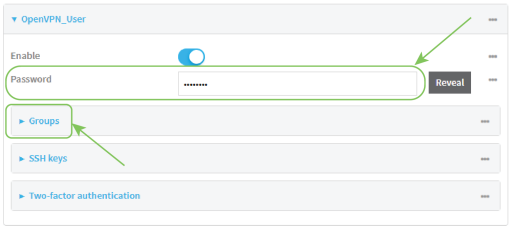
4. Add an OpenVPN authentication user:
   a. Click **Authentication** > **Users**.
   b. For **Add**, type a name for the user (for example, **OpenVPN_User**) and click **+**.



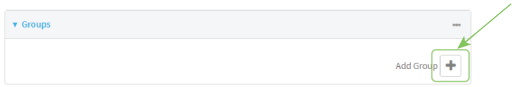
   c. Type a password for the user.

      This password is used for local authentication of the user. You can also configure the user to use RADIUS or TACACS+ authentication by configuring authentication methods. See User authentication methods for information.

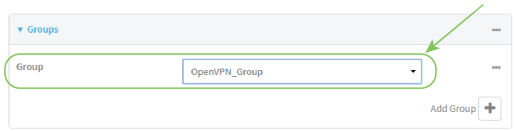
   d. Click to expand the **Groups** node.



   e. Click **+** to add a group to the user.



   f. Select a **Group** with **OpenVPN access** enabled.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Use the **add auth group** command to add a new authentication. For example, to add a group named **OpenVPN_Group**:

```
(config)> add auth group OpenVPN_Group
(config auth group OpenVPN_Group)>
```

4. Enable OpenVPN access rights for users of this group:

```
(config auth group OpenVPN_Group)> acl openvpn enable true
```

5. Add an OpenVPN tunnel to which users of this group will have access:
   a. Determine available tunnels:

```
(config auth group OpenVPN_Group)> .. .. .. vpn openvpn server ?

Servers: A list of openvpn servers

 Additional Configuration
 ------------------------------------------------------------------------
----------
 OpenVPN_server1           OpenVPN server

(config auth group OpenVPN_Group)>
```

   b. Add a tunnel:

```
(config auth group OpenVPN_Group)> add auth group test acl openvpn
tunnels end /vpn/openvpn/server/OpenVPN_server1
(config auth group OpenVPN_Group)>
```

6. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure an OpenVPN client by using an .ovpn file

Required configuration items

- Enable the OpenVPN client.
  The OpenVPN client is enabled by default.
- The firewall zone to be used by the OpenVPN client.

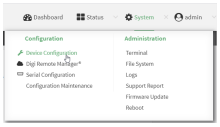
Additional configuration items

- The route metric for the OpenVPN client.
- The login credentials for the OpenVPN client, if configured on the OpenVPN server.

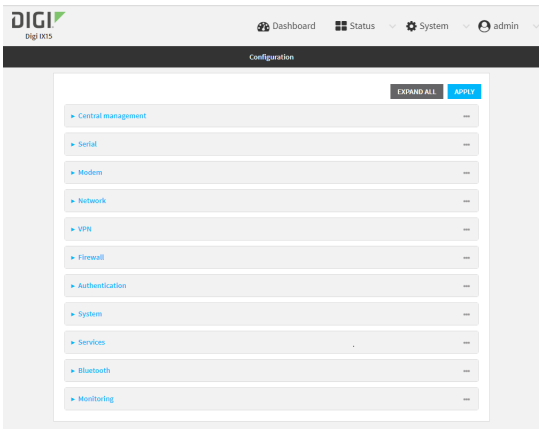
See Configure SureLink active recovery for OpenVPN for information about OpenVPN active recovery.

WebUI

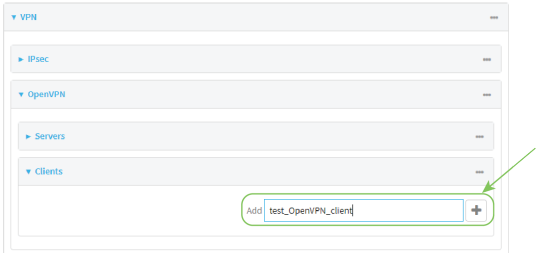
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



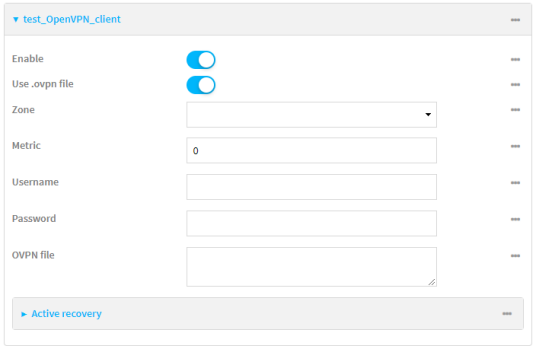
The **Configuration** window is displayed.



3. Click **VPN** > **OpenVPN** > **Clients**.
4. For **Add**, type a name for the OpenVPN client and click ✚.



The new OpenVPN client configuration is displayed.



5. The OpenVPN client is enabled by default. To disable, click **Enable**.
6. The default behavior is to use an OVPN file for client configuration. To disable this behavior and configure the client manually, click **Use .ovpn file** to disable. If **Use .ovpn file** is disabled, see Configure an OpenVPN client without using an .ovpn file for configuration information.
7. For **Zone**, select the firewall zone for the OpenVPN client.
8. (Optional) Select the **Metric** for the OpenVPN client. If multiple active routes match a destination, the route with the lowest metric will be used.
9. (Optional) For **Username** and **Password**, type the login credentials as configured on the OpenVPN server.
10. For **OVPN file**, paste the content of the client.ovpn file.
11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type:

```
(config)> add vpn openvpn client name
(config vpn openvpn client name)>
```

where *name* is the name of the OpenVPN server.

The OpenVPN client is enabled by default. To disable the client, type:

```
(config vpn openvpn client name)> enable false
(config vpn openvpn client name)>
```

4. Set the firewall zone for the OpenVPN client:

```
(config vpn openvpn client name)> zone value
(config vpn openvpn client name)>
```

To view a list of available zones:

```
(config vpn openvpn client name)> zone ?

Zone: The zone for the openvpn client interface.
Format:
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup
Current value:

(config vpn openvpn client name)>
```

5. (Optional) Set the route metric for the OpenVPN server. If multiple active routes match a destination, the route with the lowest metric will be used.

```
(config vpn openvpn client name)> metric value
(config vpn openvpn client name)>
```

where *value* is an interger between **0** and **65535**. The default is **0**.

6. (Optional) Set the login credentials as configured on the OpenVPN server:

```
(config vpn openvpn client name)> username value
(config vpn openvpn client name)> password value
(config vpn openvpn client name)>
```

7. Paste the content of the client.ovpn file into the value of the **config_file** parameter:

```
(config vpn openvpn client name)> config_file value
(config vpn openvpn client name)>
```

8. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure an OpenVPN client without using an .ovpn file

Required configuration items

- Enable the OpenVPN client.

  The OpenVPN client is enabled by default.
- The mode used by the OpenVPN server, either routing (TUN), or bridging (TAP).
- The firewall zone to be used by the OpenVPN client.
- The IP address of the OpenVPN server.
- Certificates and keys:
  - The **CA certificate** (usually in a ca.crt file).
  - The **Public key** (for example, client.crt)
  - The **Private key** (for example, client.key).

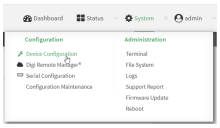
Additional configuration items

- The route metric for the OpenVPN client.
- The login credentials for the OpenVPN client, if configured on the OpenVPN server.
- Additional OpenVPN parameters.

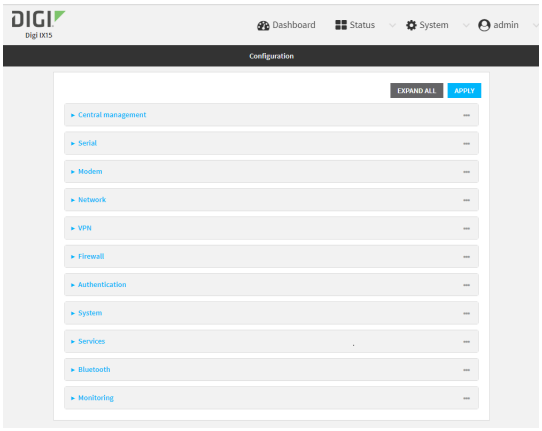
See Configure SureLink active recovery for OpenVPN for information about OpenVPN active recovery.

WebUI

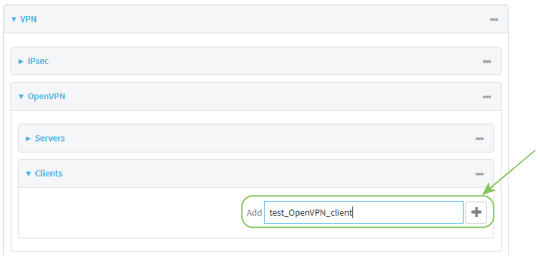
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

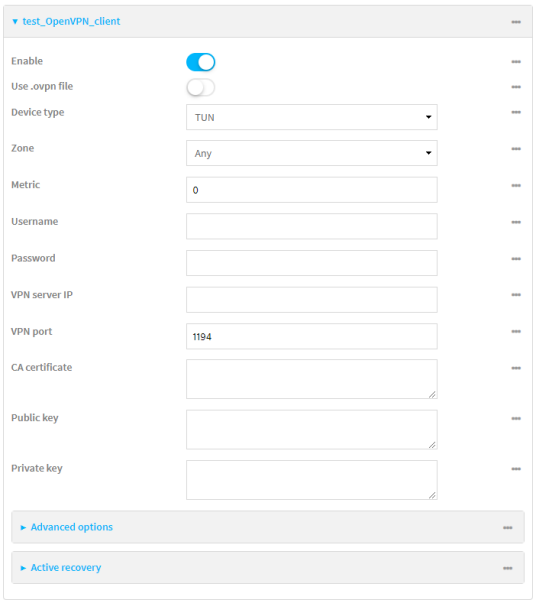
   The **Configuration** window is displayed.



3. Click **VPN** > **OpenVPN** > **Clients**.
4. For **Add**, type a name for the OpenVPN client and click **+**.



The new OpenVPN client configuration is displayed.



5. The OpenVPN client is enabled by default. To disable, click **Enable**.
6. The default behavior is to use an OVPN file for client configuration. To disable this behavior and configure the client manually, click **Use .ovpn file** to disable.

7. For **Device type**, select the mode used by the OpenVPN server, either **TUN** or **TAP**.
8. For **Zone**, select the firewall zone for the OpenVPN client.
9. (Optional) Select the **Metric** for the OpenVPN client. If multiple active routes match a destination, the route with the lowest metric will be used.
10. (Optional) For **Username** and **Password**, type the login credentials as configured on the OpenVPN server.
11. For **VPN server IP**, type the IP address of the OpenVPN server.
12. (Optional) Set the **VPN port** used by the OpenVPN server. The default is **1194**.
13. Paste the contents of the **CA certificate** (usually in a ca.crt file), the **Public key** (for example, client.crt), and the **Private key** (for example, client.key) into their respective fields. The contents will be hidden when the configuration is saved.
14. (Optional) Click to expand **Advanced Options** to manually set additional OpenVPN parameters.
    a. Click **Enable** to enable the use of additional OpenVPN parameters.
    b. Click **Override** if the additional OpenVPN parameters should override default options.
    c. For **OpenVPN parameters**, type the additional OpenVPN parameters. For example, to override the configuration by using a configuration file, enter **--config filename**, for example, **--config /etc/config/openvpn_config**.
15. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type:

```
(config)> add vpn openvpn client name
(config vpn openvpn client name)>
```

   where *name* is the name of the OpenVPN server.

   The OpenVPN client is enabled by default. To disable the client, type:

```
(config vpn openvpn client name)> enable false
(config vpn openvpn client name)>
```

4. The default behavior is to use an OVPN file for client configuration. To disable this behavior and configure the client manually:

```
(config vpn openvpn client name)> use_file false
(config vpn openvpn client name)>
```

5. Set the mode used by the OpenVPN server:

```
(config vpn openvpn client name)> device_type value
(config vpn openvpn client name)>
```

where *value* is either **tun** or **tap**. The default is **tun**.

6. Set the firewall zone for the OpenVPN client:

```
(config vpn openvpn client name)> zone value
(config vpn openvpn client name)>
```

To view a list of available zones:

```
(config vpn openvpn client name)> zone ?

Zone: The zone for the openvpn client interface.
Format:
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup
Current value:

(config vpn openvpn client name)>
```

7. (Optional) Set the route metric for the OpenVPN server. If multiple active routes match a destination, the route with the lowest metric will be used.

```
(config vpn openvpn client name)> metric value
(config vpn openvpn client name)>
```

where *value* is an interger between **0** and **65535**. The default is **0**.

8. (Optional) Set the login credentials as configured on the OpenVPN server:

```
(config vpn openvpn client name)> username value
(config vpn openvpn client name)> password value
(config vpn openvpn client name)>
```

9. Set the IP address of the OpenVPN server:

```
(config vpn openvpn client name)> server ip_address
(config vpn openvpn client name)>
```

10. (Optional) Set the port used by the OpenVPN server:

```
(config vpn openvpn client name)> port port
(config vpn openvpn client name)>
```

    The default is **1194**.

11. Paste the contents of the CA certificate (usually in a ca.crt file) into the value of the **cacert** parameter:

```
(config vpn openvpn client name)> cacert value
(config vpn openvpn client name)>
```

12. Paste the contents of the public key (for example, client.crt) into the value of the **public_cert** parameter:

```
(config vpn openvpn client name)> public_cert value
(config vpn openvpn client name)>
```

13. Paste the contents of the private key (for example, client.key) into the value of the **private_key** parameter:

```
(config vpn openvpn client name)> private_key value
(config vpn openvpn client name)>
```

14. (Optional) Set additional OpenVPN parameters.
    a. Enable the use of additional OpenVPN parameters:

```
(config vpn openvpn client name)> advanced_options enable true
(config vpn openvpn client name)>
```

    b. Configure whether the additional OpenVPN parameters should override default options:

```
(config vpn openvpn client name)> advanced_options override true
(config vpn openvpn client name)>
```

    c. Set the additional OpenVPN parameters:

```
(config vpn openvpn client name)> advanced_options extra parameters
(config vpn openvpn client name)>
```

15. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

16. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure SureLink active recovery for OpenVPN

You can configure the IX15 device to regularly probe OpenVPN client connections to determine if the connection has failed and take remedial action.

Required configuration items

- A valid OpenVPN client configuration. See Configure an OpenVPN client by using an .ovpn file or Configure an OpenVPN client without using an .ovpn file for configuration instructions.
- Enable OpenVPN active recovery.
- The behavior of the IX15 device upon OpenVPN failure: either
  - Restart the OpenVPN interface
  - Reboot the device.

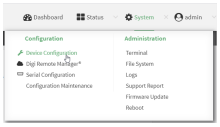
Additional configuration items

- The interval between connectivity tests.
- Whether the interface should be considered to have failed if one of the test targets fails, or all of the test targets fail.
- The number of probe attempts before the OpenVPN connection is considered to have failed.
- The amount of time that the device should wait for a response to a probe attempt before considering it to have failed.

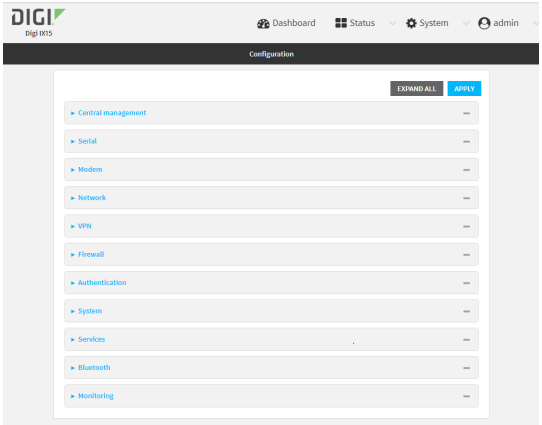
To configure the IX15 device to regularly probe the OpenVPN connection:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



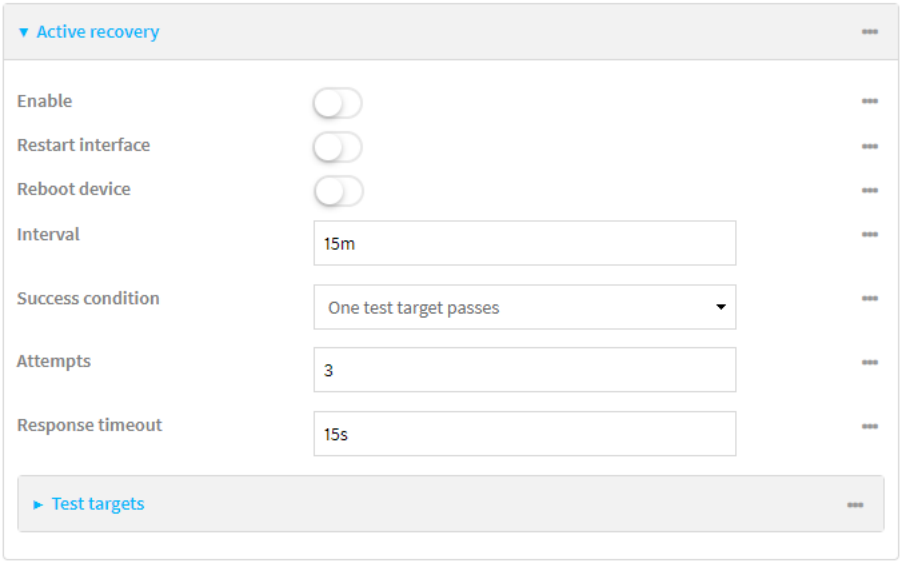
The **Configuration** window is displayed.



3. Click **VPN** > **OpenVPN** > **Clients**.
4. Create a new OpenVPN client or select an existing one:

- To create a new OpenVPN client, see Configure an OpenVPN client by using an .ovpn file or Configure an OpenVPN client without using an .ovpn file.
- To edit an existing OpenVPN client, click to expand the appropriate client.

5. After creating or selecting the OpenVPN client, click **Active recovery**.



6. **Enable** active recovery.
7. For **Restart interface**, enable to configure the device to restart the interface when its connection is considered to have failed. This is useful for interfaces that may regain connectivity after restarting, such as a cellular modem.
8. For **Reboot device**, enable to instruct the device to reboot when the WAN connection is considered to have failed.
9. Change the **Interval** between connectivity tests.

   Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

   For example, to set **Interval** to ten minutes, enter **10m** or **600s**.

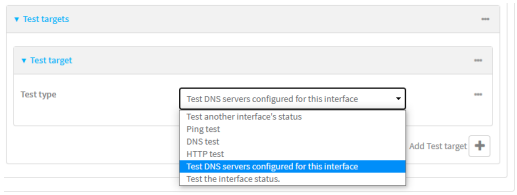
   The default is 15 minutes.
10. For **Success condition**, determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets.
11. For **Attempts**, type the number of probe attempts before the WAN is considered to have failed.
12. For **Response timeout**, type the amount of time that the device should wait for a response to a probe attempt before considering it to have failed.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Response timeout** to ten minutes, enter **10m** or **600s**.

    The default is 15 seconds.

13. Add a test target:
    a. Click to expand **Test targets**.
    b. For **Add Test target**, click **+**.
    c. Select the **Test type**:



- **Test another interface's status**: Allows you to test another interface's status, to create a failover or coupled relationship between interfaces. If **Test another interface's status** is selected:
  - For **Test Interface**, select the alternate interface to be tested.
  - For **IP version**, select the alternate interface's IP version. This allows you to determine the alternate interface's status for a particular IP version.
  - For **Expected status**, select whether the expected status of the alternate interface is **Up** or **Down**. For example, if **Expected status** is set to **Down**, but the alternate interface is determined to be up, then this test will fail.
- **Ping test**: Tests connectivity by sending an ICMP echo request to the hostname or IP address specified in **Ping host**. You can also optionally change the number of bytes in the **Ping payload size**.
- **DNS test**: Tests connectivity by sending a DNS query to the specified **DNS server**.
- **HTTP test**: Tests connectivity by sending an HTTP or HTTPS GET request to the URL specified in **Web servers**. The URL should take the format of **http[s]://*hostname*|[*path*]**.
- **Test DNS servers configured for this interface**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
- **Test the interface status**: The interface is considered to be down based on:
  - **Down time**: The amount of time that the interface can be down before this test is considered to have failed.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Down time** to ten minutes, enter **10m** or **600s**.

    The default is 60 seconds.
  - **Initial connection time**: The amount of time to wait for an initial connection to the interface before this test is considered to have failed.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Initial connection time** to ten minutes, enter **10m** or **600s**.

    The default is 60 seconds.

14. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

   ```
   > config
   (config)>
   ```
3. Create a new OpenVPN client, or edit an existing one:
   - To create a new OpenVPN client, see Configure an OpenVPN client by using an .ovpn file or Configure an OpenVPN client without using an .ovpn file.
   - To edit an existing OpenVPN client, change to the OpenVPN client's node in the configuration schema. For example, for an OpenVPN client named **openvpn_client1**, change to the **openvpn_client1** node in the configuration schema:

     ```
     (config)> vpn openvpn client openvpn_client1
     (config vpn openvpn client openvpn_client1)>
     ```
4. Enable active recovery:

   ```
   (config vpn openvpn client openvpn_client1)> surelink enable true
   (config vpn openvpn client openvpn_client1)>
   ```
5. To configure the device to restart the interface when its connection is considered to have failed:

   ```
   (config vpn openvpn client openvpn_client1)> surelink restart true
   (config vpn openvpn client openvpn_client1)>
   ```

   This is useful for interfaces that may regain connectivity after restarting, such as a cellular modem.
6. To configure the device to reboot when the interface is considered to have failed:

   ```
   (config vpn openvpn client openvpn_client1)> surelink reboot enable
   (config vpn openvpn client openvpn_client1)>
   ```
7. Set the **Interval** between connectivity tests:

   ```
   (config vpn openvpn client openvpn_client1)> surelink interval value
   (config vpn openvpn client openvpn_client1)>
   ```

   where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

   For example, to set **interval** to ten minutes, enter either **10m** or **600s**:

```
(config vpn openvpn client openvpn_client1)> surelink interval 600s
(config vpn openvpn client openvpn_client1)>
```

The default is 15 minutes.

8. Determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets:

```
(config vpn openvpn client openvpn_client1)> surelink success_condition
value
(config vpn openvpn client openvpn_client1)>
```

Where *value* is either **one** or **all**.

9. Set the number of probe attempts before the WAN is considered to have failed:

```
(config vpn openvpn client openvpn_client1)> surelink  attempts num
(config vpn openvpn client openvpn_client1)>
```

The default is **3**.

10. Set the amount of time that the device should wait for a response to a probe attempt before considering it to have failed:

```
(config vpn openvpn client openvpn_client1)> surelink timeout value
(config vpn openvpn client openvpn_client1)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **timeout** to ten minutes, enter either **10m** or **600s**:

```
(config vpn openvpn client openvpn_client1)> surelink timeout 600s
(config vpn openvpn client openvpn_client1)>
```

The default is 15 seconds.

11. Configure test targets:
    a. Add a test target:

```
(config vpn openvpn client openvpn_client1)> add surelink target end
(config vpn openvpn client openvpn_client1 surelink target 0)>
```

    b. Set the test type:

```
(config vpn openvpn client openvpn_client1 surelink target 0)> test
value
(config vpn openvpn client openvpn_client1 surelink target 0)>
```

    where *value* is one of:

    - **ping**: Tests connectivity by sending an ICMP echo request to a specified hostname or IP address.

- Specify the hostname or IP address:

```
(config vpn openvpn client openvpn_client1 surelink target
0)> ping_host host
(config vpn openvpn client openvpn_client1 surelink target
0)>
```

- (Optional) Set the size, in bytes, of the ping packet:

```
(config vpn openvpn client openvpn_client1 surelink target
0)> ping_size [num]
(config vpn openvpn client openvpn_client1 surelink target
0)>
```

- **dns**: Tests connectivity by sending a DNS query to the specified DNS server.
  - Specify the DNS server. Allowed value is the IP address of the DNS server.

```
(config vpn openvpn client openvpn_client1 surelink target
0)> dns_server ip_address
(config vpn openvpn client openvpn_client1 surelink target
0)>
```

- **dns_configured**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
- **http**: Tests connectivity by sending an HTTP or HTTPS GET request to the specified URL.
  - Specify the url:

```
(config vpn openvpn client openvpn_client1 surelink target
0)> http_url value
(config vpn openvpn client openvpn_client1 surelink target
0)>
```

  where *value* uses the format **http[s]://*hostname*/[*path*]**
- **interface_up**: The interface is considered to be down based on the interfaces down time, and the amount of time an initial connection to the interface takes before this test is considered to have failed.
  - (Optional) Set the amount of time that the interface can be down before this test is considered to have failed:

```
(config vpn openvpn client openvpn_client1 surelink target
0)> interface_down_time value
(config vpn openvpn client openvpn_client1 surelink target
0)>
```

  where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w**|**d**|**h**|**m**|**s**}.

  For example, to set **timeout** to ten minutes, enter either **10m** or **600s**:

```
(config vpn openvpn client openvpn_client1 surelink target
0)> interface_down_time 600s
(config vpn openvpn client openvpn_client1 surelink target
0)>
```

The default is 60 seconds.

- (Optional) Set the amount of time to wait for an initial connection to the interface before this test is considered to have failed:

```
(config vpn openvpn client openvpn_client1 surelink target
0)> interface_timeout value
(config vpn openvpn client openvpn_client1 surelink target
0)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **timeout** to ten minutes, enter either **10m** or **600s**:

```
(config vpn openvpn client openvpn_client1 surelink target
0)> interface_timeout 600s
(config vpn openvpn client openvpn_client1 surelink target
0)>
```

The default is 60 seconds.

- **other**: Allows you to test another interface's status, to create a failover or coupled relationship between interfaces:

```
(config vpn openvpn client openvpn_client1 surelink target
0)> other value
(config vpn openvpn client openvpn_client1 surelink target
0)>
```

If **other** is set:

  - Set the alternate interface to be tested:
    i. Use the **?** to determine available interfaces:
    ii. Set the interface. For example:

```
(config vpn openvpn client openvpn_client1 surelink
target 0)> other_interface /network/interface/eth1
(config vpn openvpn client openvpn_client1 surelink
target 0)>
```

  - Set the alternate interface's IP version. This allows you to determine the alternate interface's status for a particular IP version.

```
(config vpn openvpn client openvpn_client1 surelink target
0)> other_ip_version value
```

```
(config vpn openvpn client openvpn_client1 surelink target
0)>
```

where *value* is one of: **any**, **both**, **ipv4**, or **ipv6**.

- Set the expected status of the alternate interface:

```
(config vpn openvpn client openvpn_client1 surelink target
0)> other_status value
(config vpn openvpn client openvpn_client1 surelink target
0)>
```

where *value* is either **up** or **down**. For example, if **other_status** is set to **down**, but the alternate interface is determined to be up, then this test will fail.

12. Save the configuration and apply the change:

```
(config vpn openvpn client openvpn_client1 connection_monitor target 0)>
save
Configuration saved.
>
```

13. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

See Show Surelink status and statistics for information about showing Surelink status for OpenVPN clients.

Show OpenVPN server status and statistics

You can view status and statistics for OpenVPN servers from either the web interface or the command line:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, select **Status** > **OpenVPN** > **Servers**.

   The **OpenVPN Servers** page appears.
3. To view configuration details about an OpenVPN server, click the (configuration) icon in the upper right of the OpenVPN server's status pane.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. To display details about all configured OpenVPN servers, type the following at the prompt:

```
> show openvpn server all

 Server           Enable  Type  Zone      IP Address       Port
 ---------------  ------  ----  --------  ---------------  ----
 OpenVPN_server1  true    tun   internal  192.168.30.1/24  1194
 OpenVPN_server2  false   tun   internal  192.168.40.1/24  1194

>
```

3. To display details about a specific server:

```
> show openvpn server name OpenVPN_server1

 Server                   : OpenVPN_server1
 Enable                   : true
 Type                     : tun
 Zone                     : internal
 IP Address               : 192.168.30.1/24
 Port                     : 1194
 Use File                 : true
 Metric                   : 0
 Protocol                 : udp
 First IP                 : 80
 Last IP                  : 99

>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show OpenVPN client status and statistics

You can view status and statistics for OpenVPN clients from either web interface or the command line:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, select **Status** > **OpenVPN** > **Clients**.

   The **OpenVPN Clients** page appears.
3. To view configuration details about an OpenVPN client, click the (configuration) icon in the upper right of the OpenVPN client's status pane.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. To display details about all configured OpenVPN clients, type the following at the prompt:

```
> show openvpn client all

 Client           Enable  Status     Username  Use File  Zone
 ---------------  ------  -------    --------  --------  --------
 OpenVPN_Client1  true    connected            true      internal
 OpenVPN_Client2  true    pending              true      internal

>
```

3. To display details about a specific client:

```
> show openvpn client name OpenVPN_client1

 Client                   : OpenVPN_client1
 Enable                   : true
 Status                   : up
 Username                 : user1
 IP address               : 123.122.121.120
 Remote                   : 120.121.122.123
 MTU                      : 1492
 Zone                     : internal
 IP Address               : 192.168.30.1/24
 Port                     : 1194
 Use File                 : true
 Metric                   : 0
 Protocol                 : udp
 Port                     : 1194
 Type                     : tun

>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Generic Routing Encapsulation (GRE)

Generic Routing Encapsulation (GRE) is an IP packet encapsulation protocol that allow for networks and routes to be advertized from one network device to another. You can use GRE to encapsulate a wide variety of network layer protocols inside virtual point-to-point links over an IP network.

Configuring a GRE tunnel

Configuring a GRE tunnel involves the following items:

Required configuration items

- A GRE loopback endpoint interface.
- GRE tunnel configuration:
  - Enable the GRE tunnel.

    The GRE tunnels are enabled by default.
  - The local endpoint interface.
  - The IP address of the remote device/peer.

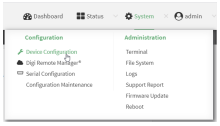
Additional configuration items

- A GRE key.
- Enable the device to respond to keepalive packets.

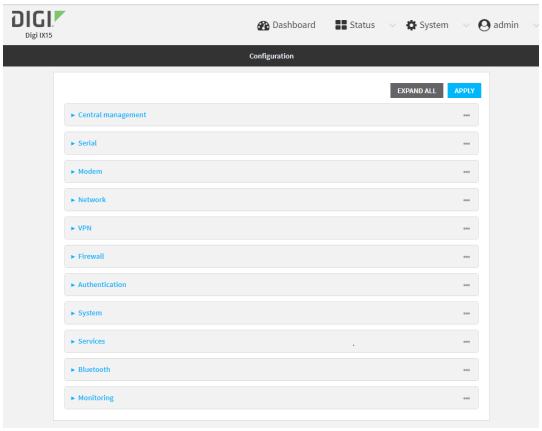
Task One: Create a GRE loopback endpoint interface

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Interfaces**.
4. For **Add Interface**, type a name for the GRE loopback endpoint interface and click **+**.
5. **Enable** the interface.

   New interfaces are enabled by default. To disable, or to enable if it has been disabled, click **Enable**.
6. For **Interface type**, select **Ethernet**.
7. For **Zone**, select **Internal**.
8. For **Device**, select **Ethernet: Loopback**.
9. Click to expand **IPv4**.
10. For **Address**, enter the IP address and subnet mask of the local GRE endpoint, for example **10.10.1.1/24**.
11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add the GRE endpoint interface. For example, to add an interface named **gre_endpoint**:

```
(config)> add network interface gre_interface
(config network interface gre_interface)>
```

4. Set the interface zone to **internal**:

```
(config network interface gre_interface)> zone internal
(config network interface gre_interface)>
```

5. Set the interface device to **loopback**:

```
(config network interface gre_interface)> device /network/device/loopback
(config network interface gre_interface)>
```

6. Set the IP address and subnet mask of the local GRE endpoint. For example, to set the local GRE endpoint's IP address and subnet mask to **10.10.1.1/24:**

```
(config network interface gre_interface)> ipv4 address 10.10.1.1/24
(config network interface gre_interface)>
```

7. Save the configuration and apply the change:

```
(config network interface gre_interface)> save
Configuration saved.
>
```

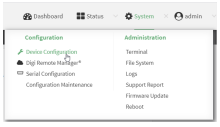
8. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

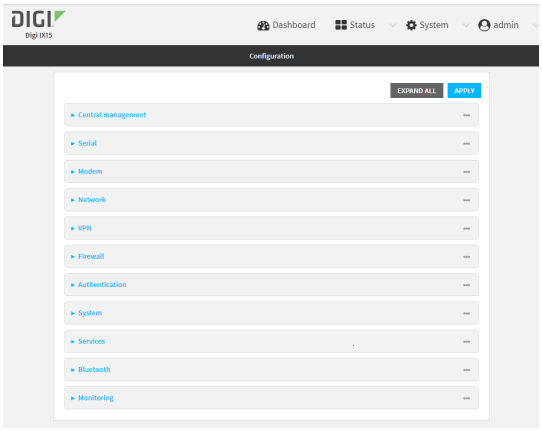
Task Two: Configure the GRE tunnel

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **VPN** > **IP Tunnels**.
4. For **Add IP tunnel**, type a name for the GRE tunnel and click ✚.
5. **Enable** the tunnel.

   New tunnels are enabled by default. To disable, or to enable if it has been disabled, click **Enable**.
6. For **Local endpoint**, select the GRE endpoint interface created in Task One.
7. For **Remote endpoint**, type the IP address of the GRE endpoint on the remote peer.
8. (Optional) For **Key**, enter a key that will be inserted in GRE packets created by this tunnel. It must match the key set by the remote endpoint. Allowed value is an interger between 0 and 4294967295, or an IP address.
9. (Optional) **Enable keepalive reply** to enable the device to reply to Cisco GRE keepalive packets.
10. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add the GRE endpoint tunnel. For example, to add a tunnel named **gre_example**:

```
(config)> add vpn iptunnel gre_example
(config vpn iptunnel gre_example)>
```

GRE tunnels are enabled by default. To disable:

```
(config vpn iptunnel gre_example)> enable false
(config vpn iptunnel gre_example)>
```

4. Set the local endpoint to the GRE endpoint interface created in Task One, for example:

```
(config vpn iptunnel gre_example)> local /network/interface/gre_endpoint
(config vpn iptunnel gre_example)>
```

5. Set the IP address of the GRE endpoint on the remote peer:

```
(config vpn iptunnel gre_example)> remote ip_address
(config vpn iptunnel gre_example)>
```

6. (Optional) Set a key that will be inserted in GRE packets created by this tunnel.

   The key must match the key set by the remote endpoint.

```
(config vpn iptunnel gre_example)> key value
(config vpn iptunnel gre_example)>
```

   where value is an interger between 0 and 4294967295, or an IP address.

7. (Optional) Enable the device to reply to Cisco GRE keepalive packets:

```
(config vpn iptunnel gre_example)> keepalive true
(config vpn iptunnel gre_example)>
```

8. Save the configuration and apply the change:

```
(config vpn iptunnel gre_example)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show GRE tunnels

To view information about currently configured GRE tunnels:

WebUI

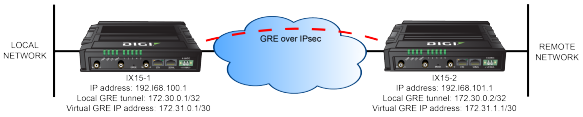
1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **Status** > **IP tunnels**.

   The **IP Tunnels**page appears.
3. To view configuration details about a GRE tunnel, click the (configuration) icon in the upper right of the tunnel's status pane.

Example: GRE tunnel over an IPSec tunnel

The IX15 device can be configured as an advertised set of routes through an IPSec tunnel. This allows you to leverage the dynamic route advertisement of GRE tunnels through a secured IPSec tunnel.

The example configuration provides instructions for configuring the IX15 device with a GRE tunnel through IPsec.



IX15-1 configuration tasks

1. Create an IPsec tunnel named **ipsec_gre1** with:
   - A pre-shared key.
   - **Remote endpoint** set to the public IP address of the IX15-2 device.
   - A policy with:
     - **Local network** set to the IP address and subnet of the local GRE tunnel, **172.30.0.1/32**.
     - **Remote network** set to the IP address and subnet of the remote GRE tunnel, **172.30.0.2/32**.
2. Create an IPsec endpoint interface named **ipsec_endpoint1**:
   a. **Zone** set to **Internal**.
   b. **Device** set to **Ethernet: Loopback**.
   c. IPv4 Address set to the IP address of the local GRE tunnel, **172.30.0.1/32**.
3. Create a GRE tunnel named **gre_tunnel1**:
   a. **Local endpoint** set to the IPsec endpoint interface, **Interface: ipsec_endpoint1**.
   b. Remote endpoint set to the IP address of the GRE tunnel on IX15-2, **172.30.0.2**.
4. Create an interface named **gre_interface1** and add it to the GRE tunnel:
   a. **Zone** set to **Internal**.
   b. **Device** set to **IP tunnel: gre_tunnel1**.
   c. IPv4 Address set to a virtual IP address on the GRE tunnel, **172.31.0.1/30**.

IX15-2 configuration tasks

1. Create an IPsec tunnel named **ipsec_gre2** with:
   - The same pre-shared key as the **ipsec_gre1** tunnel on IX15-1.
   - **Remote endpoint** set to the public IP address of IX15-1.
   - A policy with:
     - **Local network** set to the IP address and subnet of the local GRE tunnel, **172.30.0.2/32**.
     - **Remote network** set to the IP address of the remote GRE tunnel, **172.30.0.1/32**.

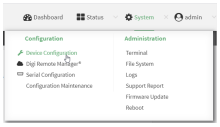
2. Create an IPsec endpoint interface named **ipsec_endpoint2**:
   a. **Zone** set to **Internal**.
   b. **Device** set to **Ethernet: Loopback**.
   c. IPv4 Address set to the IP address of the local GRE tunnel, **172.30.0.2/32**.
3. Create a GRE tunnel named **gre_tunnel2**:
   a. **Local endpoint** set to the IPsec endpoint interface, **Interface: ipsec_endpoint2**.
   b. Remote endpoint set to the IP address of the GRE tunnel on IX15-1, **172.30.0.1**.
4. Create an interface named **gre_interface2** and add it to the GRE tunnel:
   a. **Zone** set to **Internal**.
   b. **Device** set to **IP tunnel: gre_tunnel2**.
   c. IPv4 Address set to a virtual IP address on the GRE tunnel, **172.31.1.1/30**.

Configuration procedures

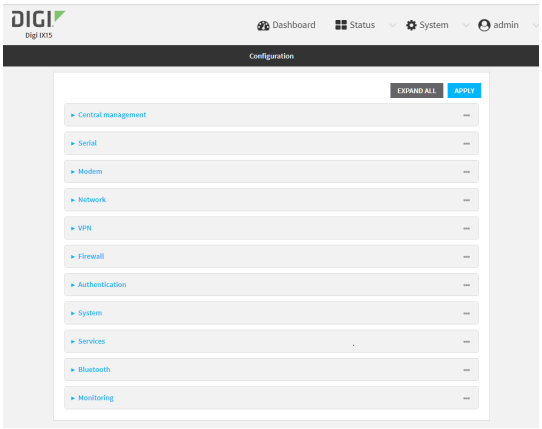
Configure the IX15-1 device
Task one: Create an IPsec tunnel

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

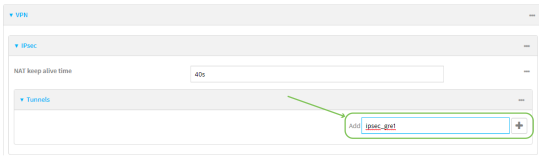


The **Configuration** window is displayed.

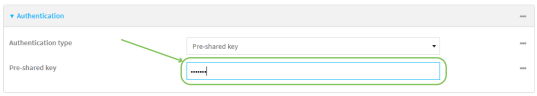


3. Click **VPN** > **IPsec** > **Tunnels**.

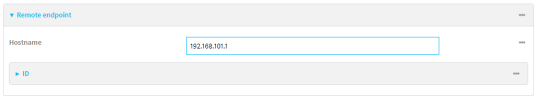
4. For **Add IPsec Tunnel**, type **ipsec_gre1** and click ✚.



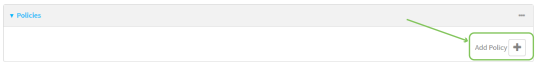
5. Click to expand **Authentication**.
6. For **Pre-shared key**, type **testkey**.



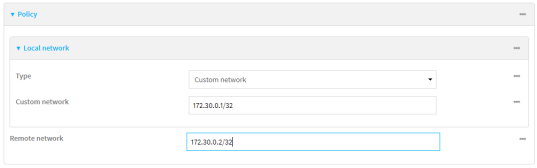
7. Click to expand **Remote endpoint**.
8. For **Hostname**, type public IP address of the IX15-2 device.



9. Click to expand **Policies**.
10. For **Add Policy**, click ✚ to add a new policy.



11. Click to expand **Local network**.
12. For **Type**, select **Custom network**.
13. For **Address**, type the IP address and subnet of the local GRE tunnel, **172.30.0.1/32**.
14. For **Remote network**, type the IP address and subnet of the remote GRE tunnel, **172.30.0.2/32**.



15. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add an IPsec tunnel named **ipsec_gre1**:

```
(config)> add vpn ipsec tunnel ipsec_gre1
(config vpn ipsec tunnel ipsec_gre1)>
```

4. Set the pre-shared key to **testkey**:

```
(config vpn ipsec tunnel ipsec_gre1)> auth secret testkey
(config vpn ipsec tunnel ipsec_gre1)>
```

5. Set the remote endpoint to public IP address of the IX15-2 device:

```
(config vpn ipsec tunnel ipsec_gre1)> remote hostname 192.168.101.1
(config vpn ipsec tunnel ipsec_gre1)>
```

6. Add a policy:

```
(config vpn ipsec tunnel ipsec_gre1)> add policy end
(config vpn ipsec tunnel ipsec_gre1 policy 0)>
```

7. Set the local network policy type to **custom**:

```
(config vpn ipsec tunnel ipsec_gre1 policy 0)> local type custom
(config vpn ipsec tunnel ipsec_gre1 policy 0)>
```

8. Set the local network address to the IP address and subnet of the local GRE tunnel, **172.30.0.1/32**:

```
(config vpn ipsec tunnel ipsec_gre1 policy 0)> local custom 172.30.0.1/32
(config vpn ipsec tunnel ipsec_gre1 policy 0)>
```

9. Set the remote network address to the IP address and subnet of the remote GRE tunnel, **172.30.0.2/32**:

```
(config vpn ipsec tunnel ipsec_gre1 policy 0)> remote network
172.30.0.2/32
(config vpn ipsec tunnel ipsec_gre1 policy 0)>
```

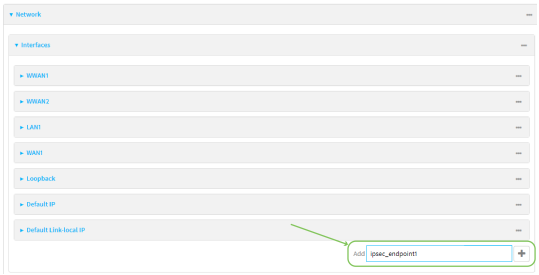
10. Save the configuration and apply the change:

```
(config ipsec tunnel ipsec_gre1 policy 0)> save
Configuration saved.
>
```

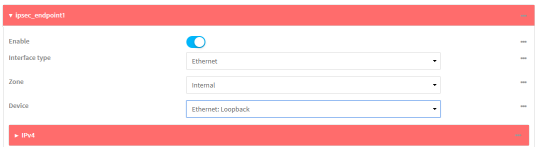
Task two: Create an IPsec endpoint interface

WebUI

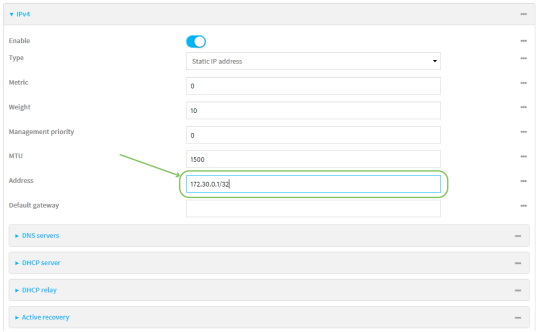
1. Click **Network** > **Interface**.
2. For **Add Interface**, type **ipsec_endpoint1** and click **+**.



3. For **Zone**, select **Internal**.
4. For **Device**, select **Ethernet: loopback**.



5. Click to expand **IPv4**.
6. For **Address**, type the IP address of the local GRE tunnel, **172.30.0.1/32**.



7. Click **Apply** to save the configuration and apply the change.



Command line

1. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

2. Add an interface named **ipsec_endpoint1**:

```
(config)> add network interface ipsec_endpoint1
(config network interface ipsec_endpoint1)>
```

3. Set the zone to **internal**:

```
(config network interface ipsec_endpoint1)> zone internal
(config network interface ipsec_endpoint1)>
```

4. Set the device to **/network/device/loopback**:

```
(config network interface ipsec_endpoint1)> device
/network/device/loopback
(config network interface ipsec_endpoint1)>
```

5. Set the IPv4 address to the IP address of the local GRE tunnel, **172.30.0.1/32**:

```
(config network interface ipsec_endpoint1)> ipv4 address 172.30.0.1/32
(config network interface ipsec_endpoint1)>
```

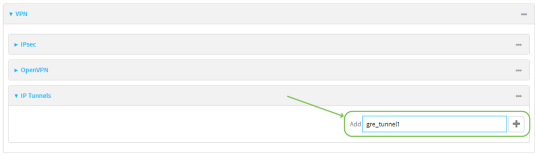
6. Save the configuration and apply the change:

```
(config vpn ipsec tunnel ipsec_endpoint1 policy 0)> save
Configuration saved.
>
```

Task three: Create a GRE tunnel

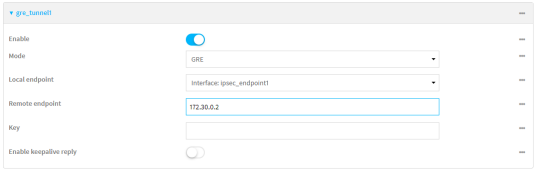
WebUI

1. Click **VPN** > **IP Tunnels**.
2. For **Add IP Tunnel**, type **gre_tunnel1** and click **+**.

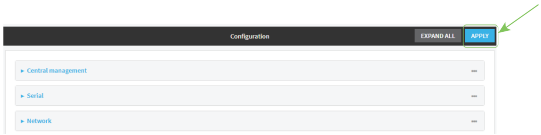


3. For **Local endpoint**, select the IPsec endpoint interface created in Task two (**Interface: ipsec_endpoint1**).

4. For **Remote endpoint**, type the IP address of the GRE tunnel on IX15-2, **172.30.0.2**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

2. Add a GRE tunnel named **gre_tunnel1**:

```
(config)> add vpn iptunnel gre_tunnel1
(config vpn iptunnel gre_tunnel1)>
```

3. Set the local endpoint to the IPsec endpoint interface created in Task two (**/network/interface/ipsec_endpoint1**):

```
(config vpn iptunnel gre_tunnel1)> local /network/interface/ipsec_
endpoint1
(config vpn iptunnel gre_tunnel1)>
```

4. Set the remote endpoint to the IP address of the GRE tunnel on IX15-2, **172.30.0.2**:

```
(config vpn iptunnel gre_tunnel1)> remote 172.30.0.2
(config vpn iptunnel gre_tunnel1)>
```

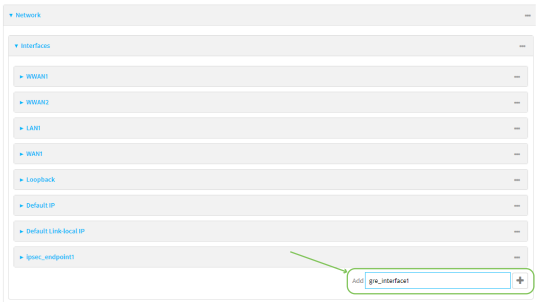
5. Save the configuration and apply the change:

```
(config vpn iptunnel gre_tunnel1)> save
Configuration saved.
>
```

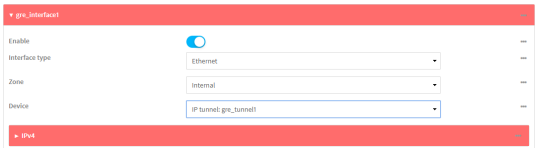
Task four: Create an interface for the GRE tunnel device

WebUI

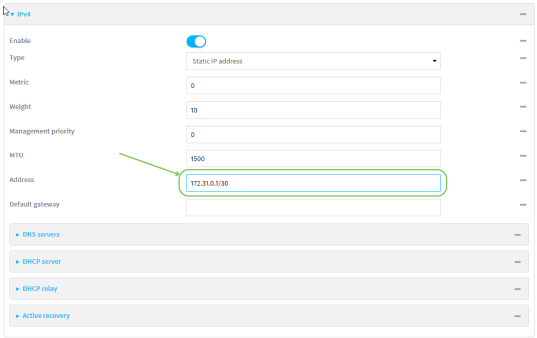
1. Click **Network** > **Interfaces**.
2. For **Add Interface**, type **gre_interface1** and click ✚.



3. For **Zone**, select **Internal**.
4. For **Device**, select the GRE tunnel created in Task three (**IP tunnel: gre_tunnel1**).



5. Click to expand **IPv4**.
6. For **Address**, type **172.31.0.1/30** for a virtual IP address on the GRE tunnel.



7. Click **Apply** to save the configuration and apply the change.



Command line

1. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

2. Add an interface named **gre_interface1**:

```
(config)> add network interface gre_interface1
(config network interface gre_interface1)>
```

3. Set the zone to **internal**:

```
(config network interface gre_interface1)> zone internal
(config network interface gre_interface1)>
```

4. Set the device to the GRE tunnel created in Task three (**/vpn/iptunnel/gre_tunnel1**):

```
(config network interface gre_interface1)> device /vpn/iptunnel/gre_
tunnel1
(config network interface gre_interface1)>
```

5. Set **172.31.0.1/30** as the virtual IP address on the GRE tunnel:

```
(config network interface gre_interface1)> ipv4 address 172.31.0.1/30
(config network interface gre_interface1)>
```

6. Save the configuration and apply the change:

```
(config network interface gre_interface1)> save
Configuration saved.
>
```

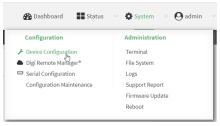
7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

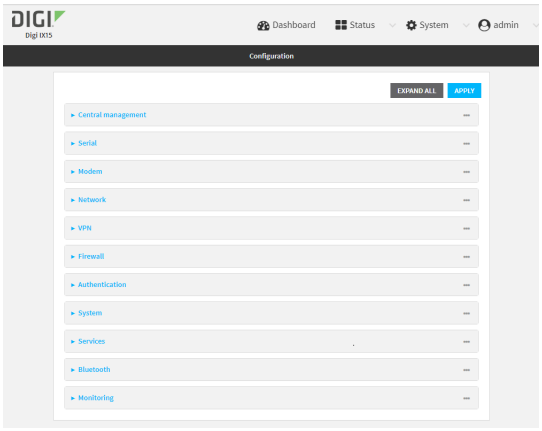
Configure the IX15-2 device
Task one: Create an IPsec tunnel

WebUI

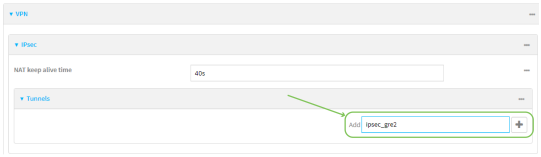
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



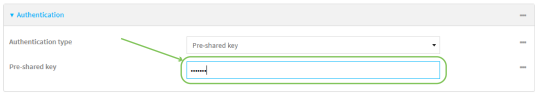
   The **Configuration** window is displayed.



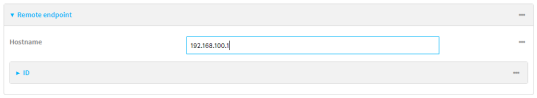
3. Click **VPN** > **IPsec** > **Tunnels**.
4. For **Add IPsec Tunnel**, type **ipsec_gre2** and click ✚.



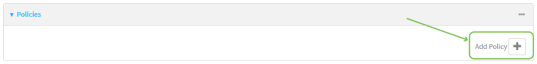
5. Click to expand **Authentication**.
6. For **Pre-shared key**, type the same pre-shared key that was configured for the IX15-1 (**testkey**).



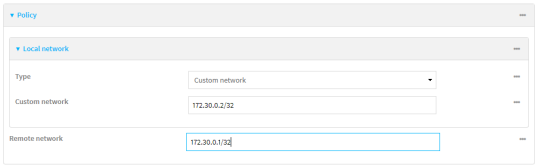
7. Click to expand **Remote endpoint**.
8. For **Hostname**, type public IP address of the IX15-1 device.



9. Click to expand **Policies**.
10. For **Add Policy**, click ✚ to add a new policy.



11. Click to expand **Local network**.
12. For **Type**, select **Custom network**.
13. For **Address**, type the IP address and subnet of the local GRE tunnel, **172.30.0.2/32**.
14. For **Remote network**, type the IP address and subnet of the remote GRE tunnel, **172.30.0.1/32**.



15. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add an IPsec tunnel named **ipsec_gre2**:

```
(config)> add vpn ipsec tunnel ipsec_gre2
(config vpn ipsec tunnel ipsec_gre2)>
```

4. Set the pre-shared key to the same pre-shared key that was configured for the IX15-1 (**testkey**):

```
(config vpn ipsec tunnel ipsec_gre2)> auth secret testkey
(config vpn ipsec tunnel ipsec_gre2)>
```

5. Set the remote endpoint to public IP address of the IX15-1 device:

```
(config vpn ipsec tunnel ipsec_gre2)> remote hostname 192.168.100.1
(config vpn ipsec tunnel ipsec_gre2)>
```

6. Add a policy:

```
(config vpn ipsec tunnel ipsec_gre2)> add policy end
(config vpn ipsec tunnel ipsec_gre2 policy 0)>
```

7. Set the local network policy type to **custom**:

```
(config vpn ipsec tunnel ipsec_gre2 policy 0)> local type custom
(config vpn ipsec tunnel ipsec_gre2 policy 0)>
```

8. Set the local network address to the IP address and subnet of the local GRE tunnel, **172.30.0.2/32**:

```
(config vpn ipsec tunnel ipsec_gre2 policy 0)> local custom 172.30.0.2/32
(config vpn ipsec tunnel ipsec_gre2 policy 0)>
```

9. Set the remote network address to the IP address and subnet of the remote GRE tunnel, **172.30.0.1/32**:

```
(config vpn ipsec tunnel ipsec_gre2 policy 0)> remote network
172.30.0.1/32
(config vpn ipsec tunnel ipsec_gre2 policy 0)>
```

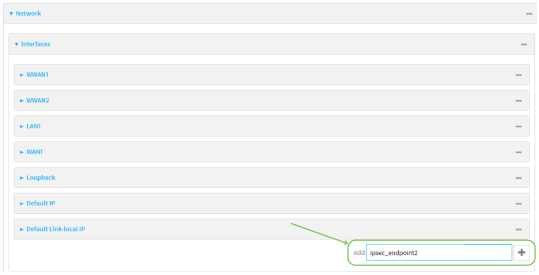
10. Save the configuration and apply the change:

```
(config vpn ipsec tunnel ipsec_gre2 policy 0)> save
Configuration saved.
>
```

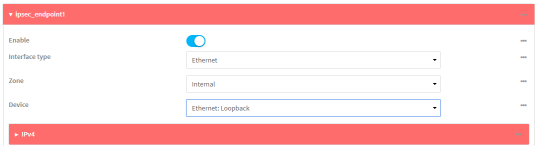
Task two: Create an IPsec endpoint interface

WebUI

1. Click **Network** > **Interfaces**.
2. For **Add Interface**, type **ipsec_endpoint2** and click ✚.

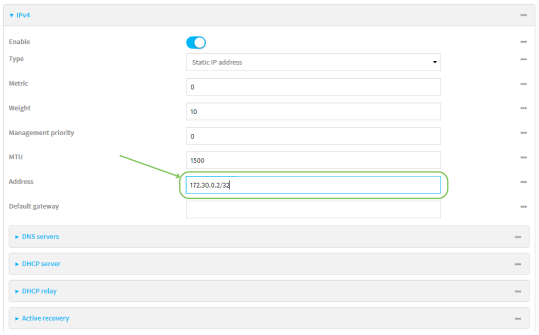


3. For **Zone**, select **Internal**.
4. For **Device**, select **Ethernet: loopback**.



5. Click to expand **IPv4**.

6. For **Address**, type the IP address of the local GRE tunnel, **172.30.0.2/32**.



7. Click **Apply** to save the configuration and apply the change.



Command line

1. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

2. Add an interface named **ipsec_endpoint2**:

```
(config)> add network interface ipsec_endpoint2
(config network interface ipsec_endpoint2)>
```

3. Set the zone to **internal**:

```
(config network interface ipsec_endpoint2)> zone internal
(config network interface ipsec_endpoint2)>
```

4. Set the device to **/network/device/loopback**:

```
(config network interface ipsec_endpoint2)> device
/network/device/loopback
(config network interface ipsec_endpoint2)>
```

5. Set the IPv4 address to the IP address of the local GRE tunnel, **172.30.0.2/32**:

```
(config network interface ipsec_endpoint2)> ipv4 address 172.30.0.2/32
(config network interface ipsec_endpoint2)>
```

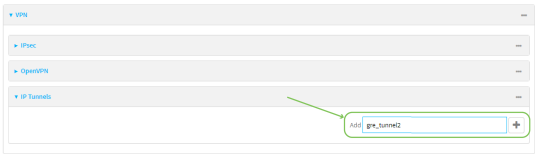
6. Save the configuration and apply the change:

```
(config vpn ipsec tunnel ipsec_endpoint2)> save
Configuration saved.
>
```

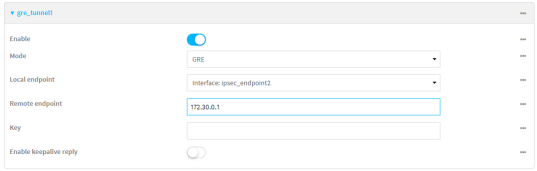
Task three: Create a GRE tunnel

WebUI

1. Click **VPN** > **IP Tunnels**.
2. For **Add IP Tunnel**, type **gre_tunnel2** and click **+**.



3. For **Local endpoint**, select the IPsec endpoint interface created in Task two (**Interface: ipsec_endpoint2**).
4. For **Remote endpoint**, type the IP address of the GRE tunnel on IX15-1, **172.30.0.1**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

2. Add a GRE tunnel named **gre_tunnel2**:

```
(config)> add vpn iptunnel gre_tunnel2
(config vpn iptunnel gre_tunnel2)>
```

3. Set the local endpoint to the IPsec endpoint interface created in Task two (**/network/interface/ipsec_endpoint2**):

```
(config vpn iptunnel gre_tunnel2)> local /network/interface/ipsec_
endpoint2
(config vpn iptunnel gre_tunnel2)>
```

4. Set the remote endpoint to the IP address of the GRE tunnel on IX15-1, **172.30.0.1**:

```
(config vpn iptunnel gre_tunnel2)> remote 172.30.0.1
(config vpn iptunnel gre_tunnel2)>
```

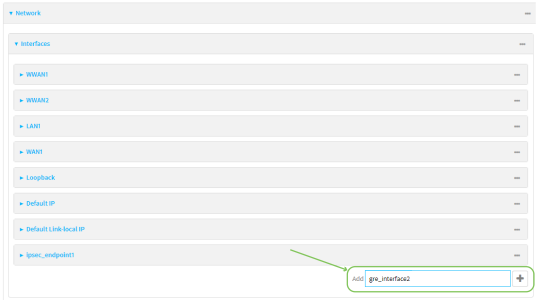
5. Save the configuration and apply the change:

```
(config vpn iptunnel gre_tunnel2)> save
Configuration saved.
>
```

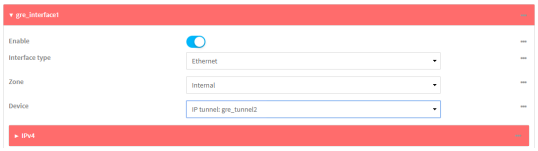
Task four: Create an interface for the GRE tunnel device

WebUI

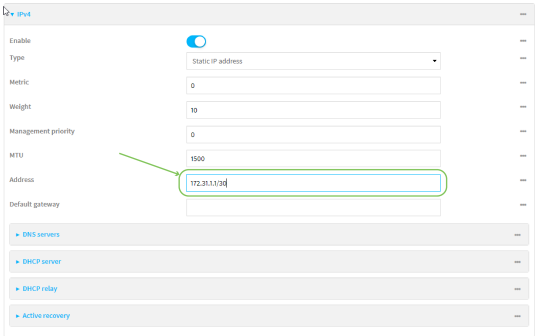
1. Click **Network** > **Interfaces**.
2. For **Add Interface**, type **gre_interface2** and click **+**.



3. For **Zone**, select **Internal**.
4. For **Device**, select the GRE tunnel created in Task three (**IP tunnel: gre_tunnel2**).



5. Click to expand **IPv4**.
6. For **Address**, type **172.31.1.1/30** for a virtual IP address on the GRE tunnel.



7. Click **Apply** to save the configuration and apply the change.



Command line

1. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

2. Add an interface named **gre_interface2**:

```
(config)> add network interface gre_interface2
(config network interface gre_interface2)>
```

3. Set the zone to **internal**:

```
(config network interface gre_interface2)> zone internal
(config network interface gre_interface2)>
```

4. Set the device to the GRE tunnel created in Task three (**/vpn/iptunnel/gre_tunnel2**):

```
(config network interface gre_interface2)> device /vpn/iptunnel/gre_
tunnel2
(config network interface gre_interface2)>
```

5. Set **172.31.0.1/30** as the virtual IP address on the GRE tunnel:

```
(config network interface gre_interface2)> ipv4 address 172.31.1.1/30
(config network interface gre_interface2)>
```

6. Save the configuration and apply the change:

```
(config network interface gre_interface2)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

L2TP

Your IX15 device supports PPP-over-L2TP (Layer 2 Tunneling Protocol).

Configure a PPP-over-L2TP tunnel

Your IX15 device supports PPP-over-L2TP (Layer 2 Tunneling Protocol). The tunnel endpoints are known as L2TP Access Concentrators (LAC) and L2TP Network Servers (LNS). Each endpoint terminates the PPP session.

Required configuration items

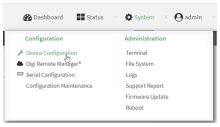
- For L2TP access concentrators:
  - The hostname or IP address of the L2TP network server.
  - The firewall zone for the tunnel.
- For L2TP network servers:
  - The IP address of the L2TP access concentrator.
  - The local IP address assigned to the L2TP virtual network interface.
  - The IP address assigned to the remote peer.
  - The firewall zone for the tunnel.

Additional configuration items

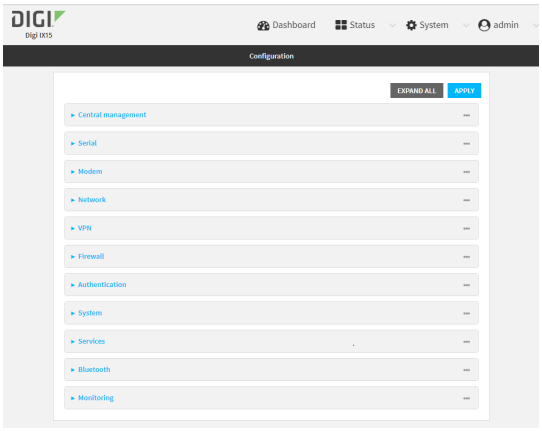
- The UDP port that L2TP servers will listen on, if other than the deafult of **1701**.
- Access control for the L2TP tunnel.
- For L2TP access concentrators:
  - L2TP network server port.
  - The username and password of the L2TP server.
  - The metric for the tunnel.
  - Enable custom PPP configuration options for the tunnel.
    - Whether to override the default configuration and only use the custom options.
    - Optional configuration data in the format of a pppd options file.
  - SureLink options for the tunnel.
- For L2TP network servers:
  - The Authentication method.
  - The metric for the tunnel.
  - Enable custom PPP configuration options for the tunnel.
    - Whether to override the default configuration and only use the custom options.
    - Optional configuration data in the format of a pppd options file.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **VPN** > **L2TP**.
4. (Optional) Type the **UDP listening port** that L2TP servers will listen on, if other than the default of **1701**.
5. Set the access control for L2TP tunnels:
   - To limit access to specified IPv4 addresses and networks:
     a. Click **IPv4 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv4 address or network that can access the device's service-type. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 192.168.1.0/24.
        - **any**: No limit to IPv4 addresses that can access the service-type.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:
     a. Click **IPv6 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv6 address or network that can access the device's service-type. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 2001:db8::/48.
        - **any**: No limit to IPv6 addresses that can access the service-type.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to hosts connected through a specified interface on the IX15 device:
     a. Click **Interfaces**.
     b. For **Add Interface**, click ✚.
     c. For **Interface**, select the appropriate interface from the dropdown.
     d. Click ✚ again to allow access through additional interfaces.

- To limit access based on firewall zones:
  a. Click **Zones**.
  b. For **Add Zone**, click ✚.
  c. For **Zone**, select the appropriate firewall zone from the dropdown.
     See Firewall configuration for information about firewall zones.
  d. Click ✚ again to allow access through additional firewall zones.

6. To add an L2TP access concentrator:
   a. Click to expand **L2TP access concentrators**.
   b. For **Add L2TP access concentrator**, type a name for the LAC and click ✚.
   c. LACs are enabled by default. To disable, click to toggle off **Enable**.
   d. For **L2TP network server**, type the hostname or IP address of the L2TP network server.
   e. (Optional) Type the **L2TP network server port** to use to connect to the server, if other than the default of **1701**.
   f. (Optional) Type the **Username** to use to log into the server.
   g. (Optional) Type the **Password** to use to log into the server.
   h. (Optional) Type the **Metric** for the tunnel, if other than the default of **1**.
   i. Select a firewall **Zone** for the tunnel. This is used by packet filtering rules and access control lists to restrict network traffic on the tunnel.
   j. (Optional): Custom PPP configuration:
      i. **Enable** custom PPP configuration.
      ii. Enable **Override** if the custom configuration should override the default configuration and only use the custom options.
      iii. For **Configuration file**, paste or type the configuration data in the format of a pppd options file.
   k. For SureLink, see Configure SureLink active recovery for PPP-over-L2TP.
7. To add an L2TP network server:
   a. Click to expand **L2TP network servers**.
   b. For **Add L2TP network server**, type a name for the LNS and click ✚.
   c. LNSs are enabled by default. To disable, click to toggle off **Enable**.
   d. For **L2TP access concentrator**, type the IP addressof the L2TP access concentrator that this server will allow connections from. This can also be:
      - A range of IP addresses, using the format x.x.x.x-y.y.y.y, for example **192.168.188.1-192.168.188.254**.
      - The keyword **any**, which means that the server will accept connections from any IP address.
   e. For **Local IP address**, type the IP address of the L2TP virtual network interface.
   f. For **Remote IP address**, type the IP address to assign to the remote peer.
   g. (Optional) For **Authentication method**, select one of the following:
      - **None**: No authentication is required.
      - **Automatic**: The device will attempt to connect using CHAP first, and then PAP.

- **CHAP**: Uses the Challenge Handshake Authentication Profile (CHAP) to authenticate.
- **PAP**: Uses the Password Authentication Profile (PAP) to authenticate.

If **Automatic**, **CHAP**, or **PAP** is selected, enter the **Username** and **Password** required to authenticate.

The default is **None**.

h. (Optional) For **Authentication method**, select the authentication method, one of:
   - **None**: No authentication is required.
   - **Automatic**: The device will attempt to connect using CHAP first, and then PAP.
   - **CHAP**: Uses the Challenge Handshake Authentication Profile (CHAP) to authenticate.
   - **PAP**: Uses the Password Authentication Profile (PAP) to authenticate.
   - If **Automatic**, **CHAP**, or **PAP** is selected, enter the **Username** and **Password** required to authenticate.
   - The default is **None**.

i. (Optional) Type the **Metric** for the tunnel, if other than the default of **1**.

j. Select a firewall **Zone** for the tunnel. This is used by packet filtering rules and access control lists to restrict network traffic on the tunnel.

k. (Optional): Custom PPP configuration:
   i. **Enable** custom PPP configuration.
   ii. Enable **Override** if the custom configuration should override the default configuration and only use the custom options.
   iii. For **Configuration file**, paste or type the configuration data in the format of a pppd options file.

8. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. (Optional) Set the UDP listening port that L2TP servers will listen on:

```
(config)> vpn l2tp port value
(config)>
```

where *value* is an integer between **1** and **65535**. The default is **1701**.

4. Set the access control for L2TP tunnels:
   - To limit access to specified IPv4 addresses and networks:

     ```
     (config)> add vpn l2tp acl address end value
     (config)>
     ```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 192.168.1.0/24.
     - **any**: No limit to IPv4 addresses that can access the service-type.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:

     ```
     (config)> add vpn l2tp acl address6 end value
     (config)>
     ```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 2001:db8::/48.
     - **any**: No limit to IPv6 addresses that can access the service-type.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to hosts connected through a specified interface on the IX15 device:

     ```
     (config)> add vpn l2tp acl interface end value
     (config)>
     ```

     Where *value* is an interface defined on your device.

     Display a list of available interfaces:

     Use **... network interface ?** to display interface information:

     ```
     (config)> ... network interface ?

     Interfaces

     Additional Configuration
     ------------------------------------------
      defaultip              Default IP
      defaultlinklocal       Default Link-local IP
      eth                    ETH
      loopback               Loopback
      modem                  Modem

     config)>
     ```

     Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config)> add vpn l2tp acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.

 Additional Configuration
 ------------------------------------------------------------
------------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

(config)>
```

Repeat this step to list additional firewall zones.

5. To add an L2TP access concentrator:
   a. Add an LAC:

```
(config)> add vpn l2tp lac name
(config add vpn l2tp lac name)>
```

where *name* is the name of the LAC. For example, to add an LAC named lac_tunnel:

```
(config)> add vpn l2tp lac lac_tunnel
(config vpn l2tp lac lac_tunnel)>
```

LACs are enabled by default. To disable:

```
(config vpn l2tp lac lac_tunnel)> enable false
(config vpn l2tp lac lac_tunnel)>
```

   b. Set the hostname or IP address of the L2TP network server:

```
(config vpn l2tp lac lac_tunnel)> lns hostname
(config vpn l2tp lac lac_tunnel)>
```

c. (Optional) Set the UDP port to use to connect to the L2TP network server:

```
(config vpn l2tp lac lac_tunnel)> port int
(config vpn l2tp lac lac_tunnel)>
```

where *int* is an integer between **1** and **65535**. The default is **1701**.

d. (Optional) Set the username to use to log into the server:

```
(config vpn l2tp lac lac_tunnel)> username username
(config vpn l2tp lac lac_tunnel)>
```

e. (Optional) Set the password to use to log into the server:

```
(config vpn l2tp lac lac_tunnel)> password password
(config vpn l2tp lac lac_tunnel)>
```

f. (Optional) Set the metric for the tunnel:

```
(config vpn l2tp lac lac_tunnel)> metric int
(config vpn l2tp lac lac_tunnel)>
```

where *int* is an integer between **0** and **65535**. The default is **1**.

g. Set the firewall zone for the tunnel. This is used by packet filtering rules and access control lists to restrict network traffic on the tunnel.

i. Use the **?** to determine available zones:

```
(config vpn l2tp lac lac_tunnel)> zone ?

Zone: The firewall zone assigned to this tunnel. This can be used
by packet
filtering rules and access control lists to restrict network
traffic on this
tunnel.
Format:
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup
Current value:

(config vpn l2tp lac lac_tunnel)>
```

ii. Set the zone:

```
(config vpn l2tp lac lac_tunnel)> zone zone
(config vpn l2tp lac lac_tunnel)>
```

h. (Optional): Custom PPP configuration:

i. Enable custom PPP configuration:

```
(config vpn l2tp lac lac_tunnel)> custom enable true
(config vpn l2tp lac lac_tunnel)>
```

ii. Enable overriding, if the custom configuration should override the default configuration and only use the custom options:

```
(config vpn l2tp lac lac_tunnel)> custom override true
(config vpn l2tp lac lac_tunnel)>
```

iii. Paste or type the configuration data in the format of a pppd options file:

```
(config vpn l2tp lac lac_tunnel)> custom config_file data
(config vpn l2tp lac lac_tunnel)>
```

i. To configure SureLink active recovery for the L2TP access connector, see Configure SureLink active recovery for PPP-over-L2TP.

6. To add an L2TP network server:

a. Add an LNS:

```
(config)> add vpn l2tp lns name
(config add vpn l2tp lac name)>
```

where *name* is the name of the LNS. For example, to add an LNS named lns_server:

```
(config)> add vpn l2tp lns lns_server
(config vpn l2tp lns lns_server)>
```

LACs are enabled by default. To disable:

```
(config vpn l2tp lns lns_server)> enable false
(config vpn l2tp lns lns_server)>
```

b. Set the IP address of the L2TP access concentrator that this server will allow connections from:

```
(config vpn l2tp lns lns_server)> lac IP_address
(config vpn l2tp lns lns_server)>
```

This can also be:

- A range of IP addresses, using the format x.x.x.x-y.y.y.y, for example **192.168.188.1-192.168.188.254**.
- The keyword **any**, which means that the server will accept connections from any IP address.

c. Set the IP address of the L2TP virtual network interface:

```
(config vpn l2tp lns lns_server)> local_address IP_address
(config vpn l2tp lns lns_server)>
```

d. Set the IP address to assign to the remote peer:

```
(config vpn l2tp lns lns_server)> remote_address IP_address
(config vpn l2tp lns lns_server)>
```

e. (Optional) Set the authentication method:

```
(config vpn l2tp lns lns_server)> auth method
(config)>
```

where *method* is one of the following:

- **none**: No authentication is required.
- **auto**: The device will attempt to connect using CHAP first, and then PAP.
- **chap**: Uses the Challenge Handshake Authentication Profile (CHAP) to authenticate.
- **pap**: Uses the Password Authentication Profile (PAP) to authenticate.

If **auto**, **chap**, or **pap** is selected, enter the **Username** and **Password** required to authenticate:

```
(config vpn l2tp lns lns_server)> username username
(config vpn l2tp lns lns_server)> password password
(config vpn l2tp lns lns_server)>
```

The default is **none**.

f. (Optional) Set the metric for the tunnel:

```
(config vpn l2tp lns lns_server)> metric int
(config vpn l2tp lns lns_server)>
```

where *int* is an integer between **0** and **65535**. The default is **1**.

g. Set the firewall zone for the tunnel. This is used by packet filtering rules and access control lists to restrict network traffic on the tunnel.

i. Use the **?** to determine available zones:

```
(config vpn l2tp lns lns_server)> zone ?

Zone: The firewall zone assigned to this tunnel. This can be used
by packet
filtering rules and access control lists to restrict network
traffic on this
tunnel.
Format:
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup
```

```
Current value:

(config vpn l2tp lns lns_server)>
```

ii. Set the zone:

```
(config vpn l2tp lns lns_server)> zone zone
(config vpn l2tp lns lns_server)>
```

h. (Optional): Custom PPP configuration:

i. Enable custom PPP configuration:

```
(config vpn l2tp lac lns lns_server)> custom enable true
(config vpn l2tp lns lns_server)>
```

ii. Enable overriding, if the custom configuration should override the default configuration and only use the custom options:

```
(config vpn l2tp lns lns_server)> custom override true
(config vpn l2tp lns lns_server)>
```

iii. Paste or type the configuration data in the format of a pppd options file:

```
(config vpn l2tp lns lns_server)> custom config_file data
(config vpn l2tp lns lns_server)>
```

7. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

8. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure SureLink active recovery for PPP-over-L2TP

You can configure the IX15 device to regularly probe PPP-over-L2TP access concatenators to determine if the connection has failed and take remedial action.

Required configuration items

- A valid PPP-over-L2TP configuration. See Configure a PPP-over-L2TP tunnel for configuration instructions.
- Enable PPP-over-L2TP active recovery.
- The behavior of the IX15 device upon PPP-over-L2TP failure: either
  - Restart the PPP-over-L2TP interface
  - Reboot the device.

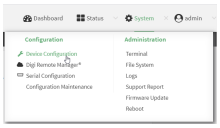
Additional configuration items

- The interval between connectivity tests.
- Whether the interface should be considered to have failed if one of the test targets fails, or all of the test targets fail.
- The number of probe attempts before the PPP-over-L2TP connection is considered to have failed.
- The amount of time that the device should wait for a response to a probe attempt before considering it to have failed.

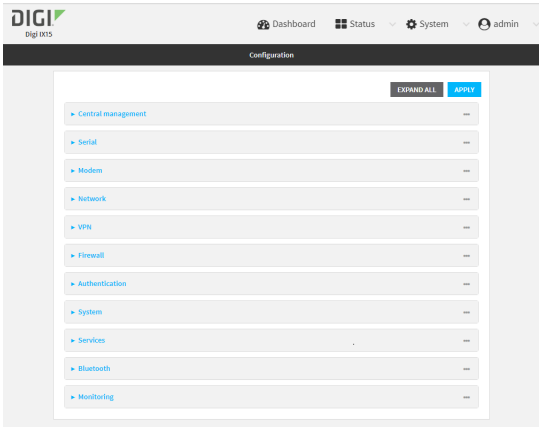
To configure the IX15 device to regularly probe the PPP-over-L2TP connection:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

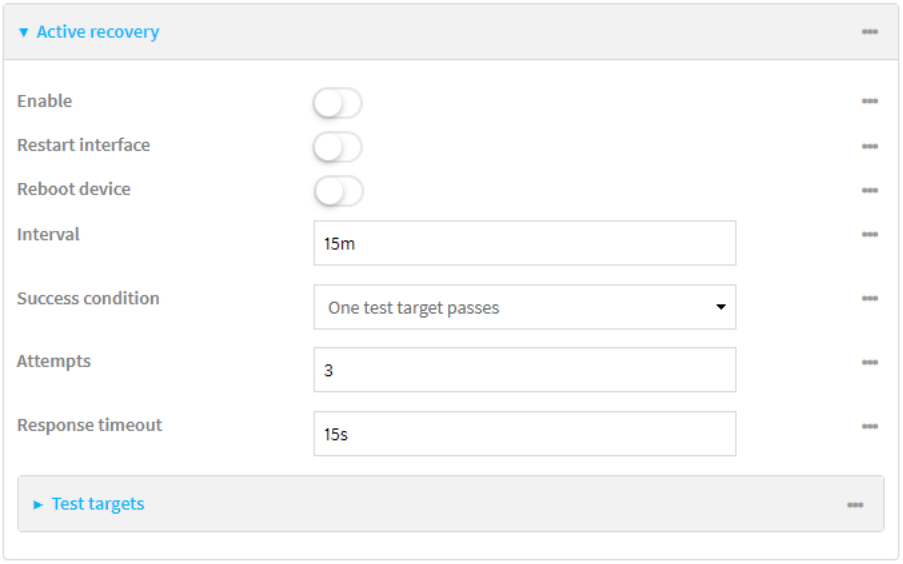


The **Configuration** window is displayed.



3. Click **VPN** > **PPP-over-L2TP**.
4. Create a new PPP-over-L2TP access concatenator or select an existing one:
   - To create a new L2TP access concatenator , see Configure a PPP-over-L2TP tunnel.
   - To edit an existing L2TP access concatenator , click to expand the appropriate tunnel.

5. After creating or selecting the PPP-over-L2TP access concatenator, click **Active recovery**.



6. **Enable** active recovery.
7. For **Restart interface**, enable to configure the device to restart the interface when its connection is considered to have failed. This is useful for interfaces that may regain connectivity after restarting, such as a cellular modem.
8. For **Reboot device**, enable to instruct the device to reboot when the WAN connection is considered to have failed.
9. Change the **Interval** between connectivity tests.

   Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

   For example, to set **Interval** to ten minutes, enter **10m** or **600s**.

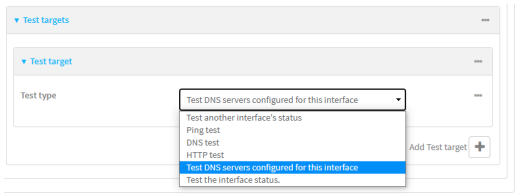
   The default is 15 minutes.
10. For **Success condition**, determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets.
11. For **Attempts**, type the number of probe attempts before the WAN is considered to have failed.
12. For **Response timeout**, type the amount of time that the device should wait for a response to a probe attempt before considering it to have failed.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Response timeout** to ten minutes, enter **10m** or **600s**.

    The default is 15 seconds.
13. Add a test target:
    a. Click to expand **Test targets**.
    b. For **Add Test target**, click **+**.

c. Select the **Test type**:



- **Test another interface's status**: Allows you to test another interface's status, to create a failover or coupled relationship between interfaces. If **Test another interface's status** is selected:
  - For **Test Interface**, select the alternate interface to be tested.
  - For **IP version**, select the alternate interface's IP version. This allows you to determine the alternate interface's status for a particular IP version.
  - For **Expected status**, select whether the expected status of the alternate interface is **Up** or **Down**. For example, if **Expected status** is set to **Down**, but the alternate interface is determined to be up, then this test will fail.
- **Ping test**: Tests connectivity by sending an ICMP echo request to the hostname or IP address specified in **Ping host**. You can also optionally change the number of bytes in the **Ping payload size**.
- **DNS test**: Tests connectivity by sending a DNS query to the specified **DNS server**.
- **HTTP test**: Tests connectivity by sending an HTTP or HTTPS GET request to the URL specified in **Web servers**. The URL should take the format of **http[s]://*hostname*|[*path*]**.
- **Test DNS servers configured for this interface**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
- **Test the interface status**: The interface is considered to be down based on:
  - **Down time**: The amount of time that the interface can be down before this test is considered to have failed.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Down time** to ten minutes, enter **10m** or **600s**.

    The default is 60 seconds.
  - **Initial connection time**: The amount of time to wait for an initial connection to the interface before this test is considered to have failed.

    Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

    For example, to set **Initial connection time** to ten minutes, enter **10m** or **600s**.

    The default is 60 seconds.

14. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Create a new PPP-over-L2TP access concatenator , or edit an existing one:
   - To create a new PPP-over-L2TP access concatenator , see Configure a PPP-over-L2TP tunnel.
   - To edit an existing PPP-over-L2TP access concatenator , change to the PPP-over-L2TP access concatenator node in the configuration schema. For example, for an PPP-over-L2TP access concatenator named **lac_tunnel**, change to the **lac_tunnel** node in the configuration schema:

```
(config)> vpn l2tp lac lac_tunnel
(config vpn l2tp lac lac_tunnel)>
```

4. Enable active recovery:

```
(config vpn l2tp lac lac_tunnel)> surelink enable true
(config vpn l2tp lac lac_tunnel)>
```

5. To configure the device to restart the interface when its connection is considered to have failed:

```
(config vpn l2tp lac lac_tunnel)> surelink restart true
(config vpn l2tp lac lac_tunnel)>
```

   This is useful for interfaces that may regain connectivity after restarting, such as a cellular modem.

6. To configure the device to reboot when the interface is considered to have failed:

```
(config vpn l2tp lac lac_tunnel)> surelink reboot enable
(config vpn l2tp lac lac_tunnel)>
```

7. Set the **Interval** between connectivity tests:

```
(config vpn l2tp lac lac_tunnel)> surelink interval value
(config vpn l2tp lac lac_tunnel)>
```

   where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

   For example, to set **interval** to ten minutes, enter either **10m** or **600s**:

```
(config vpn l2tp lac lac_tunnel)> surelink interval 600s
(config vpn l2tp lac lac_tunnel)>
```

   The default is 15 minutes.

8. Determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets:

```
(config vpn l2tp lac lac_tunnel)> surelink success_condition value
(config vpn l2tp lac lac_tunnel)>
```

   Where *value* is either **one** or **all**.

9. Set the number of probe attempts before the WAN is considered to have failed:

```
(config vpn l2tp lac lac_tunnel)> surelink  attempts num
(config vpn l2tp lac lac_tunnel)>
```

   The default is **3**.

10. Set the amount of time that the device should wait for a response to a probe attempt before considering it to have failed:

```
(config vpn l2tp lac lac_tunnel)> surelink timeout value
(config vpn l2tp lac lac_tunnel)>
```

   where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

   For example, to set **interval** to ten minutes, enter either **10m** or **600s**:

```
(config vpn l2tp lac lac_tunnel)> surelink timeout 600s
(config vpn l2tp lac lac_tunnel)>
```

   The default is 15 seconds.

11. Configure test targets:
    a. Add a test target:

```
(config vpn l2tp lac lac_tunnel)> add surelink target end
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

    b. Set the test type:

```
(config vpn l2tp lac lac_tunnel surelink target 0)> test value
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

    where *value* is one of:

    - **ping**: Tests connectivity by sending an ICMP echo request to a specified hostname or IP address.
      - Specify the hostname or IP address:

```
(config vpn l2tp lac lac_tunnel surelink target 0)> ping_host
host
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

      - (Optional) Set the size, in bytes, of the ping packet:

```
(config vpn l2tp lac lac_tunnel surelink target 0)> ping_size
[num]
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

- **dns**: Tests connectivity by sending a DNS query to the specified DNS server.
  - Specify the DNS server. Allowed value is the IP address of the DNS server.

```
(config vpn l2tp lac lac_tunnel surelink target 0)> dns_
server ip_address
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

- **dns_configured**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
- **http**: Tests connectivity by sending an HTTP or HTTPS GET request to the specified URL.
  - Specify the url:

```
(config vpn l2tp lac lac_tunnel surelink target 0)> http_url
value
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

    where *value* uses the format **http[s]://*hostname*/[*path*]**
- **interface_up**: The interface is considered to be down based on the interfaces down time, and the amount of time an initial connection to the interface takes before this test is considered to have failed.
  - (Optional) Set the amount of time that the interface can be down before this test is considered to have failed:

```
(config vpn l2tp lac lac_tunnel surelink target 0)>
interface_down_time value
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

    where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

    For example, to set **interval** to ten minutes, enter either **10m** or **600s**:

```
(config vpn l2tp lac lac_tunnel surelink target 0)>
interface_down_time 600s
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

    The default is 60 seconds.
  - (Optional) Set the amount of time to wait for an initial connection to the interface before this test is considered to have failed:

```
(config vpn l2tp lac lac_tunnel surelink target 0)>
interface_timeout value
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **interval** to ten minutes, enter either **10m** or **600s**:

```
(config vpn l2tp lac lac_tunnel surelink target 0)>
interface_timeout 600s
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

The default is 60 seconds.

- **other**: Allows you to test another interface's status, to create a failover or coupled relationship between interfaces:

```
(config vpn l2tp lac lac_tunnel surelink target 0)> other
value
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

If **other** is set:

- Set the alternate interface to be tested:
  i. Use the **?** to determine available interfaces:
  ii. Set the interface. For example:

```
(config vpn l2tp lac lac_tunnel surelink target 0)>
other_interface /network/interface/eth1
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

- Set the alternate interface's IP version. This allows you to determine the alternate interface's status for a particular IP version.

```
(config vpn l2tp lac lac_tunnel surelink target 0)> other_
ip_version value
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

where *value* is one of: **any**, **both**, **ipv4**, or **ipv6**.

- Set the expected status of the alternate interface:

```
(config vpn l2tp lac lac_tunnel surelink target 0)> other_
status value
(config vpn l2tp lac lac_tunnel surelink target 0)>
```

where *value* is either **up** or **down**. For example, if **other_status** is set to **down**, but the alternate interface is determined to be up, then this test will fail.

See Show Surelink status and statistics for information about showing Surelink status for IPsec tunnels.

L2TP with IPsec

L2TP is commonly used in conjunction with IPsec in transport mode (to provide security).

Your IX15 supoorts L2TP with IPsec by configuring a transport-mode IPsec tunnel between the two endpoints, and then an L2TP tunnel with its LNS and LAC configured the same as the IPsec tunnel's endpoints. See Configure an IPsec tunnel for information about configuring an IPsec tunnel.

Note The IX15 does not currently support the configuration of IPsec protocol/port traffic selectors. This means that you cannot restrict traffic on the IPsec tunnel to L2TP traffic (typically UDP port 1701).

While multiple L2TP clients are supported on the IX15 by configuring a separate LNS for each client, multiple clients behind a Network Address Translation (NAT) device are not supported, because they will all appear to have the same IP address.

Show L2TP tunnel status

WebUI

Show the status of L2TP access connectors from the WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, select **Status**. Under VPN, select **L2TP** > **Access Connectors**.

   The **L2TP Access Connectors** page appears.
3. To view configuration details about an L2TP access connector, click the (configuration) icon in the upper right of the tunnel's status pane.

Show the status of L2TP network servers from the WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, select **Status**. Under VPN, select **L2TP** > **Network Servers**.

   The **L2TP Network Servers** page appears.
3. To view configuration details about an L2TP network server, click the (configuration) icon in the upper right of the tunnel's status pane.

Command line

Show the status of L2TP access connectors from the Admin CLI

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. To display details about all configured L2TP access connectors, type the following at the prompt:

```
> show l2tp lac

 Name       Enabled  Status  Device
 ---------  -------  ------  -----------
 lac_test1  true     up      test_device0
 lac_test2  true     pending
>
```

3. To display details about a specific tunnel:

```
> show l2tp lac name lac_test2

 lac_test2 L2TP Access Concentrator Status
 -------------------------------------
 Enabled         : true
 Status          : pending

>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show the status of L2TP network servers from the Admin CLI

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. To display details about all configured L2TP access connectors, type the following at the prompt:

```
> show l2tp lns

 Name       Enabled  Status  Device
 ---------  -------  ------  -----------
 lns_test1  true     up      test_device0
 lns_test2  true     pending

>
```

3. To display details about a specific tunnel:

```
> show l2tp lns name lns_test2

 lns_test2 L2TP Access Concentrator Status
 -------------------------------------
 Enabled         : true
 Status          : pending

>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

L2TPv3 Ethernet

Your IX15 device supports Layer 2 Tunneling Protocol Version 3 (L2TPv3) static unmanaged Ethernet tunnels.

Configure an L2TPv3 tunnel

Your IX15 device supports Layer 2 Tunneling Protocol Version 3 (L2TPv3) static unmanaged Ethernet tunnels.

Required configuration items

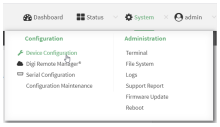
- A name for the L2TPv3 tunnel.
- Enable the tunnel.
- The remote endpoint IP address.
- The local endpoint IP address.
- The session ID.
- The peer session ID.

Additional configuration items

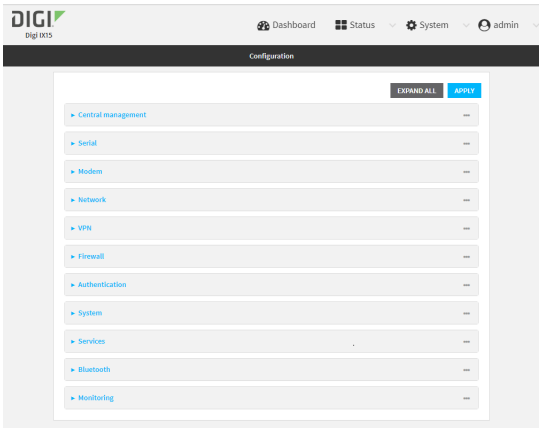
- Encapsulation type. If UDP is selected:
  - The ID for the tunnel.
  - The ID of the peer's tunnel.
  - Determine whether to enable UDP checksum.
- The session cookie.
- The peer session cookie.
- The Layer2SpecificHeader type.
- The Sequence numbering control.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **VPN** > **L2TPv3 ethernet**.
4. For **Add L2TPv3 ethernet tunnel**, type a name for the tunnel and click ✚.
5. For **Remote endpoint**, type the IPv4 address of the remote endpoint.
6. For **Local endpoint**, select the interface that will be the local endpoint.
7. For **Tunnel ID**, type the tunnel identifier for this tunnel. This must match the value for **Peer tunnel ID** on the remote peer. Allowed value is any integer between 1 and 4294967295.
8. For **Peer tunnel ID**, type the **Tunnel ID** of the remote peer.
9. (Optional) For **Encapsulation type**, select either **UDP** or **IP**. If **UDP** is selected:
    a. For **UDP source port**, type the number of the source UDP port to be used for the tunnel.
    b. For **UDP destination port**, type the number of the destination UDP port to be used for the tunnel.
    c. (Optional) Click to enable **UDP checksum** to calculate and check the UDP checksum.
10. Click to expand **Sessions**.
    a. For **Add Sesssion**, type a name for a session carried by the parent tunnel and click ✚.
    b. For **Session ID**, type the session identifier for this session. This must match the value for **Peer session ID** on the remote peer. Allowed value is any integer between 1 and 4294967295.
    c. For **Peer session ID**, type the **Session ID** of the remote peer.
    d. (Optional) For **Cookie**, type the cookie value to be assigned to the session. Allowed value is 8 or 16 hex digits.
    e. (Optional) For **Peer cookie**, type the **Cookie** value of the remote peer.
    f. For **Layer2SpecificHeader type**, select the Layer2Specific header type. This must match what is configured on the remote peer.
    g. For **Sequence numbering control**, determine the sequence number control to prevent or detect out of order packets. Allowed values are:
        - **None**: No sequence numbering.
        - **Send**: Add a sequence number to each outgoing packet.
        - **Receive**: Reorder packets if they are received out of order.
        - **Both**: Add a sequence number to each outgoing packet, and reorder packets if they are received out of order.

The default is **None**.

h. Repeat for additional sessions.

11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a L2TPv3 Ethernet tunnel. For example, to add a tunnel named **L2TPv3_example**:

```
(config)> add vpn l2tpv3 L2TPv3_example
(config vpn l2tpeth L2TPv3_example)>
```

   The tunnel is enabled by default. To disable:

```
(config vpn l2tpeth L2TPv3_example)> enable false
(config vpn l2tpeth L2TPv3_example)>
```

4. Set the IPv4 address of the remote endpoint:

```
(config vpn l2tpeth L2TPv3_example)> remote IP_address
(config vpn l2tpeth L2TPv3_example)>
```

5. Set the interface of the local endpoint:

   i. Use the **?** to determine available interfaces:

   ii. Set the interface. For example:

```
(config vpn l2tpeth L2TPv3_example)> local /network/interface/eth1
(config vpn l2tpeth L2TPv3_example)>
```

6. Set the tunnel identifier for this tunnel. This must match the value for peer tunnel ID on the remote peer.

```
(config vpn l2tpeth L2TPv3_example)> tunnel_id value
(config vpn l2tpeth L2TPv3_example)>
```

   where *value* is any integer between 1 and 4294967295.

7. Set the tunnel ID of the remote peer:

```
(config vpn l2tpeth L2TPv3_example)> peer_tunnel_id value
(config vpn l2tpeth L2TPv3_example)>
```

where *value* is any integer between 1 and 4294967295.

8. (Optional) Set the encapsulation type:

```
(config vpn l2tpeth L2TPv3_example)> encapsulation  value
(config vpn l2tpeth L2TPv3_example)>
```

where *value* is either **udp** or **ip**. The default is **upd**.

If **udp** is set:

a. Set the source UDP port to be used for the tunnel:

```
(config vpn l2tpeth L2TPv3_example)> udp_source_port port
(config vpn l2tpeth L2TPv3_example)>
```

b. Set the destination UDP port to be used for the tunnel.

```
(config vpn l2tpeth L2TPv3_example)> udp_destination_port port
(config vpn l2tpeth L2TPv3_example)>
```

c. (Optional) To calculate and check the UDP checksum:

```
(config vpn l2tpeth L2TPv3_example)> udp_checksum true
(config vpn l2tpeth L2TPv3_example)>
```

9. Add a session carried by the parent tunnel:

```
(config vpn l2tpeth L2TPv3_example)> add session session_example
(config vpn l2tpeth L2TPv3_example session_example)>
```

10. Set the session identifier for this session. This must match the value for peer session ID on the remote peer.

```
(config vpn l2tpeth L2TPv3_example session_example)> session_id value
(config vpn l2tpeth L2TPv3_example session_example)>
```

where *value* is any integer between 1 and 4294967295.

11. Set the session ID of the remote peer:

```
(config vpn l2tpeth L2TPv3_example session_example)> peer_session_id
value
(config vpn l2tpeth L2TPv3_example session_example)>
```

where *value* is any integer between 1 and 4294967295.

12. (Optional) Set the cookie value to be assigned to the session.

```
(config vpn l2tpeth L2TPv3_example session_example)> cookie value
(config vpn l2tpeth L2TPv3_example session_example)>
```

Allowed *value* is 8 or 16 hex digits.

13. (Optional) Set the cookie value of the remote peer:

```
(config vpn l2tpeth L2TPv3_example session_example)> peer cookie value
(config vpn l2tpeth L2TPv3_example session_example)>
```

Allowed *value* is 8 or 16 hex digits.

14. Set the Layer2Specific header type. This must match what is configured on the remote peer.

```
(config vpn l2tpeth L2TPv3_example session_example)> l2spec_type value
(config vpn l2tpeth L2TPv3_example session_example)>
```

where *value* is either **none** or **default**. The default is **default**.

15. Set the sequence number control to prevent or detect out of order packets.

```
(config vpn l2tpeth L2TPv3_example session_example)> seq value
(config vpn l2tpeth L2TPv3_example session_example)>
```

where *value* is one of:

- **none**: No sequence numbering.
- **send**: Add a sequence number to each outgoing packet.
- **recv**: Reorder packets if they are received out of order.
- **both**: Add a sequence number to each outgoing packet, and reorder packets if they are received out of order.

  The default is **none**.

16. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

17. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show L2TPV3 tunnel status

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, select **Status**. Under VPN, select **L2TPv3 Ethernet**.

   The **L2TPv3 Ethernet** page appears.
3. To view configuration details about an L2TPV3 tunnel, click the (configuration) icon in the upper right of the tunnel's status pane.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. To display details about all configured L2TPv3 Ethernet tunnels, type the following at the prompt:

```
> show l2tpeth

 Tunnel Session     Enabled  Device        Status
 -----------------  -------  ------------  ------
 test/session/test  true     le_test_test  up

>
```

3. To display details about a specific tunnel:

```
> show l2tpeth name /vpn/l2tpeth/test/session/test

 test/session/test Tunnel Session Status
 ---------------------------------------
 Enabled                  : true
 Status                   : up


 Local IP                 : 4.3.2.1
 Remote IP                : 10.10.10.1
 Tunnel ID                : modem
 Peer Tunnel ID           : 10.10.10.1 === 4.3.2.1
 Session ID               : 255
 Peer Session ID          : 1476
 Lifetime (Actual)        : 600

 Device                   : le_test_test
 RX Packets               : 2,102
 RX Bytes                 : 462
 TX Packets               : 2,787
 TX Byptes                : 3,120

>
```

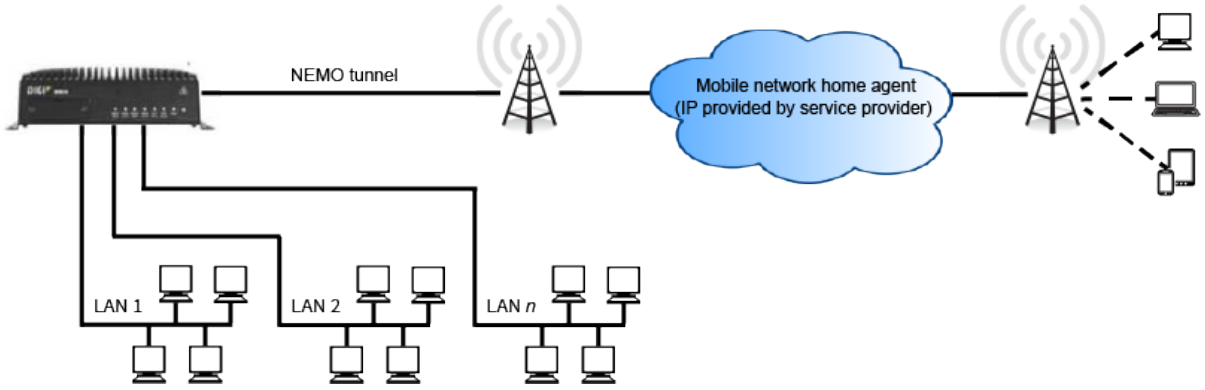
4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

NEMO

Network Mobility (NEMO) is a mobile networking technology that provides access to one or more Local Area Networks (LANs) on your device. NEMO creates a tunnel between the home agent on the mobile private network and the IX15 device, isolating the connection from internet traffic and advertising the IP subnets of the LANs for remote access and device management.

Dynamic Mobile Network Routing (DMNR) is the implementation of NEMO for Verizon Wireless Private Networks. DMNR support requires the use of Verizon SIM cards that have DMNR enabled.



Configure a NEMO tunnel

Configuring an NEMO tunnel with a remote device involves configuring the following items:

Required configuration items

- Enable the NEMO tunnel.

  The NEMO tunnel is enabled by default.
- The IP address of the NEMO virtual network interface.
- The firewall zone of the NEMO tunnel.
- The IP address of the NEMO home agent server. This is provided by your cellular carrier.
- The home agent's authentication key. This is provided by your cellular carrier.
- Home agent registration lifetime. This is provided by your cellular carrier.
- The local network interfaces that will be advertised on NEMO.

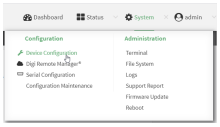
Additional configuration items

- The home agent Software Parameter Index (SPI).
- Path MTU discovery.

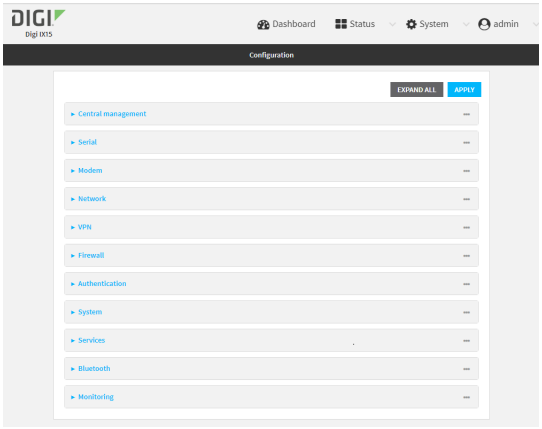
  Path MTU discovery is enabled by default. If it is disabled, identify the MTU.
- Care of address: the local network interface that is used to communicate with the peer.
  - If set to **Interface**, identify the local interface to be used. Generally, this will be the Wirelesss WAN (**Modem**).
  - If set to **IP address**, enter the IP address.
- The local network of the GRE endpoint negotiated by NEMO.
  - If the local network is set to Interface, identify the local interface to be used.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **VPN** > **NEMO**.

   The NEMO tunnel is enabled by default. To disable, click to toggle off **Enable**.
4. For **Home IP address**, type the IPv4 address of the NEMO virtual network interface.
5. For **Zone**, select **Internal**.

   The Internal firewall zone configures the IX15 device to trust traffic going to the tunnel and allows it through the network.
6. For **Home agent server IP** address, type the IPv4 address of the NEMO home agent. This is provided by your cellular carrier.
7. For **Key**, type the key used to authenticate to the home agent. This is provided by your cellular carrier.
8. For **Home agent SPI**, type the Security Parameter Index (SPI) value, which is used in the authentication extension when registering. This should be normally left at the default setting of **256** unless your service provider indicates a different value.
9. For **Home agent registration lifetime, in seconds**, type the number of seconds number of seconds until the authorization key expires. This is provided by your cellular carrier.
10. For **MTU discovery**, leave enabled to determine the maximum transmission unit (MTU) size.

    If disabled, for **MTU**, type the MTU size. The default MTU size for LANs on the IX15 device is 1500. The MTU size of the NEMO tunnel will be smaller, to take into account the required headers.
11. Click to expand **Care of address** to configure the local WAN interface of the internet facing network.
    a. For **Type**, select the method to determine the local network interface that is used to communicate with the peer.

- If **Default route** is selected, the network interface that is used will be the same as the default route.
- If **Interface** is selected, specify the local network interface.
- If **IP address** is selected, type the IP address.

The default is **Default route**.

12. Click to expand **GRE tunnel local endpoint**.
    a. For **Type**, select the local endpoint of the GRE endpoint negotiated by NEMO.
       - If **Default route** is selected, the network interface that is used will be the same as the default route.
       - If **Interface** is selected, specify the local network interface.

       The default is **Default route**.
13. Click to expand **Local networks**.
    a. For **Add Interface**, click ✚ to add a local network to use as a virtual NEMO network interface.



    b. For **Interface**, select the local interface to use as a virtual NEMO network interface. Generally, this will be the a Local Area Network (LAN).
    c. (Optional) Repeat for additional interfaces.
14. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a NEMO tunnel. For example, to add a NEMO tunnel named **nemo_example**:

```
(config)> add vpn nemo nemo_example
(config vpn nemo nemo_example)>
```

   The NEMO tunnel is enabled by default. To disable:

```
(config vpn nemo nemo_example)> enable false
(config vpn nemo nemo_example)>
```

4. Set the IPv4 address of the NEMO virtual network interface:

```
(config vpn nemo nemo_example)> home_address IPv4_address
(config vpn nemo nemo_example)>
```

5. Set the IPv4 address of the NEMO home agent. This is provided by your cellular carrier.

```
(config vpn nemo nemo_example)> home_agent IPv4_address
(config vpn nemo nemo_example)>
```

6. Set the key used to authenticate to the home agent. This is provided by your cellular carrier.

```
(config vpn nemo nemo_example)> key value
(config vpn nemo nemo_example)>
```

7. Set the the number of seconds number of seconds until the authorization key expires. This is provided by your cellular carrier.

```
(config vpn nemo nemo_example)> lifetime integer
(config vpn nemo nemo_example)>
```

   Allowed values are any integer between 1 and 65535.

8. MTU discovery is enabled by default, which allows the device to determine the maximum transmission unit (MTU) size. To disable:

```
(config vpn nemo nemo_example)> mtu_discovery false
(config vpn nemo nemo_example)>
```

   If disabled, set the MTU size. The default MTU size for LANs on the IX15 device is 1500. The MTU size of the NEMO tunnel will be smaller, to take into account the required headers.

```
(config vpn nemo nemo_example)> mtu integer
(config vpn nemo nemo_example)>
```

   Allowed values are any integer between 68 and 1476.

9. Set the Security Parameter Index (SPI) value, which is used in the authentication extension when registering. This should be normally left at the default setting of **256** unless your service provider indicates a different value.

```
(config vpn nemo nemo_example)> spi integer
(config vpn nemo nemo_example)>
```

   Allowed values are any integer between 256 and 4294967295.

10. Set the firewall zone for the NEMO tunnel to internal:

```
(config vpn nemo nemo_example)> zone internal
(config vpn nemo nemo_example)>
```

    The Internal firewall zone configures the IX15 device to trust traffic going to the tunnel and allows it through the network.

11. Configure the Care-of-Address, the local WAN interface of the internet facing network.
    a. Set the method to determine the Care-of-Address:

    ```
    (config vpn nemo nemo_example)> coaddress type value
    (config vpn nemo nemo_example)>
    ```

    where *value* is one of:

    - **defaultroute**: Uses the same network interface as the default route.
    - **interface**

      If **interface** is used, set the interface:

      i. Use the **?** to determine available interfaces:

      ii. Set the interface. For example:

      ```
      (config vpn nemo nemo_example)> coaddress interface eth1
      (config vpn nemo nemo_example)>
      ```

    - **ip**

      If **ip** is used, set the IP address:

      ```
      (config vpn nemo nemo_example)> coaddress address IP_address
      (config vpn nemo nemo_example)>
      ```

    The default is **defaultroute**.
12. Set the GRE tunnel local endpoint:
    a. Set the method to determine the GRE tunnel local endpoint:

    ```
    (config vpn nemo nemo_example)> tun_local type value
    (config vpn nemo nemo_example)>
    ```

    where *value* is one of:

    - **defaultroute**: Uses the same network interface as the default route.
    - **interface**

      If **interface** is used, set the interface.

      i. Use the **?** to determine available interfaces:

      ii. Set the interface. For example:

      ```
      (config vpn nemo nemo_example)> tun_local interface eth1
      (config vpn nemo nemo_example)>
      ```

    The default is **defaultroute**.
13. Configure one or more local networks to use as a virtual NEMO network interface. Generally, this will be a Local Area Network (LAN):
    a. Add a local network to use as a virtual NEMO network interface:

    ```
    (config vpn nemo nemo_example)> add network end eth
    (config vpn nemo nemo_example)>
    ```

    b. (Optional) Repeat for additional interfaces.

14. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

15. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show NEMO status

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, select **Status** > **NEMO**.

   The **NEMO** page appears.
3. To view configuration details about an NEMO tunnel, click the (configuration) icon in the upper right of the tunnel's status pane.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. To display details about all configured NEMO tunnels, type the following at the prompt:

```
> show nemo

 NEMO  Enable  Status  Address  Agent    CoAddress
 ----  ------  ------  -------  -------  ----------
 demo  false
 test  true    up      1.2.3.4  4.3.2.1  10.10.10.1

>
```

3. To display details about a specific tunnel:

```
> show nemo name test

 test NEMO Status
 ----------------
 Enabled                 : true
 Status                  : up
 Home Agent              : 4.3.2.1
 Care of Address         : 10.10.10.1
 Interface               : modem
 GRE Tunnel              : 10.10.10.1 === 4.3.2.1
 Metric                  : 255
```

```
MTU                       : 1476
Lifetime (Actual)         : 600

Local Network  Subnet           Status
-------------  ---------------  ----------
lan1           192.168.2.1/24   Advertized
LAN2           192.168.3.1/24   Advertized


>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Services

This chapter contains the following topics:

Allow remote access for web administration and SSH 423
Configure the web administration service 427
Configure SSH access 437
Use SSH with key authentication 445
Configure telnet access 448
Configure DNS 453
Simple Network Management Protocol (SNMP) 460
Location information 466
Modbus gateway 495
System time 512
Network Time Protocol 516
Configure a multicast route 523
Enable service discovery (mDNS) 526
Use the iPerf service 529
Configure the ping responder service 535

Allow remote access for web administration and SSH

By default, only devices connected to the IX15's LAN have access to the device via web administration and SSH. To enable these services for access from remote devices:

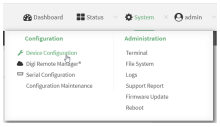
- The IX15 device must have a publicly reachable IP address.
- The **External** firewall zone must be added to the web administration or SSH service. See Firewall configuration for information on zones.
- See Set the idle timeout for IX15 users for information about setting the inactivity timeout for the web administration and SSH services.

To allow web administration or SSH for the External firewall zone:

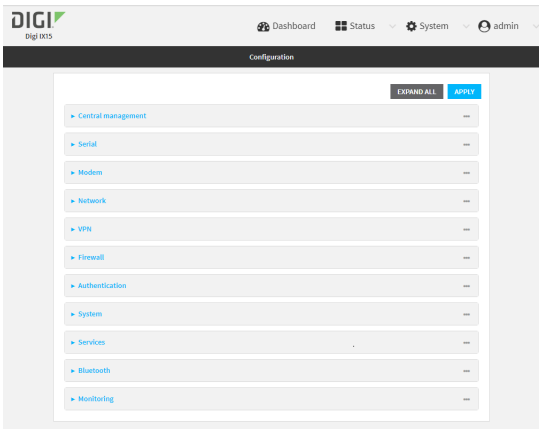
Add the External firewall zone to the web administration service

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

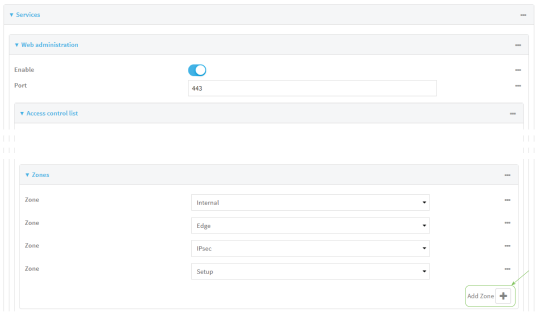


The **Configuration** window is displayed.

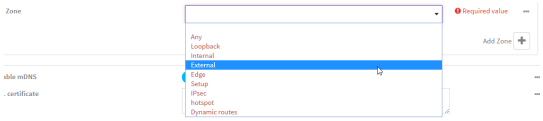


3. Click **Services** > **Web administration** > **Access Control List** > **Zones**.

4. For **Add Zone**, click ✚.



5. Select **External**.



6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add the external zone to the web administration service:

```
(config)> add service web_admin acl zone end external
(config)>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

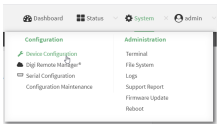
5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

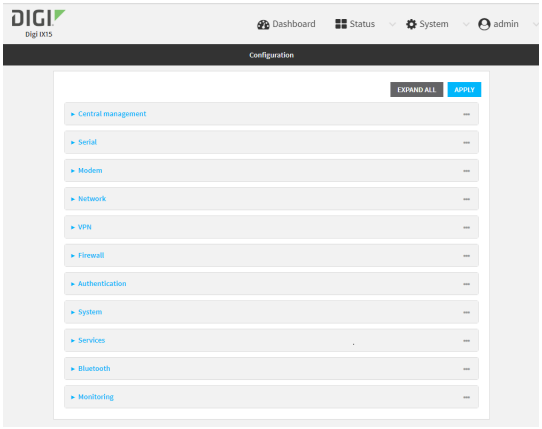
Add the External firewall zone to the SSH service

WebUI

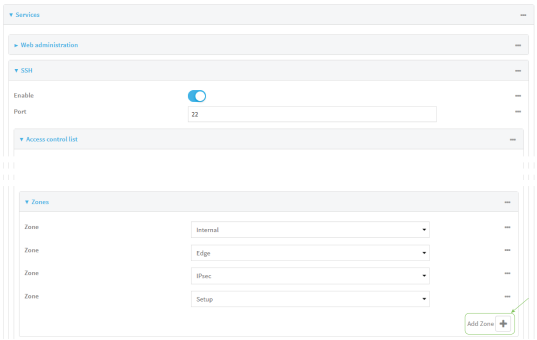
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



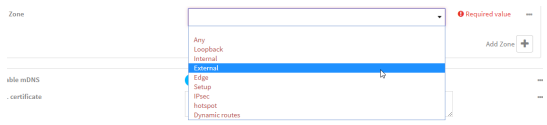
The **Configuration** window is displayed.



3. Click **Configuration** > **Services** > **SSH** > **Access Control List** > **Zones**.
4. For **Add Zone**, click ✚.



5. Select **External**.



6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add the **External** zone to the SSH service:

```
(config)> add service ssh acl zone end external
(config)>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure the web administration service

The web administration service allows you to monitor and configure the IX15 device by using the WebUI, a browser-based interface.

By default, the web administration service is enabled and uses the standard HTTPS port, 443. The default access control for the service uses the **Internal** firewall zone, which means that only devices connected to the IX15's LAN can access the WebUI. If this configuration is sufficient for your needs, no further configuration is required. See Allow remote access for web administration and SSH for information about configuring the web administration service to allow access from remote devices.

Required configuration items

- The web administration service is enabled by default.
- Configure access control for the service.

Additional configuration items

- Port to use for web administration service communication.
- Multicast DNS (mDNS) support.
- An SSL certificate to use for communications with the service.
- Support for legacy encryption protocols.

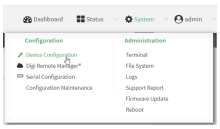
See Set the idle timeout for IX15 users for information about setting the inactivity timeout for the web administration services.

Enable or disable the web administration service

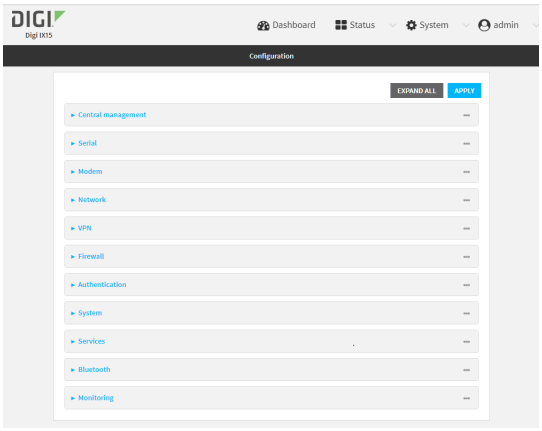
The web administration service is enabled by default. To disable the service, or enable it if it has been disabled:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **Web administration**.
4. Click **Enable**.
5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable or disable the web administration service:
   - To enable the service:

```
(config)> service web_admin enable true
(config)>
```

   - To disable the sevice:

```
(config)> service web_admin enable false
(config)>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

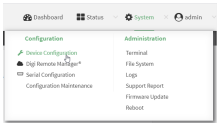
5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

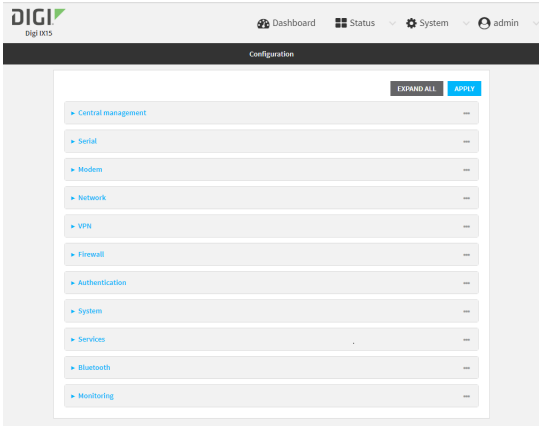
Configure the service

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **Services** > **Web administration**.
4. (Optional) For **Port**, enter the port number for the service. Normally this should not be changed.
5. Click **Access control list** to configure access control:

- To limit access to specified IPv4 addresses and networks:
  a. Click **IPv4 Addresses**.
  b. For **Add Address**, click ✚.
  c. For **Address**, enter the IPv4 address or network that can access the device's web administration service. Allowed values are:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 192.168.1.0/24.
     - **any**: No limit to IPv4 addresses that can access the web administration service.
  d. Click ✚ again to list additional IP addresses or networks.
- To limit access to specified IPv6 addresses and networks:
  a. Click **IPv6 Addresses**.
  b. For **Add Address**, click ✚.
  c. For **Address**, enter the IPv6 address or network that can access the device's web administration service. Allowed values are:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 2001:db8::/48.
     - **any**: No limit to IPv6 addresses that can access the web administration service.
  d. Click ✚ again to list additional IP addresses or networks.
- To limit access to hosts connected through a specified interface on the IX15 device:
  a. Click **Interfaces**.
  b. For **Add Interface**, click ✚.
  c. For **Interface**, select the appropriate interface from the dropdown.
  d. Click ✚ again to allow access through additional interfaces.
- To limit access based on firewall zones:
  a. Click **Zones**.
  b. For **Add Zone**, click ✚.
  c. For **Zone**, select the appropriate firewall zone from the dropdown.
     See Firewall configuration for information about firewall zones.
  d. Click ✚ again to allow access through additional firewall zones.

6. Multicast DNS (mDNS) is enabled by default. mDNS is a protocol that resolves host names in small networks that do not have a DNS server. To disable mDNS, or enable it if it has been disabled, click **Enable mDNS**.
7. For **SSL certificate**, if you have your own signed SSL certificate, paste the certificate and private key. If **SSL certificate** is blank, the device will use an automatically-generated, self-signed certificate.
   - The SSL certificate and private key must be in PEM format.
   - The private key can use one of the following algorithms:
     - RSA
     - DSA
     - ECDSA

- ECDH

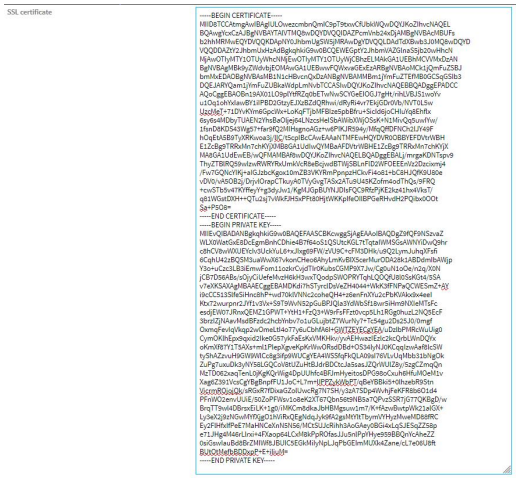
Note Password-protected certificate keys are not supported.

Example:

a. Generate the SSL certificate and private key, for example:

```
# openssl req -newkey rsa:2048 -nodes -keyout key.pem -x509 -days 365
-out certificate.pem
```

b. Paste the contents of **certificate.pem** and **key.pem** into the **SSL certificate** field. The contents of the **certificate.pem** must be first. For example:



8. For **Allow legacy encryption protocols**, enable this option to allow clients to connect to the HTTPS session by using encryption protocols older than TLS 1.2, in addition to TLS 1.2 and later protocols. This option is disabled by default, which means that only TLS 1.2 and later encryption protocols are allowed with HTTPS connections.
9. **View** is set to **Auto** by default and normally should not be changed.
10. **Legacy port redirection** is used to redirect client HTTP requests to the HTTPS service. Legacy port redirection is enabled by default, and normally these settings should not be changed. To disable legacy port redirection, click to expand **Legacy port redirection** and deselect **Enable**.
11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Configure access control:
   - To limit access to specified IPv4 addresses and networks:

     ```
     (config)> add service web_admin acl address end value
     (config)>
     ```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 192.168.1.0/24.
     - **any**: No limit to IPv4 addresses that can access the web administratrion service.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:

     ```
     (config)> add service web_admin acl address6 end value
     (config)>
     ```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 2001:db8::/48.
     - **any**: No limit to IPv6 addresses that can access the web administratrion service.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to hosts connected through a specified interface on the IX15 device:

     ```
     (config)> add service web_admin acl interface end value
     (config)>
     ```

     Where *value* is an interface defined on your device.

     Display a list of available interfaces:

     Use **... network interface ?** to display interface information:

     ```
     (config)> ... network interface ?

     Interfaces

     Additional Configuration
     ------------------------------------------
      defaultip              Default IP
      defaultlinklocal       Default Link-local IP
      eth                    ETH
      loopback               Loopback
      modem                  Modem

     config)>
     ```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config)> add service web_admin acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.

 Additional Configuration
 ------------------------------------------------------------
-----------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

(config)>
```

Repeat this step to list additional firewall zones.

4. (Optional) If you have your own signed SSL certificate, if you have your own signed SSL certificate, set the certificate and private key by pasting their contents into the **service web_admin cert** command. Enclose the certificate and private key contents in quotes (**"**).

```
(config)> service web_admin cert "ssl-cert-and-private-key"
(config)>
```

- If **SSL certificate** is blank, the device will use an automatically-generated, self-signed certificate.
- The SSL certificate and private key must be in PEM format.
- The private key can use one of the following algorithms:
  - RSA
  - DSA
  - ECDSA
  - ECDH

Note Password-protected certificate keys are not supported.

Example

a. Generate the SSL certificate and private key, for example:

```
# openssl req -newkey rsa:2048 -nodes -keyout key.pem -x509 -days 365
-out certificate.pem
```

b. Paste the contents of **certificate.pem** and **key.pem** into the **service web_admin cert** command. Enclose the contents of **certificate.pem** and **key.pem** in quotes. For example:

```
(config)> service web_admin cert "-----BEGIN CERTIFICATE-----
MIID8TCCAtmgAwIBAgIULOwezcmbnQmIC9pT9txwCfUbkWQwDQYJKoZIhvcNAQEL
BQAwgYcxCzAJBgNVBAYTAlVTMQ8wDQYDVQQIDAZPcmVnb24xDjAMBgNVBAcMBUFs
b2hhMRMwEQYDVQQKDApNY0JhbmUgSW5jMRAwDgYDVQQLDAdTdXBwb3J0MQ8wDQYD
VQQDDAZtY2JhbmUxHzAdBgkqhkiG9w0BCQEWEGptY2JhbmVAZGlnaS5jb20wHhcN
MjAwOTIyMTY1OTUyWhcNMjEwOTIyMTY1OTUyWjCBhzELMAkGA1UEBhMCVVMxDzAN
BgNVBAgMBk9yZWdvbjEOMAwGA1UEBwwFQWxvaGExEzARBgNVBAoMCk1jQmFuZSBJ
bmMxEDAOBgNVBAsMB1N1cHBvcnQxDzANBgNVBAMMBm1jYmFuZTEfMB0GCSqGSIb3
DQEJARYQam1jYmFuZUBkaWdpLmNvbTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCC
AQoCggEBAOBn19AX01LO9plYtfRZq0bETwNwSCYGeEIOGJ7gHt/rihLVBJS1woYv
u1Oq1ohYxIawBY1iIPBD2GtzyEJXzBZdQRhwi/dRyRi4vr7EkjGDr0Vb/NVT0L5w
UzcMeT+71DYvKYm6GpcWx+LoKqFTjbMFBIze5pbBfru+SicId6joCHIuYq8Ehflx
6sy6s4MDbyTUAEN2YhsBaOljej64LNzcsHeISbAWibXWjOSsK+N1MivQq5uwIYw/
1fsnD8KDS43Wg57+far9fQ2MIHsgnoAGz+w6PIKJR594y/MfqQffDFNCh2lJY49F
hOqEtA5B9TyXRKwoa3j/lIC/t5cpIBcCAwEAAaNTMFEwHQYDVR0OBBYEFDVtrWBH
E1ZcBg9TRRxMn7chKYjXMB8GA1UdIwQYMBaAFDVtrWBHE1ZcBg9TRRxMn7chKYjX
MA8GA1UdEwEB/wQFMAMBAf8wDQYJKoZIhvcNAQELBQADggEBALj/mrgaKDNTspv9
ThyZTBlRQ59wIzwRWRYRxUmkVcR8eBcjwdBTWjSBLnFlD2WFOEEEnVz2Dzcixmj4
/Fw7GQNcYIKj+aIGJzbcKgox10mZB3VKYRmPpnpzHCkvFi4o81+bC8HJQfK9U80e
vDV0/vA5OB2j/DrjvlOrapCTkuyA0TVyGvgTASx2ATu9U45KZofm4odThQs/9FRQ
+cwSTb5v47KYffeyY+g3dyJw1/KgMJGpBUYNJDIsFQC9RfzPjKE2kz41hx4VksT/
q81WGstDXH++QTu2sj7vWkFJH5xPFt80HjtWKKpIfeOIlBPGeRHvdH2PQibx0OOt
Sa+P5O8=
-----END CERTIFICATE-----
-----BEGIN PRIVATE KEY-----
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBKcwggSjAgEAAoIBAQDgZ9fQF9NSzvaZ
WLX0WatGxE8DcEgmBnhCDhie4B7f64oS1QSUtcKGL7tTqtaIWMSGsAWNYiDwQ9hr
c8hCV8wWXUEYcIv3UckYuL6+xJIxg69FW/zVU9C+cFM3DHk/u9Q2LymJuhqXFsfi
6CqhU42zBQSM3uaWwX67vkonCHeo6AhyLmKvBIX5cerMurODA28k1ABDdmIbAWjp
Y3o+uCzc3LB3iEmwFom11ozkrCvjdTIr0KubsCGMP9X7Jw/Cg0uN1oOe/n2q/X0N
jCB7D56ABs/sOjyCiUefeMvzH6kH3wxTQodpSWOPRYTqhLQOQfU8l0SsKGt4/5SA
v7eXKSAXAgMBAAECggEBAMDKdi7hSTyrclDsVeZH4044+WkK3fFNPaQCWESmZ+AY
i9cCC513SlfeSiHnc8hP+wd70klVNNc2coheQH4+z6enFnXYu2cPbKVAkx9x4eeI
Ktx72wurpnr2JYf1v3Vx+S9T9WvN52pGuBPJQla3YdWbSf18wr5iHm9NXIeMTsFc
esdjEW07JRnxQEMZ1GPWT+YtH1+FzQ3+W9rFsFFzt0vcp5Lh1RGg0huzL2NQ5EcF
3brzIZjNAavMsdBFzdc2hcbYnbv7o1uGLujbtZ7WurNy7+Tc54gu2Ds25J0/0mgf
OxmqFevIqVkqp2wOmeLtI4o77y6uCbhfA6I+GWTZEYECgYEA/uDzlbPMRcWuUig0
CymOKlhEpx9qxid2Ike0G57ykFaEsKxVMKHkv/yvAEHwazIEzlc2kcQrbLWnDQYx
oKmXf87Y1T5AXs+ml1PlepXgveKpKrWwORsdDBd+OS34lyNJ0KCqqIzwAaf8lcSW
tyShAZzvuH9GW9WlCc8g3ifp9WUCgYEA4WSSfqFkQLA09sI76VLvUqMbb31bNgOk
ZuPg7uxuDk3yNY58LGQCoV8tUZuHtBJdrBDCtcJa5sasJZQrWUlZ8y/5zgCZmqQn
MzTD062xaqTenL0jKgKQrWig4DpUUhfc4BFJmHyeitosDPG98oCxuh6HfuMOeM1v
```

```
Xag6Z391VcsCgYBgBnpfFU1JoC+L7m+lIPPZykWbPT/qBeYBBki5+0lhzebR9Stn
VicrmROjojQk/sRGxR7fDixaGZolUwcRg7N7SH/y3zA7SDp4WvhjFeKFR8b6O1d4
PFnWO2envUUiE/50ZoPFWsv1o8eK2XT67Qbn56t9NB5a7QPvzSSR7jG77QKBgD/w
BrqTT9wl4DBrsxEiLK+1g0/iMKCm8dkaJbHBMgsuw1m7/K+fAzwBwtpWk21alGX+
Ly3eX2j9zNGwMYfXjgO1hViRxQEgNdqJyk9fA2gsMtYltTbymVYHyzMweMD88fRC
Ey2FlHfxIfPeE7MaHNCeXnN5N56/MCtSUJcRihh3AoGAey0BGi4xLqSJESqZZ58p
e71JHg4M46rLlrxi+4FXaop64LCxM8kPpROfasJJu5nlPpYHye959BBQnYcAheZZ
0siGswIauBd8BrZMIWf8JBUIC5EGkMiIyNpLJqPbGEImMUXk4Zane/cL7e06U8ft
BUtOtMefbBDDxpP+E+iIiuM=
-----END PRIVATE KEY-----"
(config)>
```

5. (Optional) Configure Multicast DNS (mDNS):

   mDNS is a protocol that resolves host names in small networks that do not have a DNS server. mDNS is enabled by default. To disable mDNS, or enable it if it has been disabled:

   - To enable the mDNS protocol:

     ```
     (config)> service web_admin mdns enable true
     (config>
     ```

   - To disable the mDNS protocl:

     ```
     (config)> service web_admin mdns enable false
     (config)>
     ```

6. (Optional) Set the port number for this service.

   The default setting of 443 normally should not be changed.

   ```
   (config)> service web_admin port 444
   (config)>
   ```

7. (Optional) Configure the device to allow legacy encryption protocols.

   Legacy encryption protocols allow clients to connect to the HTTPS session by using encryption protocols older than TLS 1.2, in addition to TLS 1.2 and later protocols. This option is disabled by default, which means that only TLS 1.2 and later encryption protocols are allowed with HTTPS connections.

   To enable legacy encryption protocols:

   ```
   (config)> service web_admin legacy_encryption true
   (config)>
   ```

8. (Optional) Disable legacy port redirection.

   Legacy port redirection is used to redirect client HTTP requests to the HTTPS service. Legacy port redirection is enabled by default, and normally these settings should not be changed.

   To disable legacy port redirection:

   ```
   (config)> service web_admin legacy enable false
   (config)>
   ```

9. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

10. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure SSH access

The IX15's default configuration has SSH access enabled, and allows SSH access to the device from authorized users within the **Internal** firewall zone. If this configuration is sufficient for your needs, no further configuration is required. See Allow remote access for web administration and SSH for information about configuring the SSH service to allow access from remote devices.

Required configuration items

- Enable SSH access.
- Configure access control for the SSH service.

Additional configuration items

- Port to use for communications with the SSH service.
- Multicast DNS (mDNS) support.
- A private key to use for communications with the SSH service.
- Create custom SSH configuration settings.

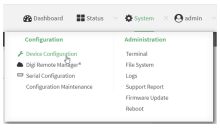
See Set the idle timeout for IX15 users for information about setting the inactivity timeout for the SSH service.

Enable or disable the SSH service

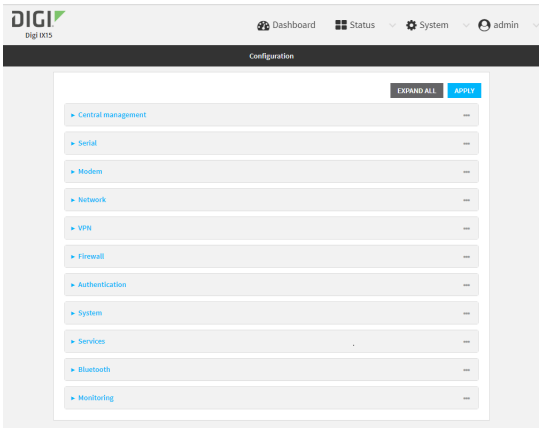
The SSH service is enabled by default. To disable the service, or enable it if it has been disabled:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **SSH**.
4. Click **Enable**.
5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable or disable the SSH service:
   - To enable the service:

```
(config)> service ssh enable true
(config)>
```

   - To disable the sevice:

```
(config)> service ssh enable false
(config)>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

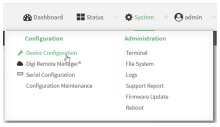
5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

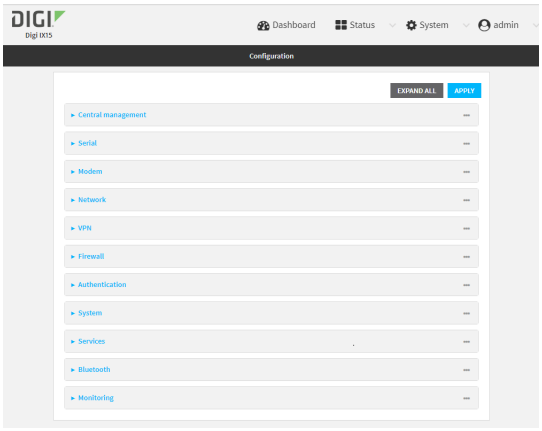
Configure the service

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **Services** > **SSH**.
4. (Optional) For **Port**, enter the port number for the service. Normally this should not be changed.
5. Click **Access control list** to configure access control:
   - To limit access to specified IPv4 addresses and networks:
     a. Click **IPv4 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv4 address or network that can access the device's SSH service. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 192.168.1.0/24.
        - **any**: No limit to IPv4 addresses that can access the SSH service.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:
     a. Click **IPv6 Addresses**.
     b. For **Add Address**, click ✚.

c. For **Address**, enter the IPv6 address or network that can access the device's SSH service. Allowed values are:
   - A single IP address or host name.
   - A network designation in CIDR notation, for example, 2001:db8::/48.
   - **any**: No limit to IPv6 addresses that can access the SSH service.

d. Click ✚ again to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX15 device:
  a. Click **Interfaces**.
  b. For **Add Interface**, click ✚.
  c. For **Interface**, select the appropriate interface from the dropdown.
  d. Click ✚ again to allow access through additional interfaces.
- To limit access based on firewall zones:
  a. Click **Zones**.
  b. For **Add Zone**, click ✚.
  c. For **Zone**, select the appropriate firewall zone from the dropdown.
     See Firewall configuration for information about firewall zones.
  d. Click ✚ again to allow access through additional firewall zones.

6. Multicast DNS (mDNS) is enabled by default. mDNS is a protocol that resolves host names in small networks that do not have a DNS server. To disable mDNS, or enable it if it has been disabled, click **Enable mDNS**.
7. For **Private key**, type the private key in PEM format. If **Private key** is blank, the device will use an automatically-generated key.
8. To create custom SSH configuration settings:
   a. Click to expand **Custom configuration**.
   b. Click **Enable**.
   c. For **Override**:
      - If **Override** is enabled, entries in **Configuration file** will be used in place of the standard SSH configuration.
      - If **Override** is not enabled, entries in **Configuration file** will be added to the standard SSH configuration.
   d. For **Configuration file**, type configuration settings in the form of an OpenSSH sshd_config file.

      For example, to enable the diffie-helman-group-sha-14 key exchange algorithm:
      i. Click **Enable** to enable SSH custom configuration.
      ii. Leave **Override** disabled.
      iii. For **Configuration file**, type the following:

```
KexAlgorithms +diffie-hellman-group14-sha1
```

9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Configure access control:
   - To limit access to specified IPv4 addresses and networks:

```
(config)> add service ssh acl address end value
(config)>
```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 192.168.1.0/24.
     - **any**: No limit to IPv4 addresses that can access the SSH service.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:

```
(config)> add service ssh acl address6 end value
(config)>
```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 2001:db8::/48.
     - **any**: No limit to IPv6 addresses that can access the SSH service.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to hosts connected through a specified interface on the IX15 device:

```
(config)> add service ssh acl interface end value
(config)>
```

     Where *value* is an interface defined on your device.

     Display a list of available interfaces:

     Use **... network interface ?** to display interface information:

```
(config)> ... network interface ?
```

```
Interfaces

Additional Configuration
-----------------------------------------
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config)> add service ssh acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.

 Additional Configuration
 ------------------------------------------------------------
-----------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

(config)>
```

Repeat this step to list additional firewall zones.

4. (Optional) Set the private key in PEM format. If not set, the device will use an automatically-generated key.

```
(config)> service ssh key key.pem
(config)>
```

5. (Optional) Configure Multicast DNS (mDNS)

mDNS is a protocol that resolves host names in small networks that do not have a DNS server. mDNS is enabled by default. To disable mDNS, or enable it if it has been disabled:

- To enable the mDNS protocol:

```
(config)> service ssh mdns enable true
(config>
```

- To disable the mDNS protocl:

```
(config)> service ssh mdns enable false
(config)>
```

6. (Optional) Set the port number for this service.

   The default setting of 22 normally should not be changed.

```
(config)> service ssh port 24
(config)>
```

7. To create custom SSH configuration settings:
   a. Enable custom configurations:

```
(config)> service ssh custom enable true
(config)>
```

   b. To override the standard SSH configuration and only use the **config_file** parameter:

```
(config)> service ssh custom override true
(config)>
```

      - If **override** is set to **true**, entries in **Configuration file** will be used in place of the standard SSH configuration.
      - If **override** is set to **false**, entries in **Configuration file** will be added to the standard SSH configuration.

        The default is **false**.

   c. Set the configuration settings:

```
(config)> service ssh custom config_file value
(config)>
```

      where *value* is one or more entires in the form of an OpenSSH sshd_config file. For example, to enable the diffie-helman-group-sha-14 key exchange algorithm:

```
(config)> service ssh custom config_file "KexAlgorithms +diffie-
hellman-group14-sha1"
(config)>
```

8. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Use SSH with key authentication

Rather than using passwords, you can use SSH keys to authenticate users connecting via SSH, SFTP, or SCP. SSH keys provide security and scalability:

- **Security**: Using SSH keys for authentication is more secure than using passwords. Unlike a password that can be guessed by an unauthorized user, SSH key pairs provide more sophisticated security. A public key configured on the Digi IX15 Gateway device is paired with a private key on the user's PC. The private key, once generated, remains on the user's PC.
- **Scalability**: SSH keys can be used on more than one Digi IX15 Gateway device.

Generating SSH key pairs

On a Microsoft Windows PC, you can generate SSH key pairs using a terminal emulator application, such as **PuTTY** or **Tera Term**.

On a Linux host, an SSH key pair is usually created automatically in the user's **.ssh** directory. The private and public keys are named **id_rsa** and **id_rsa.pub**. If you need to generate an SSH key pair, you can use the **ssh-keygen** application.

For example, the following entry generates an RSA key pair in the user's **.ssh** directory:

```
ssh-keygen -t rsa -f ~/.ssh/id_rsa
```

The private key file is named **id_rsa** and the public key file is named **id_rsa.pub**. (The **.pub** extension is automatically appended to the name specified for the private key output file.)

Required configuration items

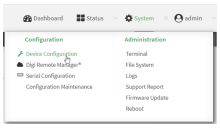
- Name for the user
- SSH public key for the user

Additional configuration items

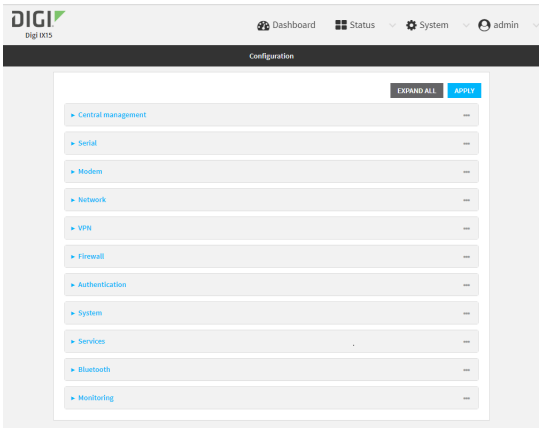
- If you want to access the Digi IX15 Gateway device using SSH over a WAN interface, configure the access control list for the SSH service to allow SSH access for the **External** firewall zone.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Authentication** > **Users**.
4. Select an existing user or create a new user. See User authentication for information about creating a new user.
5. Click **SSH keys**.
6. In **Add SSH key**, enter a name for the SSH key and click ✚.
7. Enter the public SSH key by pasting or typing a public encryption key that this user can use for passwordless SSH login.
8. Click **Apply** to save the configuration and apply the change.



Command line

You can add configure passwordless SSH login for an existing user or include the support when creating a new user. See User authentication for information about creating a new user. These instructions assume an existing user named **temp_user**.

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add an SSH key for the user by using the ssh_key command and pasting or typing a public encryption key:

```
(config)> add auth user maria ssh_key key_name key
(config)>
```

   where:

- *key_name* is a name for the key.
- *key* is a public SSH key, which you can enter by pasting or typing a public encryption key that this user can use for passwordless SSH login

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure telnet access

By default, the telnet service is disabled.

Note Telnet is an insecure protocol and should only be used for backward-compatibility reasons, and only if the network connection is otherwise secured.

Required configuration items

- Enable telnet access.
- Configure access control for the telnet service.

Additional configuration items

- Port to use for communications with the telnet service.
- Multicast DNS (mDNS) support.

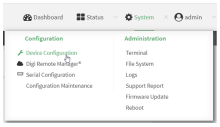
See Set the idle timeout for IX15 users for information about setting the inactivity timeout for the telnet service.

Enable the telnet service

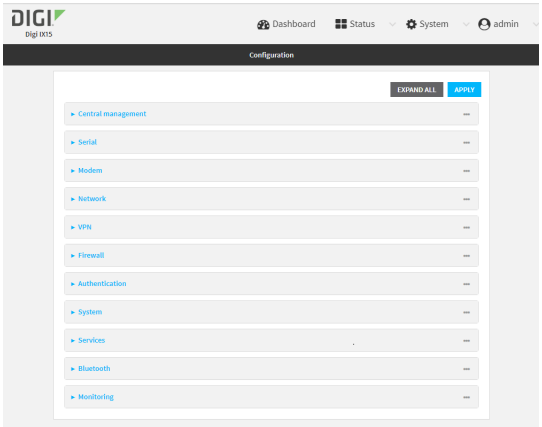
The telnet service is disabled by default. To enable the service:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **telnet**.

4. Click **Enable**.
5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable the telnet service:

```
(config)> service telnet enable true
(config)>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

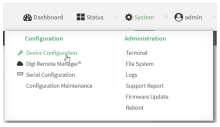
5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

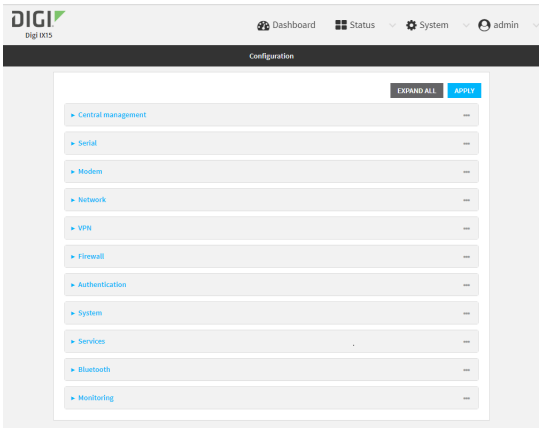
Configure the service

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **Services** > **telnet**.
4. (Optional) For **Port**, enter the port number for the service. Normally this should not be changed.
5. Click **Access control list** to configure access control:
   - To limit access to specified IPv4 addresses and networks:
     a. Click **IPv4 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv4 address or network that can access the device's telnet service. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 192.168.1.0/24.
        - **any**: No limit to IPv4 addresses that can access the telnet service.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:
     a. Click **IPv6 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv6 address or network that can access the device's telnet service. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 2001:db8::/48.
        - **any**: No limit to IPv6 addresses that can access the telnet service.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to hosts connected through a specified interface on the IX15 device:
     a. Click **Interfaces**.
     b. For **Add Interface**, click ✚.
     c. For **Interface**, select the appropriate interface from the dropdown.
     d. Click ✚ again to allow access through additional interfaces.

- To limit access based on firewall zones:
  a. Click **Zones**.
  b. For **Add Zone**, click ✚.
  c. For **Zone**, select the appropriate firewall zone from the dropdown.
     See Firewall configuration for information about firewall zones.
  d. Click ✚ again to allow access through additional firewall zones.

6. Multicast DNS (mDNS) is disabled by default. mDNS is a protocol that resolves host names in small networks that do not have a DNS server. To enable mDNS, click **Enable mDNS**.
7. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Configure access control:
   - To limit access to specified IPv4 addresses and networks:

```
(config)> add service telnet acl address end value
(config)>
```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 192.168.1.0/24.
     - **any**: No limit to IPv4 addresses that can access the telnet service.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:

```
(config)> add service telnet acl address6 end value
(config)>
```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 2001:db8::/48.
     - **any**: No limit to IPv6 addresses that can access the telnet service.

     Repeat this step to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX15 device:

```
(config)> add service telnet acl interface end value
(config)>
```

Where *value* is an interface defined on your device.

Display a list of available interfaces:

Use **... network interface ?** to display interface information:

```
(config)> ... network interface ?

Interfaces

Additional Configuration
-----------------------------------------
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config)> add service telnet acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.

 Additional Configuration
 ------------------------------------------------------------
-----------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup
```

```
(config)>
```

Repeat this step to list additional firewall zones.

4. (Optional) Configure Multicast DNS (mDNS)

   mDNS is a protocol that resolves host names in small networks that do not have a DNS server. mDNS is disabled by default. To enable:

```
(config)> service telnet mdns enable true
(config>
```

5. (Optional) Set the port number for this service.

   The default setting of 23 normally should not be changed.

```
(config)> service telnet port 25
(config)>
```

6. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure DNS

The IX15 device includes a caching DNS server which forwards queries to the DNS servers that are associated with the network interfaces, and caches the results. This server is used within the device, and cannot be disabled. Use the access control list to restrict external access to this server.

Required configuration items

- Configure access control for the DNS service.

Additional configuration items

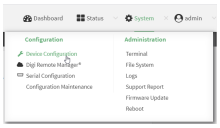
- Whether the device should cache negative responses.
- Whether the device should always perform DNS queries to all available DNS servers.
- Whether to prevent upstream DNS servers from returning private IP addresses.
- Additional DNS servers, in addition to the ones associated with the device's network interfaces.
- Specific host names and their IP addresses.

  The device is configured by default with the hostname **digi.device**, which corresponds to the **192.168.210.1** IP address.

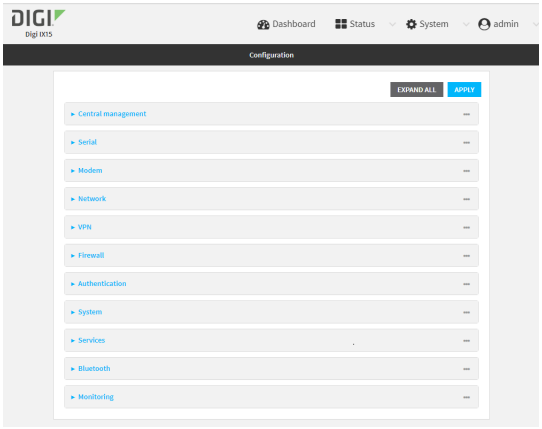
To configure the DNS server:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **DNS**.
4. Click **Access control list** to configure access control:
   - To limit access to specified IPv4 addresses and networks:
     a. Click **IPv4 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv4 address or network that can access the device's DNS service. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 192.168.1.0/24.
        - **any**: No limit to IPv4 addresses that can access the DNS service.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:
     a. Click **IPv6 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv6 address or network that can access the device's DNS service. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 2001:db8::/48.
        - **any**: No limit to IPv6 addresses that can access the DNS service.
     d. Click ✚ again to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX15 device:
  a. Click **Interfaces**.
  b. For **Add Interface**, click ✚.
  c. For **Interface**, select the appropriate interface from the dropdown.
  d. Click ✚ again to allow access through additional interfaces.
- To limit access based on firewall zones:
  a. Click **Zones**.
  b. For **Add Zone**, click ✚.
  c. For **Zone**, select the appropriate firewall zone from the dropdown.
  See Firewall configuration for information about firewall zones.
  d. Click ✚ again to allow access through additional firewall zones.

5. (Optional) **Cache negative responses** is enabled by default. Disabling this option may improve performance on networks with transient DNS results, when one or more DNS servers may have positive results. To disable, click **Cache negative responses**.
6. (Optional) **Query all servers** is enabled by default. This option is useful when only some DNS servers will be able to resolve hostnames. To disable, click **Query all servers**.
7. (Optional) **Rebind protection**, if enabled, prevents upstream DNS servers from returning private IP addresses. To enable, click **Rebind protection**.
8. (Optional) **Allow localhost rebinding** is enabled by default if **Rebind protection** is enabled. This is useful for Real-time Black List (RBL) servers.
9. (Optional) To add additional DNS servers:
   a. Click **DNS servers**.
   b. For **Add Server**, click ✚.
   c. (Optional) Enter a label for the DNS server.
   d. For **DNS server**, enter the IP address of the DNS server.
   e. **Domain** restricts the device's use of this DNS server based on the domain. If no domain are listed, then all queries may be sent to this server.
10. (Optional) To add host names and their IP addresses that the device's DNS server will resolve:
    a. Click **Additional DNS hostnames**.
    b. For **Add Host**, click ✚.
    c. Type the **IP address** of the host.
    d. For **Name**, type the hostname.
11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Configure access control:
   - To limit access to specified IPv4 addresses and networks:

```
(config)> add service dns acl address end value
(config)>
```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 192.168.1.0/24.
     - **any**: No limit to IPv4 addresses that can access the DNS service.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:

```
(config)> add service dns acl address6 end value
(config)>
```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 2001:db8::/48.
     - **any**: No limit to IPv6 addresses that can access the DNS service.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to hosts connected through a specified interface on the IX15 device:

```
(config)> add service dns acl interface end value
(config)>
```

     Where *value* is an interface defined on your device.

     Display a list of available interfaces:

     Use **... network interface ?** to display interface information:

```
(config)> ... network interface ?

Interfaces

Additional Configuration
------------------------------------------
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
```

```
 loopback                Loopback
 modem                   Modem

config)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config)> add service dns acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.

 Additional Configuration
 ------------------------------------------------------------
-----------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

(config)>
```

Repeat this step to list additional firewall zones.

4. (Optional) Cache negative responses

   By default, the device's DNS server caches negative responses. Disabling this option may improve performance on networks with transient DNS results, when one or more DNS servers may have positive results. To disable:

```
(config)> service dns cache_negative_responses false
(config>
```

5. (Optional) Query all servers

   By default, the device's DNS server queries all available DNS servers. Disabling this option may improve performance on networks with transient DNS results, when one or more DNS servers may have positive results. To disable:

```
(config)> service dns query_all_servers false
(config>
```

6. (Optional) Rebind protection

   By default, rebind protection is disabled. If enabled, this prevents upstream DNS servers from returning private IP addresses. To enable:

```
(config)> service dns stop_dns_rebind false
(config)>
```

7. (Optional) Allow localhost rebinding

   By default, localhost rebinding is enabled by default if rebind protection is enabled. This is useful for Real-time Black List (RBL) servers. To disable:

```
(config)> service dns rebind_localhost_ok false
(config)>
```

8. (Optional) Add additional DNS servers
   a. Add a DNS server:

```
(config)> add service dns server end
(config service dns server 0)>
```

   b. Set the IP address of the DNS server:

```
(config service dns server 0)> address ip-addr
(config service dns server 0)>
```

   c. To restrict the device's use of this DNS server based on the domain, use the **domain** command. If no domain are listed, then all queries may be sent to this server.

```
(config service dns server 0)> domain domain
(config service dns server 0)>
```

   d. (Optional) Set a label for this DNS server:

```
(config service dns server 0)> label label
(config service dns server 0)>
```

9. (Optional) Add host names and their IP addresses that the device's DNS server will resolve
   a. Add a host:

```
(config)> add service dns host end
(config service dns host 0)>
```

   b. Set the IP address of the host:

```
(config service dns host 0)> address ip-addr
(config service dns host 0)>
```

c. Set the host name:

```
(config service dns host 0)> name host-name
(config service dns host 0)>
```

10. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

11. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show DNS server

You can display status for DNS servers. This command is available only at the Admin CLI.

Command line

Show DNS information

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Use the show dns command at the system prompt:

```
> show dns

 Interface  Label  Server                    Domain
 ---------  -----  ------------------------  ------
 eth1              192.168.3.1
 eth1              fd00:2704::1
 eth1              fe80::227:4ff:fe2b:ae12
 eth1              fe80::227:4ff:fe44:105b
 eth1              fe80::240:ffff:fe80:23b0

>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Simple Network Management Protocol (SNMP)

Simple Network Management Protocol (SNMP) is a protocol for remotely managing and monitoring network devices. Network administrators can use the SNMP architecture to manage nodes, including servers, workstations, routers, switches, hubs, and other equipment on an IP network, manage network performance, find and solve network problems, and plan for network growth.

The IX15 device supports both SNMPv3 and SNMPv2c in read-only mode. Both are disabled by default. SNMPv1 is not supported.

SNMP Security

By default, the IX15 device automatically blocks SNMP packets from being received over WAN and LAN interfaces. As a result, if you want a IX15 device to receive SNMP packets, you must configure the SNMP access control list to allow the device to receive the packets. See Configure Simple Network Management Protocol (SNMP).

Configure Simple Network Management Protocol (SNMP)

Required configuration items

- Enable SNMP.
- Firewall configuration using access control to allow remote connections to the SNMP agent.
- The user name and password used to connect to the SNMP agent.

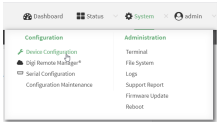
Additional configuration items

- The port used by the SNMP agent.
- Authentication type (either MD5 or SHA).
- Privacy protocol (either DES or AES).
- Privacy passphrase, if different that the SNMP user password.
- Enable Multicast DNS (mDNS) support.

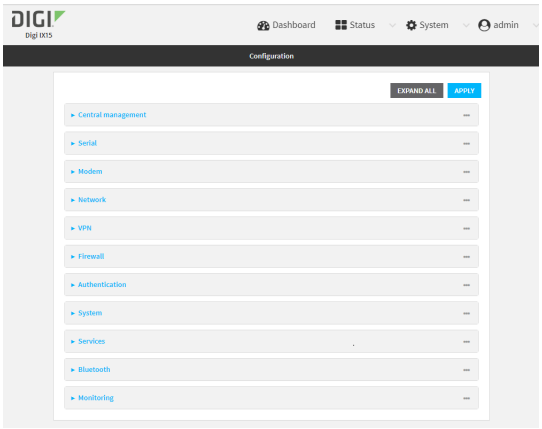
To configure the SNMP agent on your IX15 device:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **SNMP**.
4. Click **Enable**.
5. Click **Access control list** to configure access control:
   - To limit access to specified IPv4 addresses and networks:
     a. Click **IPv4 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv4 address or network that can access the device's SNMP agent. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 192.168.1.0/24.
        - **any**: No limit to IPv4 addresses that can access the SNMP agent.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:
     a. Click **IPv6 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv6 address or network that can access the device's SNMP agent. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 2001:db8::/48.
        - **any**: No limit to IPv6 addresses that can access the SNMP agent.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to hosts connected through a specified interface on the IX15 device:
     a. Click **Interfaces**.
     b. For **Add Interface**, click ✚.
     c. For **Interface**, select the appropriate interface from the dropdown.
     d. Click ✚ again to allow access through additional interfaces.
   - To limit access based on firewall zones:
     a. Click **Zones**.
     b. For **Add Zone**, click ✚.

c. For **Zone**, select the appropriate firewall zone from the dropdown.
   See Firewall configuration for information about firewall zones.
d. Click ✚ again to allow access through additional firewall zones.

6. Type the **Username** used to connect to the SNMP agent.
7. Type the **Password** used to connect to the SNMP agent.
8. (Optional) For **Port**, type the port number. The default is **161**.
9. (Optional) Multicast DNS (mDNS) is disabled by default. mDNS is a protocol that resolves host names in small networks that do not have a DNS server. To enable mDNS, click **Enable mDNS**.
10. (Optional) Select the **Authentication type**, either **MD5** or **SHA**. The default is **MD5**.
11. (Optional) Type the **Privacy passphrase**. If not set, the password, entered above, is used.
12. (Optional) Select the **Privacy protocol**, either **DES** or **AES**. The default is **DES**.
13. (Optional) Click **Enable version 2c access** to enable read-only access to SNMP version 2c.
14. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable the SNMP agent:

```
(config)> service snmp enable true
(config)>
```

4. Configure access control:
   - To limit access to specified IPv4 addresses and networks:

```
(config)> add service snmp acl address end value
(config)>
```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 192.168.1.0/24.
     - **any**: No limit to IPv4 addresses that can access the SNMP service.

     Repeat this step to list additional IP addresses or networks.

- To limit access to specified IPv6 addresses and networks:

```
(config)> add service snmp acl address6 end value
(config)>
```

Where *value* can be:

- A single IP address or host name.
- A network designation in CIDR notation, for example, 2001:db8::/48.
- **any**: No limit to IPv6 addresses that can access the SNMP service.

Repeat this step to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX15 device:

```
(config)> add service snmp acl interface end value
(config)>
```

Where *value* is an interface defined on your device.

Display a list of available interfaces:

Use **... network interface ?** to display interface information:

```
(config)> ... network interface ?

Interfaces

Additional Configuration
-----------------------------------------
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config)> add service snmp acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.
```

```
 Additional Configuration
 ------------------------------------------------------------
-----------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

(config)>
```

Repeat this step to list additional firewall zones.

5. Set the name of the user that will be used to connect to the SNMP agent.

```
(config)> service snmp username name
(config)>
```

6. Set the password for the user that will be used to connect to the SNMP agent:

```
(config)> service snmp password pwd
(config)>
```

7. (Optional) Set the port number for the SNMP agent. The default is **161**.

```
(config)> service snmp port port
(config)>
```

8. (Optional) Configure Multicast DNS (mDNS)

   mDNS is a protocol that resolves host names in small networks that do not have a DNS server. For the SNMP agent, mDNS is disabled by default. To enable:

```
(config)> service snmp mdns enable true
(config>
```

9. (Optional) Set the authentication type. Allowed values are **MD5** or **SHA**. The default is **MD5**.

```
(config)> service snmp auth_type SHA
(config)>
```

10. (Optional) Set the privacy passphrase. If not set, the password, entered above, is used.

```
(config)> service snmp privacy pwd
(config)>
```

11. (Optional) Set the privacy protocol, either **DES** or **AES**. The default is **DES**.

```
(config)> service snmp privacy_protocol AES
(config)>
```

12. (Optional) Enable read-only access to to SNMP version 2c.

```
(config)> service snmp enable 2c true
(config)>
```

13. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

14. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Download MIBs

This procedure is available from the WebUI only.

Required configuration items

- Enable SNMP.

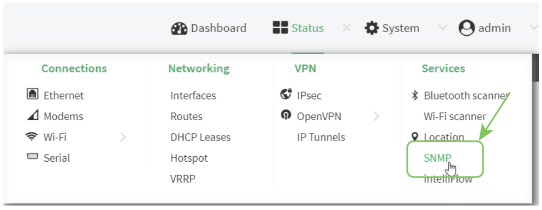
To download a .zip archive of the SNMP MIBs supported by this device:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. Enable SNMP.

   See Configure Simple Network Management Protocol (SNMP) for information about enabling and configuring SNMP support on the IX15 device.
3. On the main menu, click **Status**. Under **Services**, click **SNMP**.

   Note If you have recently enabled SNMP and the SNMP option is not visible, refresh your browser.



   The **SNMP** page is displayed.



4. Click **Download**.

Location information

Your IX15 device can be configured to use the following location sources:

- User-defined static location.
- Location messages forwarded to the device from other location-enabled devices.

You can also configure your IX15 device to forward location messages, either from the IX15 device or from external sources, to a remote host. Additionally, the device can be configured to use a geofence, to allow you to determine actions that will be taken based on the physical location of the device.

This section contains the following topics:

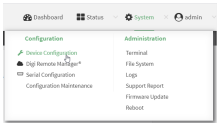
Configure the location service 467
Configure the device to use a user-defined static location 469
Configure the device to accept location messages from external sources 471
Forward location information to a remote host 475
Configure geofencing 482
Show location information 494

Configure the location service

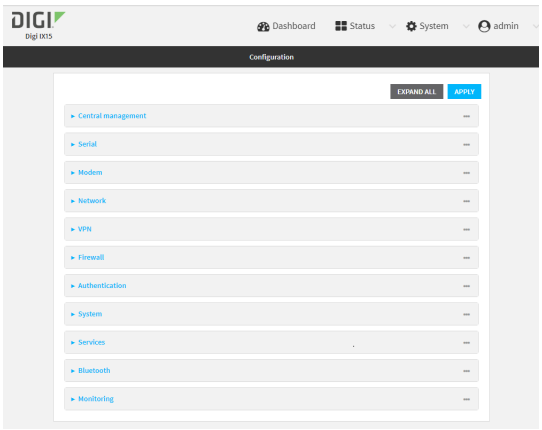
The location service is enabled by default. You can disable it, or you can enable it if it has been disabled.

WebUI

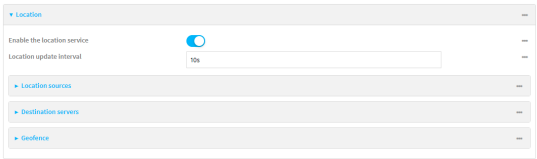
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **Location**.



4. The location service is enabled by default. To disable, or to enable if it has been disabled, click **Enable**.
5. For **Location update interval**, type the amount of time to wait between polling location sources for new location data. The default is ten seconds.

   Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

   For example, to set **Location update interval** to ten minutes, enter **10m** or **600s**.
6. For information about configuring **Location sources**, see the following:
   a. To set a static location for the device, see Configure the device to use a user-defined static location.

b. To accept location information from an external location-enabled server, see Configure the device to accept location messages from external sources.

If multiple location sources are enabled at the same time, the device's location will be determined based on the order that the location sources are listed here.

7. For information about configuring **Destination servers**, see Forward location information to a remote host.
8. For information about configuring **Geofence**, see Configure geofencing.
9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable or disable the GNSS module:
   - To enable the module:

```
(config)> service location gnss true
(config)>
```

   - To disable the module:

```
(config)> service location gnss false
(config)>
```

4. Set the amount of time that the IX15 device will wait before polling location sources for updated location data:

```
(config)> service location interval value
(config)>
```

   where *value* is any number of hours, minutes, or seconds, and takes the format ***number*** {**h**|**m**|**s**}.

   For example, to set **interval** to ten minutes, enter either **10m** or **600s**:

```
(config)> service location interval 600s
(config)>
```

   The default is 10 seconds.

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

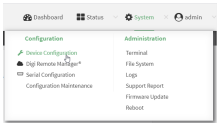
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure the device to use a user-defined static location

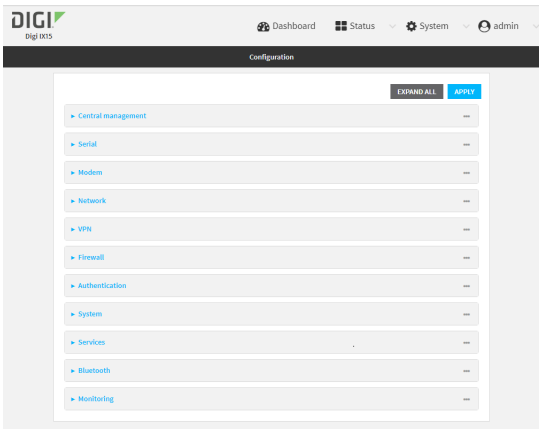
You can configured your IX15 device to use a user-defined static location.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **Location** > **Location sources**.
4. Click ✚ to add a location source.
5. (Optional) Type a **Label** for this location source.
6. For **Latitude**, type the latitude of the device. Allowed values are **-90** and **90**, with up to six decimal places.
7. For **Longitude**, type the longitude of the device. Allowed values are **-180** and **180**, with up to six decimal places.
8. For **Altitude**, type the altitude of the device. Allowed values are an integer followed by **m** or **km**, for example, **100m** or **1km**.

9. The location source is enabled by default. Click **Enable the location source** to disable the location source, or to enable it if it has been disabled.
10. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a location source:

```
(config)> add service location source end
(config service location source 0)>
```

4. (Optional) Set a label for this location source:

```
(config service location source 0)> label "label"
(config)>
```

5. Set the **type** of location source to **server**:

```
(config service location source 0)> type user_defined
(config service location source 0)>
```

6. Set the latitude of the device:

```
(config service location source 0 coordinates latitude int
(config service location source 0)>
```

   where *int* is any integer between **-90** and **90**, with up to six decimal places.
7. Set the longitude of the device:

```
(config service location source 0 coordinates longitude int
(config service location source 0)>
```

   where *int* is any integer between **-180** and **180**, with up to six decimal places.
8. Set the altitude of the device:

```
(config service location source 0 coordinates altitude alt
(config service location source 0)>
```

   Where *alt* is an integer followed by **m** or **km**, for example, **100m** or **1km**.

9. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

10. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure the device to accept location messages from external sources

You can configure the IX15 device to accept NMEA and TAIP messages from external sources. For example, location-enabled devices connected to the IX15 device can forward their location information to the device, and then the IX15 device can serve as a central repository for this location information and forward it to a remote host. See Forward location information to a remote host for information about configuring the IX15 device to forward location messages.

This procedure configures a UDP port on the IX15 device that will be used to listen for incoming messages.

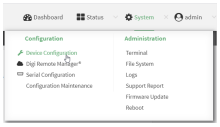
Required configuration items

- The location server must be enabled.
- UDP port that the Digi IX15 Gateway device will listen to for incoming location messages.
- Access control list configuration to provide access to the port through the firewall.

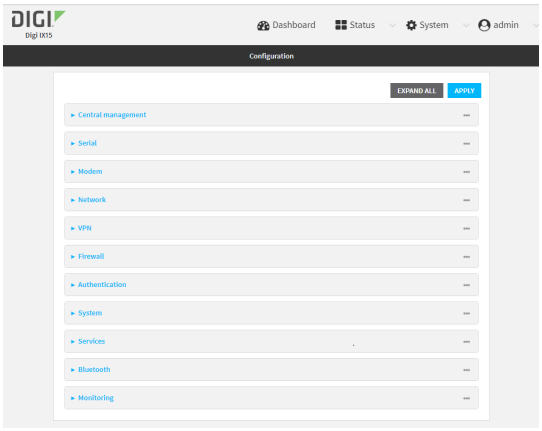
To configure the device to accept location messages from external sources:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **Services** > **Location** > **Location sources**.
4. Click ✚ to add a location source.
5. (Optional) Type a **Label** for this location source.
6. For **Type of location source**, select **Server**.
7. For **Location server port**, type the number of the UDP port that will receive incoming location messages.
8. Click **Access control list** to configure access control:
   - To limit access to specified IPv4 addresses and networks:
     a. Click **IPv4 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv4 address or network that can access the device's location server UDP port. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 192.168.1.0/24.
        - **any**: No limit to IPv4 addresses that can access the location server UDP port.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:
     a. Click **IPv6 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv6 address or network that can access the device's location server UDP port. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 2001:db8::/48.
        - **any**: No limit to IPv6 addresses that can access the location server UDP port.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to hosts connected through a specified interface on the IX15 device:
     a. Click **Interfaces**.
     b. For **Add Interface**, click ✚.

c. For **Interface**, select the appropriate interface from the dropdown.

d. Click ✚ again to allow access through additional interfaces.

- To limit access based on firewall zones:

a. Click **Zones**.

b. For **Add Zone**, click ✚.

c. For **Zone**, select the appropriate firewall zone from the dropdown.

See Firewall configuration for information about firewall zones.

d. Click ✚ again to allow access through additional firewall zones.

9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a location source:

```
(config)> add service location source end
(config service location source 0)>
```

4. (Optional) Set a label for this location source:

```
(config service location source 0)> label "label"
(config service location source 0)>
```

5. Set the **type** of location source to **server**:

```
(config service location source 0)> type server
(config service location source 0)>
```

6. Set the UDP port that will receive incoming location messages.

```
(config service location source 0)> server port port
(config service location source 0)>
```

7. Click **Access control list** to configure access control:

   - To limit access to specified IPv4 addresses and networks:

```
(config)> add service location source 1 acl address end value
(config)>
```

Where *value* can be:

- A single IP address or host name.
- A network designation in CIDR notation, for example, 192.168.1.0/24.
- **any**: No limit to IPv4 addresses that can access the location server UDP port.

Repeat this step to list additional IP addresses or networks.

- To limit access to specified IPv6 addresses and networks:

```
(config)> add service location source 1 acl address6 end value
(config)>
```

Where *value* can be:

- A single IP address or host name.
- A network designation in CIDR notation, for example, 2001:db8::/48.
- **any**: No limit to IPv6 addresses that can access the location server UDP port.

Repeat this step to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX15 device:

```
(config)> add service location source 1 acl interface end value
(config)>
```

Where *value* is an interface defined on your device.

Display a list of available interfaces:

Use **... network interface ?** to display interface information:

```
(config)> ... network interface ?

Interfaces

Additional Configuration
-----------------------------------------
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config)> add service location source 1 acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.

 Additional Configuration
 ------------------------------------------------------------
-----------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

(config)>
```

Repeat this step to list additional firewall zones.

8. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

2. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Forward location information to a remote host

You can configure location clients on the IX15 device that forward location messages in either NMEA or TAIP format to a remote host.

Required configuration items

- Enable the location service.
- The hostname or IP address of the remote host to which the location messages will be forwarded.
- The communication protocol, either TCP or UDP.
- The destination port on the remote host to which the messages will be forwarded.
- Message protocol type of the messages being forwarded, either NMEA or TAIP.

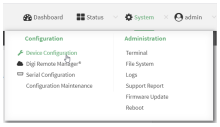
Additional configuration items

- Additional remote hosts to which the location messages will be forwarded.
- Location update interval, which determines how often the device will forward location information to the remote hosts.
- A description of the remote hosts.
- Specific types of NMEA or TAIP messages that should be forwarded.
- If the message protocol is NMEA, configure a talker ID to be used for all messages.
- Text that will be prepended to the forwarded message.
- A vehicle ID that is used in the TAIP ID message and can also be prepended to the forwarded message.

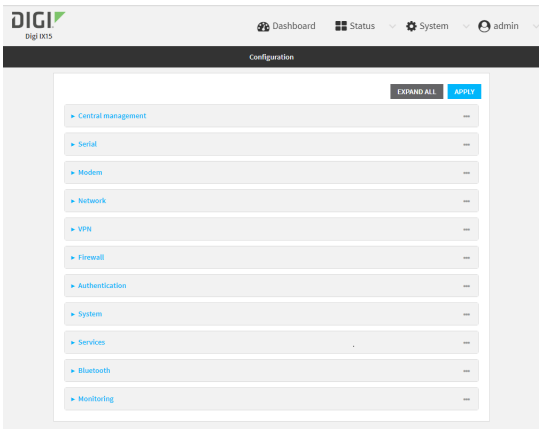
Configure the Digi IX15 Gateway device to forward location information:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **Location** > **Destination servers**.
4. For **Add destination server**, click ✚.
5. (Optional) For **Label**, type a description of the location destination server.
6. For **Destination server**, enter the hostname or IP address of the remote host to which location messages will be sent.
7. For **Destination server port**, enter the UDP or TCP port on the remote host to which location messages will be sent.
8. For **Communication protocol**, select either **UDP** or **TCP**.

9. For **Forward interval multiplier**, select the number of **Location update intervals** to wait before forwarding location data to this server. See Configure the location service for more information about setting the **Location update interval**.
10. For **NMEA filters**, select the filters that represent the types of messages that will be forwarded. By default, all message types are forwarded.
    - To remove a filter:
      a. Click the down arrow (▾) next to the appropriate message type.
      b. Click **Delete**.
    - To add a message type:
      a. For **Add NMEA filter** or **Add TAIP filter**, click ✚.
      b. Select the filter type. Allowed values are:
         - **GGA**: Reports time, position, and fix related data.
         - **GLL**: Reports position data: position fix, time of position fix, and status.
         - **GSA**: Reports GPS DOP and active satellites.
         - **GSV**: Reports the number of SVs in view, PRN, elevation, azimuth, and SNR.
         - **RMC**: Reports position, velocity, and time.
         - **VTG**: Reports direction and speed over ground.
11. For **TAIP filters**, select the filters that represent the types of messages that will be forwarded. By default, all message types are forwarded.
    - To remove a filter:
      a. Click the down arrow (▾) next to the appropriate message type.
      b. Click **Delete**.
    - To add a message type:
      a. For **Add NMEA filter** or **Add TAIP filter**, click ✚.
      b. Select the filter type. Allowed values are:
         - **AL**: Reports altitude and vertical velocity.
         - **CP**: Compact position: reports time, latitude, and longitude.
         - **ID**: Reports the vehicle ID.
         - **LN**: Long navigation: reports the latitude, longitude, and altitude, the horizontal and vertical speed, and heading.
         - **PV**: Position/velocity: reports the latitude, longitude, and heading.
12. For **Outgoing message type**, select either **NMEA** or **TAIP** for the type of message that the device will forward to a remote host.
    - (Optional) If **NMEA** is selected, select a **Talker ID**.

      The talker ID is a two-character prefix in the NMEA message that identifies the source type. The talker ID set here will override the talker ID from all sources, and all forwarded sentences will use the configured ID. The default setting is **Default**, which means that the talker ID provided by the source will be used.
13. (Optional) For **Prepend text**, enter text to prepend to the forwarded message. Two variables can be included in the prepended text:

- **%s**: Includes the Digi IX15 Gateway device's serial number in the prepended text.
- **%v**: Includes the vehicle ID in the prepended text.

For example, to include both the device's serial number and vehicle ID in the prepend message, you can enter the following in the **Prepend** field:

```
__|%s|__|%v|__
```

14. Type a four-digit alphanumeric **Vehicle ID** that will be included with to location messages. If no vehicle ID is configured, this setting defaults to 0000.
15. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a remote host to which location messages will be sent:

```
(config)> add service location forward end
(config service location forward 0)>
```

4. Set the hostname or IP address of the remote host to which location messages will be sent:

```
(config service location forward 0)> server host
(config service location forward 0)>
```

5. Set the communication protocol to either **upd** or **tcp**:

```
(config service location forward 0)> protocol protocol
(config service location forward 0)>
```

6. Set the TCP or UDP port on the remote host to which location messages will be sent:

```
(config service location forward 0)> server_port 8000
(config service location forward 0)>
```

7. Set the number of **Location update intervals** to wait before forwarding location data to this server. See Configure the location service for more information about setting the **Location update interval**.

```
(config service location forward 0)> interval_multiplier int
(config service location forward 0)>
```

8. Set the protocol type for the messages. Allowed values are **taip** or **nmea**; the default is **taip**:

```
(config service location forward 0)> type nmea
(config service location forward 0)>
```

   - (Optional) If the protocol type is set to **nmea**, configure a **Talker ID**.

     The talker ID is a two-character prefix in the NMEA message that identifies the source type. The talker ID set here will override the talker ID from all sources, and all forwarded sentences will use the configured ID.

     a. Use the **?** to determine available talker IDs:

```
(config service location forward 0)> talker_id ?

Talker ID: Setting a talker ID will override the talker ID from
all remote
sources, and all forwarded sentences from remote sources will
use the configured
ID.
Format:
  Default
  GA
  GB
  GI
  GL
  GN
  GP
  GQ
Default value: Default
Current value: Default

(config service location forward 0)>
```

     b. Set the talker ID:

```
(config service location forward 0)> talker_id value
(config service location forward 0)>
```

     The default setting is **Default**, which means that the talker ID provided by the source will be used.

9. (Optional) Set the text to prepend to the forwarded message. Two variables can be included in the prepended text:
   - **%s**: Includes the Digi IX15 Gateway device's serial number in the prepended text.
   - **%v**: Includes the vehicle ID in the prepended text.

```
(config service location forward 0)> prepend __|%s|__|%v|__
(config service location forward 0)>
```

10. (Optional) Set the vehicle ID.

    Allowed value is a four digit alphanumerical string (for example, 01A3 or 1234). If no vehicle ID is configured, this setting defaults to 0000.

```
(config service location forward 0)> vehicle-id 1234
(config service location forward 0)>
```

11. (Optional) Provide a description of the remote host:

```
(config service location forward 0)> label "Remote host 1"
(config service location forward 0)>
```

12. (Optional) Specify types of messages that will be forwarded. Allowed values vary depending on the message protocol type. By default, all message types are forwarded.
    - If the message protocol type is NMEA:

      Allowed values are:
      - **gga**: Reports time, position, and fix related data.
      - **gll**: Reports position data: position fix, time of position fix, and status.
      - **gsa**: Reports GPS DOP and active satellites.
      - **gsv**: Reports the number of SVs in view, PRN, elevation, azimuth, and SNR.
      - **rmc**: Reports position, velocity, and time.
      - **vtg**: Reports direction and speed over ground.

      To remove a message type:

      a. Use the **show** command to determine the index number of the message type to be deleted:

      ```
      (config service location forward 0)> show filter_nmea
      0 gga
      1 gll
      2 gsa
      3 gsv
      4 rmc
      5 vtg
      (config service location forward 0)>
      ```

      b. Use the index number to delete the message type. For example, to delete the **gsa** (index number 2) message type:

      ```
      (config service location forward 0)> del filter_nmea 2
      (config service location forward 0)>
      ```

      To add a message type:

      a. Change to the **filter_nmea** node:

      ```
      (config service location forward 0)> filter_nmea
      (config service location forward 0 filter_nmea)>
      ```

      b. Use the **add** command to add the message type. For example, to add the **gsa** message type:

      ```
      (config service location forward 0 filter_nmea)> add gsa end
      (config service location forward 0 filter_nmea)>
      ```

    - If the message protocol type is TAIP:

Allowed values are:

- **al**: Reports altitude and vertical velocity.
- **cp**: Compact position: reports time, latitude, and longitude.
- **id**: Reports the vehicle ID.
- **ln**: Long navigation: reports the latitude, longitude, and altitude, the horizontal and vertical speed, and heading.
- **pv**: Position/velocity: reports the latitude, longitude, and heading.

To remove a message type:

a. Use the **show** command to determine the index number of the message type to be deleted:

```
(config service location forward 0)> show filter_taip
0 al
1 cp
2 id
3 ln
4 pv
(config service location forward 0)>
```

b. Use the index number to delete the message type. For example, to delete the **id** (index number 2) message type:

```
(config service location forward 0)> del filter_taip 2
(config service location forward 0)>
```

To add a message type:

a. Change to the **filter_taip** node:

```
(config service location forward 0)> filter_taip
(config service location forward 0 filter_taip)>
```

b. Use the **add** command to add the message type. For example, to add the **id** message type:

```
(config service location forward 0 filter_taip)> add id end
(config service location forward 0 filter_taip)>
```

13. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

14. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure geofencing

Geofencing is a mechanism to create a virtual perimeter that allows you configure your IX15 device to perform actions when entering or exiting the perimeter. For example, you can configure a device to factory default if its location service indicates that it has been moved outside of the geofence.

Multiple geofences can be defined for one device, allowing for a complex configuration in which different actions are taken depending on the physical location of the device.

Required configuration items

- Location services must be enabled.
- The geofence must be enabled.
- The boundary type of the geofence, either circular or polygonal.
  - If boundary type is circular, the latitude and longitude of the center point of the circle, and the radius.
  - If boundary type is polygonal, the latitude and longitude of the polygon's vertices (a vertex is the point at which two sides of a polygon meet). Three vertices will create a triangular polygon; four will create a square, etc. Complex polygons can be defined.
- Actions that will be taken when the device's location triggers a geofence event. You can define actions for two types of events:
  - Actions taken when the device enters the boundary of the geofence, or is inside the boundary when the device boots.
  - Actions taken when the device exits the boundary of the geofence, or is outside the boundary when the device boots.

  For each event type:
  - Determine if the action(s) associated with the event type should be performed when the device boots inside or outside of the geofence boundary.
  - The number of update intervals that should take place before the action(s) are taken.

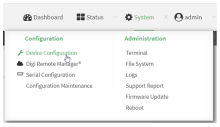
  Multiple actions can be configured for each type of event. For each action:
  - The type of action, either a factory erase or executing a custom script.
  - If a custom script is used:
    - The script that will be executed.
    - Whether to log output and errors from the script.
    - The maximum memory that the script will have available.
    - Whether the script should be executed within a sandbox that will prevent the script from affecting the system itself.

Additional configuration items

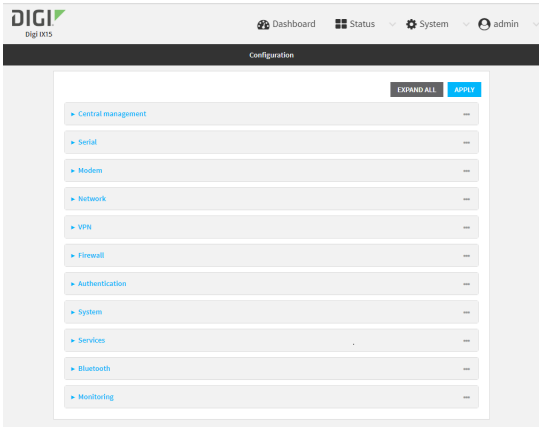
- Update interval, which determines the amount of time that the geofence should wait between polling for updated location data.

WebUI

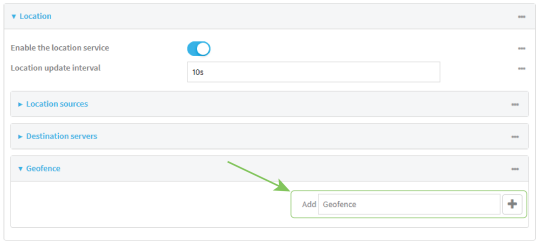
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **Location** > **Geofence**.
4. For **Add Geofence**, type a name for the geofence and click **+**.



The geofence is enabled by default. Click **Enable** to disable, or to enable if it has been disabled.

5. For **Update interval**, type the amount of time that the geofence should wait between polling for updated location data. The default is one minute.

   Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

   For example, to set **Update interval** to ten minutes, enter **10m** or **600s**.
6. For **Boundary type**, select the type of boundary that the geofence will have.
   - If **Circular** is selected:
     a. Click to expand **Center**.
     b. Type the **Latitude** and **Longitude** of the center point of the circle. Allowed values are:

- For **Latitude**, any integer between **-90** and **90**, with up to six decimal places.
- For **Longitude**, any integer between **-180** and **180**, with up to six decimal places.

c. For **Radius**, type the radius of the circle. Allowed values are an integer followed by **m** or **km**, for example, **100m** or **1km**.

- If **Polygonal** is selected:

a. Click to expand **Coordinates**.

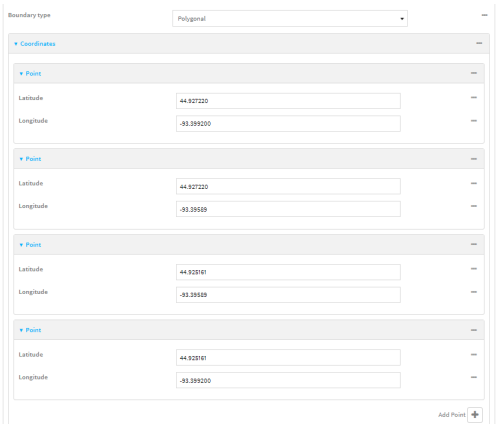
b. Click ✚ to add a point that represents a vertex of the polygon. A vertex is the point at which two sides of a polygon meet.

c. Type the **Latitude** and **Longitude** of one of the vertices of the polygon. Allowed values are:

- For **Latitude**, any integer between **-90** and **90**, with up to six decimal places.
- For **Longitude**, any integer between **-180** and **180**, with up to six decimal places.

d. Click ✚ again to add an additional point, and continue adding points to create the desired polygon.

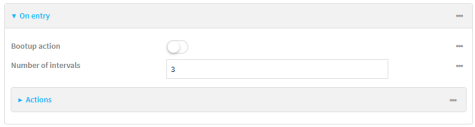
For example, to configure a square polygon around the Digi headquarters, configure a polygon with four points:



This defines a square-shaped polygon equivalent to the following:

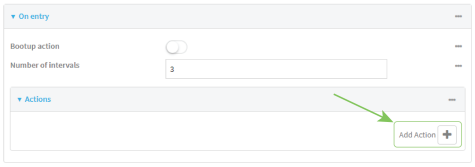


7. Define actions to be taken when the device's location triggers a geofence event:
   - To define actions that will be taken when the device enters the geofence, or is inside the geofence when it boots:
     a. Click to expand **On entry**.



     b. (Optional) Enable **Bootup action** to configure the device to perform the **On entry** actions if the device is inside the geofence when it boots.
     c. For **Number of intervals**, type or select the number of **Update Intervals** that must take place prior to performing the **On entry** actions.

        For example, if the **Update interval** is **1m** (one minute) and the **Number of intervals** is **3**, the **On entry** actions will not be performed until the device has been inside the geofence for three minutes.
     d. Click to expand **Actions**.
     e. Click ✚ to create a new action.



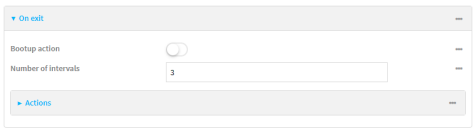
     f. For Action type, select either:
        - **Factory erase** to erase the device configuration when the action is triggered.
        - **Custom script** to execute a custom script when the action is triggered.

        If **Custom script** is selected:
        i. Click to expand **Custom script**.
        ii. For **Commands**, type the script that will be executed when the action is triggered. If the script begins with **#!**, then the proceeding file path will be used to invoke the script interpreter. If not, then the default shell will be used.
        iii. Enable **Log script output** to log the output of the script to the system log.
        iv. Enable **Log script errors** to log errors from the script to the system log.
        v. (Optional) For **Maximum memory**, type the maximum amount of system memory that will be available for the script and it spawned processes.

           Allowed values are any integer followed by one of the following: **b**|**bytes**|**KB**|**k**|**MB**|**M**|**GB**|**G**|**TB**|**T**.

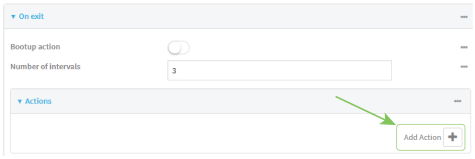
           For example. the allocate one megabyte of memory to the script and its spawned processes, type **1MB** or **1M**.
        vi. **Sandbox** is enabled by default. This prevents the script from adversely affecting the system. If you disable **Sandbox**, the script may render the system unusable.
        vii. Repeat for any additional actions.

- To define actions that will be taken when the device exits the geofence, or is outside the geofence when it boots:
  a. Click to expand **On exit**.



  b. (Optional) Enable **Bootup action** to configure the device to perform the **On exit** actions if the device is inside the geofence when it boots.
  c. For **Number of intervals**, type or select the number of **Update Intervals** that must take place prior to performing the **On exit** actions.

     For example, if the **Update interval** is **1m** (one minute) and the **Number of intervals** is **3**, the **On entry** actions will not be performed until the device has been inside the geofence for three minutes.
  d. Click to expand **Actions**.
  e. Click ✚ to create a new action.



  f. For Action type, select either:
     - **Factory erase** to erase the device configuration when the action is triggered.
     - **Custom script** to execute a custom script when the action is triggered.

     If **Custom script** is selected:
     i. Click to expand **Custom script**.
     ii. For **Commands**, type the script that will be executed when the action is triggered. If the script begins with **#!**, then the proceeding file path will be used to invoke the script interpreter. If not, then the default shell will be used.
     iii. Enable **Log script output** to log the output of the script to the system log.
     iv. Enable **Log script errors** to log errors from the script to the system log.
     v. (Optional) For **Maximum memory**, type the maximum amount of system memory that will be available for the script and it spawned processes.

        Allowed values are any integer followed by one of the following: **b**|**bytes**|**KB**|**k**|**MB**|**M**|**GB**|**G**|**TB**|**T**.

        For example. the allocate one megabyte of memory to the script and its spawned processes, type **1MB** or **1M**.
     vi. **Sandbox** is enabled by default. This prevents the script from adversely affecting the system. If you disable **Sandbox**, the script may render the system unusable.
     vii. Repeat for any additional actions.

8. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a geofence:

```
(config)> add service location geofence name
(config service location geofence name)>
```

   where *name* is a name for the geofence. For example:

```
(config)> add service location geofence test_geofence
(config service location geofence test_geofence)>
```

   The geofence is enabled by default. To disable:

```
(config service location geofence test_geofence)> enable false
(config service location geofence test_geofence)>
```

4. Set the amount of time that the geofence should wait between polling for updated location data:

```
(config service location geofence test_geofence)> update_interval value
(config service location geofence test_geofence)>
```

   where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

   For example, to set **update_interval** to ten minutes, enter either **10m** or **600s**:

```
(config service location geofence test_geofence)> update_interval 600s
(config service location geofence test_geofence)>
```

   The default is **1m** (one minute).

5. Set the boundary type for the geofence:

```
(config service location geofence test_geofence)> boundary value
(config service location geofence test_geofence)>
```

where *value* is either **circular** or **polygonal**.

- If **boundary** is set to **circular** :
  a. Set the latitude and longitude of the center point of the circle:

```
(config service location geofence test_geofence)> center
latitude int
(config service location geofence test_geofence)> center
longitude int
(config service location geofence test_geofence)>
```

  where *int* is:
  - For **latitude**, any integer between **-90** and **90**, with up to six decimal places.
  - For **longitude**, any integer between **-180** and **180**, with up to six decimal places.

  b. Set the radius of the circle:

```
(config service location geofence test_geofence)> radius radius
(config service location geofence test_geofence)>
```

  where *radius* is an integer followed by **m** or **km**, for example, **100m** or **1km**.

- If **boundary** is set to **polygonal**:
  a. Set the coordinates of one vertex of the polygon. A vertex is the point at which two sides of a polygon meet.
    i. Add a vertex:

```
(config service location geofence test_geofence)> add
coordinates end
(config service location geofence test_geofence coordinates
0)>
```

    ii. Set the latitude and longitude of the vertex:

```
(config service location geofence test_geofence coordinates
0)> latitude int
(config service location geofence test_geofence coordinates
0)> longitude int
(config service location geofence test_geofence coordinates
0)>
```

  where *int* is:
  - For **latitude**, any integer between **-90** and **90**, with up to six decimal places.
  - For **longitude**, any integer between **-180** and **180**, with up to six decimal places.

iii. Configure additional vortices:

```
(config service location geofence test_geofence coordinates
0)> ..
(config service location geofence test_geofence coordinates)>
add end
(config service location geofence test_geofence coordinates
1)> latitude int
(config service location geofence test_geofence coordinates
1)> longitude int
(config service location geofence test_geofence coordinates
1)>
```

where *int* is:

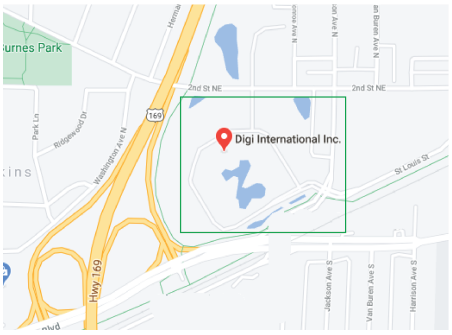
- For **latitude**, any integer between **-90** and **90**, with up to six decimal places.
- For **longitude**, any integer between **-180** and **180**, with up to six decimal places.

Repeat for each vortex of the polygon.
For example, to configure a square polygon around the Digi headquarters, configure a polygon with four points:

```
(config service location geofence test_geofence)> add
coordinates end
(config service location geofence test_geofence coordinates
0)> latitude 44.927220
(config service location geofence test_geofence coordinates
0)> longitude -93.399200
(config service location geofence test_geofence coordinates
0)> ..
(config service location geofence test_geofence coordinates)>
add end
(config service location geofence test_geofence coordinates
1)> latitude 44.927220
(config service location geofence test_geofence coordinates
1)> longitude -93.39589
(config service location geofence test_geofence coordinates
1)> ..
(config service location geofence test_geofence coordinates)>
add end
(config service location geofence test_geofence coordinates
2)> latitude 44.925161
(config service location geofence test_geofence coordinates
2)> longitude -93.39589
(config service location geofence test_geofence coordinates
2)> ..
(config service location geofence test_geofence coordinates)>
add end
```

```
(config service location geofence test_geofence coordinates
3)> latitude 44.925161
(config service location geofence test_geofence coordinates
3)> longitude -93.399200
(config service location geofence test_geofence coordinates
3)>
```

This defines a square-shaped polygon equivalent to the following:



6. Define actions to be taken when the device's location triggers a geofence event:
   - To define actions that will be taken when the device enters the geofence, or is inside the geofence when it boots:
     a. (Optional) Configure the device to preform the actions if the device is inside the geofence when it boots:

```
(config)> service location geofence test_geofence on_entry
bootup true
(config)>
```

     b. Set the number of update_intervals that must take place prior to performing the actions:

```
(config)> service location geofence test_geofence on_entry num_
intervals int
(config)>
```

     For example, if the update interval is **1m** (one minute) and the **num_intervals** is set to **3**, the actions will not be performed until the device has been inside the geofence for three minutes.

     c. Add an action:
        i. Type **...** to return to the root of the configuration:

```
(config service location geofence test_geofence coordinates
3)> ...
(config)>
```

ii. Add the action:

```
(config)> add service location geofence test_geofence on_
entry action end
(config service location geofence test_geofence on_entry
action 0)>
```

d. Set the type of action:

```
(config service location geofence test_geofence on_entry action
0)> type value
(config service location geofence test_geofence on_entry action
0)>
```

where *value* is either:

- **factory_erase**—Erases the device configuration when the action is triggered.
- **script**—Executes a custom script when the action is triggered.

factory_erase or **script**.

If **type** is set to **script**:

i. Type or paste the script, closed in quote marks:

```
(config service location geofence test_geofence on_entry
action 0)> commands "script"
(config service location geofence test_geofence on_entry
action 0)>
```

If the script begins with **#!**, then the proceeding file path will be used to invoke the script interpreter. If not, then the default shell will be used.

ii. To log the output of the script to the system log:

```
(config service location geofence test_geofence on_entry
action 0)> syslog_stdout true
(config service location geofence test_geofence on_entry
action 0)>
```

iii. To log the errors from the script to the system log:

```
(config service location geofence test_geofence on_entry
action 0)> syslog_stderr true
(config service location geofence test_geofence on_entry
action 0)>
```

iv. (Optional) Set the maximum amount of system memory that will be available for the script and it spawned processes:

```
(config service location geofence test_geofence on_entry
action 0)> max_memory value
(config service location geofence test_geofence on_entry
action 0)>
```

where *value* is any integer followed by one of the following: **b**|**bytes**|**KB**|**k**|**MB**|**M**|**GB**|**G**|**TB**|**T**.

For example. the allocate one megabyte of memory to the script and its spawned processes:

```
(config service location geofence test_geofence on_entry
action 0)> max_memory 1MB
(config service location geofence test_geofence on_entry
action 0)>
```

v. A sandbox is enabled by default to prevent the script from adversely affecting the system. To disable the sandbox:

```
(config service location geofence test_geofence on_entry
action 0)> sandbox false
(config service location geofence test_geofence on_entry
action 0)>
```

If you disable the sandbox, the script may render the system unusable.

vi. Repeat for any additional actions.

- To define actions that will be taken when the device exits the geofence, or is outside the geofence when it boots:
  a. (Optional) Configure the device to preform the actions if the device is outside the geofence when it boots:

```
(config)> service location geofence test_geofence on_exit bootup
true
(config)>
```

  b. Set the number of update_intervals that must take place prior to performing the actions:

```
(config)> service location geofence test_geofence on_exit num_
intervals int
(config)>
```

For example, if the update interval is **1m** (one minute) and the **num_intervals** is set to **3**, the actions will not be performed until the device has been outside the geofence for three minutes.

  c. Add an action:
    i. Type **...** to return to the root of the configuration:

```
(config service location geofence test_geofence coordinates
3)> ...
(config)>
```

    ii. Add the action:

```
(config)> add service location geofence test_geofence on_exit
action end
```

```
(config service location geofence test_geofence on_exit
action 0)>
```

d. Set the type of action:

```
(config service location geofence test_geofence on_exit action
0)> type value
(config service location geofence test_geofence on_exit action
0)>
```

where *value* is either:

- **factory_erase**—Erases the device configuration when the action is triggered.
- **script**—Executes a custom script when the action is triggered.

factory_erase or **script**.

If **type** is set to **script**:

i. Type or paste the script, closed in quote marks:

```
(config service location geofence test_geofence on_exit
action 0)> commands "script"
(config service location geofence test_geofence on_exit
action 0)>
```

If the script begins with **#!**, then the proceeding file path will be used to invoke the script interpreter. If not, then the default shell will be used.

ii. To log the output of the script to the system log:

```
(config service location geofence test_geofence on_exit
action 0)> syslog_stdout true
(config service location geofence test_geofence on_exit
action 0)>
```

iii. To log the errors from the script to the system log:

```
(config service location geofence test_geofence on_exit
action 0)> syslog_stderr true
(config service location geofence test_geofence on_exit
action 0)>
```

iv. (Optional) Set the maximum amount of system memory that will be available for the script and it spawned processes:

```
(config service location geofence test_geofence on_exit
action 0)> max_memory value
(config service location geofence test_geofence on_exit
action 0)>
```

where *value* is any integer followed by one of the following: **b**|**bytes**|**KB**|**k**|**MB**|**M**|**GB**|**G**|**TB**|**T**.

For example. the allocate one megabyte of memory to the script and its spawned processes:

```
(config service location geofence test_geofence on_exit
action 0)> max_memory 1MB
(config service location geofence test_geofence on_exit
action 0)>
```

v. A sandbox is enabled by default to prevent the script from adversely affecting the system. To disable the sandbox:

```
(config service location geofence test_geofence on_exit
action 0)> sandbox false
(config service location geofence test_geofence on_exit
action 0)>
```

If you disable the sandbox, the script may render the system unusable.

vi. Repeat for any additional actions.

7. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

8. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show location information

You can view status and statistics about location information from either the WebUI or the command line.

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **Status**.
3. Under **Services**, click **Location**.

   The device's current location is displayed, along with the status of any configured geofences.

Command line

Show location information

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Use the show location command at the system prompt:

```
> show location

 Location Status
```

```
---------------
State              : enabled
Source             : 192.168.2.3
Latitude           : 44* 55' 14.809" N (44.92078)
Longitude          : 93* 24' 47.262" w (-93.413128)
Altitude           : 279 meters
Velocity           : 0 meters per second
Direction          : None
Quality            : Standard GNSS (2D/3D)
UTC Date and Time : Thurs, 03 March 2022 10:16:23 03
No. of Satellites : 7

>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show geofence information

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Use the show location geofence command at the system prompt:

```
> show location geofence

Geofence       Status  State   Transitions  Last Transition
-------------  ------  ------  -----------  ---------------
test_geofence  Up      Inside  0

>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Modbus gateway

The IX15 supports the ability to function as a Modbus gateway, to provide serial-to-Ethernet connectivity to Programmable Logic Controllers (PLCs), Remote Terminal Units (RTUs), and other industrial devices. MODBUS provides client/server communication between devices connected on different types of buses and networks, and the IX15 gateway allows for communication between buses and and networks that use the Modbus protocol.

This section contains the following topics:

Configure the Modbus gateway 496
Show Modbus gateway status and statistics 509

Configure the Modbus gateway

Required configuration items

- Server configuration:
  - Enable the server.
  - Connection type, either socket or serial.
    - If the connection type is socket, the IP protocol to be used.
    - If the connection type is serial, the serial port to be used.
- Client configuration:
  - Enable the client.
  - Connection type, either socket or serial.
    - If the connection type is socket:
      - The IP protocol to be used.
      - The hostname or IPv4 address of the remote host on which the Modbus server is running.
    - If the connection type is serial:
      - The serial port to be used.
  - Modbus address or addresses to determine if messages should be forwarded to a destination device.

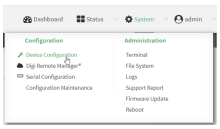
Additional configuration items

- Server configuration:
  - The packet mode.
  - The maximum time between bytes in a packet.
  - If the connection type is set to socket:
    - The port to use.
    - The inactivity timeout.
    - Access control list.
  - If the connection type is set to serial:
    - Whether to use half duplex (two wire) mode.
- Client configuration:
  - The packet mode.
  - The maximum time between bytes in a packets.
  - Whether to send broadcast messages.
  - Response timeout
  - If connection type is set to socket:
    - The port to use.
    - The inactivity timeout.
  - If connection type is set to serial:
    - Whether to use half duplex (two wire) mode.

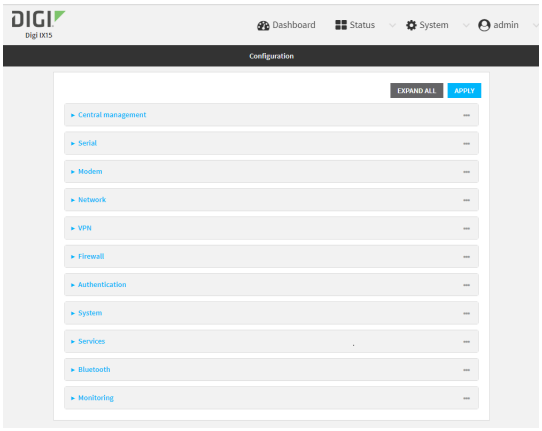
- Whether packets should be delivered to a fixed Modbus address.
- Whether packets should have their Modbus address adjusted downward before to delivery.

WebUI

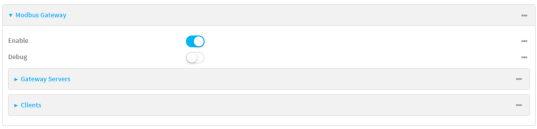
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



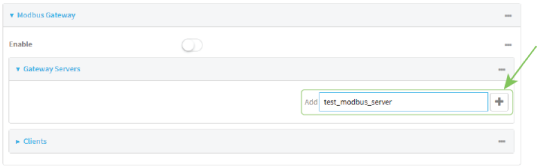
3. Click **Services** > **Modbus Gateway**.



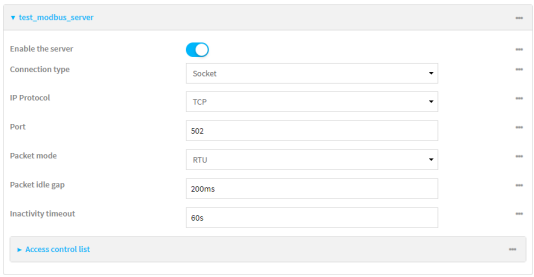
4. Click **Enable** to enable the gateway.
5. Click **Debug** to allow verbose logging in the system log.

Configure gateway servers

1. Click to expand **Gateway Servers**.
2. For **Add Modbus server**, type a name for the server and click ✚.



The new Modbus gateway server configuration is displayed.



3. The new Modbus gateway server is enabled by default. Toggle off **Enable the server** to disable.
4. For **Connection type**, select **Socket** or **Serial**. Available options in the gateway server configuration vary depending on this setting.
   - If **Socket** is selected for **Connection type**:
     a. For **IP Protocol**, select **TCP** or **UDP**. The default is **TCP**.
     b. For **Port**, enter or select an appropriate port. The default is port **502**.
   - If **Serial** is selected for **Connection type**:
     a. For **Serial port**, select the appropriate serial port on the IX15 device.
5. For **Packet mode**, select **RTU** or **RAW** (if **Connection type** is set to **Socket**) or **ASCII** (if **Connection type**is set to **Serial**) for the type of packet that will be used by this connection. The default is **RTU**.
6. For **Packet idle gap**, type the maximum allowable time between bytes in a packet.

   Allowed values are between 10 milliseconds and one second, and take the format ***number*** {**ms|s**}.

   For example, to set **Packet idle gap** to 20 milliseconds, enter **20ms**.
7. If **Connection type** is set to **Socket**, for **Inactivity timeout**, type the amount of time to wait before disconnecting the socket when it has become inactive.

   Allowed values are any number of minutes or seconds up to a maximum of 15 minutes, and take the format ***number***{**m|s**}.

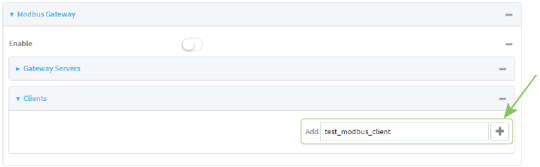
   For example, to set **Inactivity timeout** to ten minutes, enter **10m** or **600s**.
8. (Optional) If **Connection type** is set to **Serial**, click **Half duplex** to enable half duplex (two wire) mode.
9. (Optional) If **Connection type** is set to **Socket**, click to expand **Access control list**:
   - To limit access to specified IPv4 addresses and networks:
     a. Click **IPv4 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv4 address or network that can access the device's web administration service. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 192.168.1.0/24.
        - **any**: No limit to IPv4 addresses that can access the web administration service.
     d. Click ✚ again to list additional IP addresses or networks.

- To limit access to specified IPv6 addresses and networks:
  a. Click **IPv6 Addresses**.
  b. For **Add Address**, click ✚.
  c. For **Address**, enter the IPv6 address or network that can access the device's web administration service. Allowed values are:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 2001:db8::/48.
     - **any**: No limit to IPv6 addresses that can access the web administration service.
  d. Click ✚ again to list additional IP addresses or networks.
- To limit access to hosts connected through a specified interface on the IX15 device:
  a. Click **Interfaces**.
  b. For **Add Interface**, click ✚.
  c. For **Interface**, select the appropriate interface from the dropdown.
  d. Click ✚ again to allow access through additional interfaces.
- To limit access based on firewall zones:
  a. Click **Zones**.
  b. For **Add Zone**, click ✚.
  c. For **Zone**, select the appropriate firewall zone from the dropdown.
     See Firewall configuration for information about firewall zones.
  d. Click ✚ again to allow access through additional firewall zones.

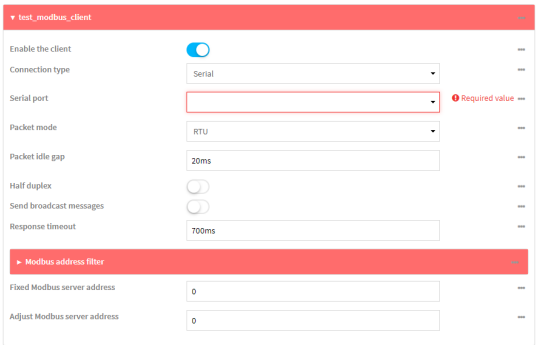
10. Repeat these steps to configure additional servers.

Configure clients

1. Click to expand **Clients**.
2. For **Add Modbus client**, type a name for the client and click ✚.



The new Modbus gateway client configuration is displayed.



3. The new Modbus gateway client is enabled by default. Toggle off **Enable the client** to disable.
4. For **Connection type**, select **Socket** or **Serial**. Available options in the gateway server configuration vary depending on this setting.
    - If **Socket** is selected for **Connection type**:
        a. For **IP Protocol**, select **TCP** or **UDP**. The default is **TCP**.
        b. For **Port**, enter or select an appropriate port. The default is port **502**.
        c. For **Remote host**, type the hostname or IP address of the remote host on which the Modbus server is running.
    - If **Serial** is selected for **Connection type**:
        a. For **Serial port**, select the appropriate serial port on the IX15 device.
5. For **Packet mode**, select **RTU** or **RAW** (if **Connection type** is set to **Socket**) or **ASCII** (if **Connection type**is set to **Serial**) for the type of packet that will be used by this connection. The default is **RTU**.
6. For **Packet idle gap**, type the maximum allowable time between bytes in a packet.

    Allowed values are between 10 milliseconds and one second, and take the format ***number*** {**ms**|**s**}.

    For example, to set **Packet idle gap** to 20 milliseconds, enter **20ms**.
7. If **Connection type** is set to **Socket**, for **Inactivity timeout**, type the amount of time to wait before disconnecting the socket when it has become inactive.

    Allowed values are any number of minutes or seconds up to a maximum of 15 minutes, and take the format ***number***{**m**|**s**}.

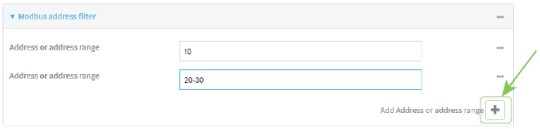
    For example, to set **Inactivity timeout** to ten minutes, enter **10m** or **600s**.
8. (Optional) If **Connection type** is set to **Serial**, click **Half duplex** to enable half duplex (two wire) mode.
9. (Optional) If **Connection type** is set to **Socket**, click to expand **Access control list**:
    - To limit access to specified IPv4 addresses and networks:
        a. Click **IPv4 Addresses**.
        b. For **Add Address**, click ✚.
        c. For **Address**, enter the IPv4 address or network that can access the device's web administration service. Allowed values are:
            - A single IP address or host name.
            - A network designation in CIDR notation, for example, 192.168.1.0/24.
            - **any**: No limit to IPv4 addresses that can access the web administration service.
        d. Click ✚ again to list additional IP addresses or networks.
    - To limit access to specified IPv6 addresses and networks:
        a. Click **IPv6 Addresses**.
        b. For **Add Address**, click ✚.
        c. For **Address**, enter the IPv6 address or network that can access the device's web administration service. Allowed values are:
            - A single IP address or host name.
            - A network designation in CIDR notation, for example, 2001:db8::/48.

- **any**: No limit to IPv6 addresses that can access the web administration service.

d. Click ✚ again to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX15 device:
  a. Click **Interfaces**.
  b. For **Add Interface**, click ✚.
  c. For **Interface**, select the appropriate interface from the dropdown.
  d. Click ✚ again to allow access through additional interfaces.
- To limit access based on firewall zones:
  a. Click **Zones**.
  b. For **Add Zone**, click ✚.
  c. For **Zone**, select the appropriate firewall zone from the dropdown.
  See Firewall configuration for information about firewall zones.
  d. Click ✚ again to allow access through additional firewall zones.

10. (Optional) Enable **Send broadcast messages** to configure the gateway to send broadcast messages to this client.
11. For **Response timeout**, type the maximum time to wait for a response to a message.
    Allowed values are between 1 millisecond and 700 milliseconds, and take the format ***number*ms**.
    For example, to set **Response timeout** to 100 milliseconds, enter **100ms**. The default is **700ms**.
12. Click to expand **Modbus address filter**.
    This filter is used by the gateway to determine if a message should be forwarded to a destination device. If the Modbus address in the message matches one or more of the filters, the message is forwarded. If it does not match the filters, the message is not forwarded.
13. For **Address or address range**, type a Modbus address or range of addresses. Allowed values are **1** through **255** or a hyphen-separated range.
    For example, to have this client filter for incoming messages that contain the Modbus address of 10, type **10**. To filter for all messages with addresses in the range of 20 to 30, type **20-30**.
    To add additional address filters for this client, click ✚.



14. For **Fixed Modbus server address**, if request messages handled by this client should always be forwarded to a specific device, type the device's Modbus address. Leave at the default setting of **0** to allow messages that match the **Modbus address filter** to be forwarded to devices based on the Modbuss address in the message.
15. For **Adjust Modbus server address**, type a value to adjust the Modbus server address downward by the specified value prior to delivering the message. Allowed values are **0** through **255**. Leave at the default setting of **0** to not adjust the server address.
    If a packet contains a Modbus server address above the amount entered here, the address will be adjusted downward by this amount before the packet is delivered. This allows you to configure clients on the gateway that will forward messages to remote devices with the same

Modbus address on different buses. For example, if there are two devices on two different buses that have the same Modbus address of 10, you can create two clients on the gateway:

- Client one:
  - **Modbus address filter** set to **10**.

  This will configure the gateway to deliver all messages that have the Modbus server address of 10 to this device.
- Client two:
  - **Modbus address filter** set to **20**.
  - **Adjust Modbus server address** set to **10**.

  This will configure the gateway to deliver all messages that have the Modbus server address address of 20 to the device with address 10.

16. Repeat these steps to configure additional clients.
17. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable the Modbus gateway:

```
(config)> service modbus_gateway enable true
(config)>
```

4. Configure servers:
   a. Add a server:

```
(config)> add service modbus_gateway server name
(config service modbus_gateway  server name)>
```

   where *name* is a name for the server, for example:

```
(config)> add service modbus_gateway server test_modbus_server
(config service modbus_gateway server test_modbus_server)>
```

   The Modbus server is enabled by default. To disable:

```
(config service modbus_gateway server test_modbus_server)> enable
false
(config service modbus_gateway server test_modbus_server)>
```

b. Set the connection type:

```
(config service modbus_gateway server test_modbus_server)> connection_
type type
(config service modbus_gateway server test_modbus_server)>
```

where *type* is either **socket** or **serial**. The default is **socket**.

- If **connection_type** is set to **socket**:
  i. Set the IP protocol:

  ```
  (config service modbus_gateway server test_modbus_server)>
  socket protocol value
  (config service modbus_gateway server test_modbus_server)>
  ```

  where *value* is either **tcp** or **udp**.

  ii. Set the port:

  ```
  (config service modbus_gateway server test_modbus_server)>
  socket port
  (config service modbus_gateway server test_modbus_server)>
  ```

  where *port* is an integer between **1** and **65535**. The default is **502**.

  iii. Set the packet mode:

  ```
  (config service modbus_gateway server test_modbus_server)>
  socket packet_mode value
  (config service modbus_gateway server test_modbus_server)>
  ```

  where *value* is either **rtu** or **raw**. The default is **rtu**.

  iv. Set the maximum allowable time between bytes in a packet:

  ```
  (config service modbus_gateway server test_modbus_server)>
  socket idle_gap value
  (config service modbus_gateway server test_modbus_server)>
  ```

  where *value* is any number between 10 milliseconds and one second, and take the format ***number*{ms|s}**.

  For example, to set idle_gap to 20 milliseconds, enter **20ms**.

  v. Set the amount of time to wait before disconnecting the socket when it has become inactive:

  ```
  (config service modbus_gateway server test_modbus_server)>
  inactivity_timeout value
  (config service modbus_gateway server test_modbus_server)>
  ```

  where *value* is any number of minutes or seconds up to a maximum of 15 minutes, and takes the format ***number***{**m|s**}.

For example, to set **inactivity_timeout** to ten minutes, enter either **10m** or **600s**:

```
(config service modbus_gateway server test_modbus_server)>
inactivity_timeout 600s
(config service modbus_gateway server test_modbus_server)>
```

- If **connection_type** is set to **serial**:
  i. Set the serial port:
    i. Use the **?** to determine available serial ports:

```
(config service modbus_gateway server test_modbus_
server)> ... serial port ?

Serial

  Additional Configuration
  ------------------------------------------------------
------------------------
  port1                     Port 1

(config service modbus_gateway server test_modbus_
server)>
```

    ii. Set the port:

```
(config service modbus_gateway server test_modbus_
server)> serial port
(config service modbus_gateway server test_modbus_
server)>
```

  ii. Set the packet mode:

```
(config service modbus_gateway server test_modbus_server)>
serial packet_mode value
(config service modbus_gateway server test_modbus_server)>
```

  where *value* is either **rtu** or **ascii**. The default is **rtu**.

  iii. Set the maximum allowable time between bytes in a packet:

```
(config service modbus_gateway server test_modbus_server)>
serial idle_gap value
(config service modbus_gateway server test_modbus_server)>
```

  where *value* is any number between 10 milliseconds and one second, and take the format ***number*{ms|s}**.

  For example, to set idle_gap to one second, enter **1000ms** or **1s**.

iv. (Optional) Enable half-duplex (two wire) mode:

```
(config service modbus_gateway server test_modbus_server)>
serial half_duplex true
(config service modbus_gateway server test_modbus_server)>
```

c. Repeat the above instructions for additional servers.

5. Configure clients:

a. Type **...** to return to the root of the configuration:

```
(config)> add service modbus_gateway server test_modbus_server)> ...
(config)>
```

b. Add a client:

```
(config)> add service modbus_gateway client name
(config service modbus_gateway  client name)>
```

where *name* is a name for the client, for example:

```
(config)> add service modbus_gateway client test_modbus_client
(config service modbus_gateway client test_modbus_client)>
```

The Modbus client is enabled by default. To disable:

```
(config service modbus_gateway client test_modbus_client)> enable
false
(config service modbus_gateway client test_modbus_client)>
```

c. Set the connection type:

```
(config service modbus_gateway client test_modbus_client)> connection_
type type
(config service modbus_gateway client test_modbus_client)>
```

where *type* is either **socket** or **serial**. The default is **socket**.

- If **connection_type** is set to **socket**:

i. Set the IP protocol:

```
(config service modbus_gateway client test_modbus_client)>
socket protocol value
(config service modbus_gateway client test_modbus_client)>
```

where *value* is either **tcp** or **udp**.

ii. Set the port:

```
(config service modbus_gateway client test_modbus_client)>
socket port
(config service modbus_gateway client test_modbus_client)>
```

where *port* is an integer between **1** and **65535**. The default is **502**.

iii. Set the packet mode:

```
(config service modbus_gateway client test_modbus_client)>
socket packet_mode value
(config service modbus_gateway client test_modbus_client)>
```

where *value* is either **rtu** or **ascii**. The default is **rtu**.

iv. Set the maximum allowable time between bytes in a packet:

```
(config service modbus_gateway client test_modbus_client)>
socket idle_gap value
(config service modbus_gateway client test_modbus_client)>
```

where *value* is any number between 10 milliseconds and one second, and take the format ***number*{ms|s}**.

For example, to set idle_gap to 20 milliseconds, enter **20ms**.

v. Set the amount of time to wait before disconnecting the socket when it has become inactive:

```
(config service modbus_gateway client test_modbus_client)>
inactivity_timeout value
(config service modbus_gateway client test_modbus_client)>
```

where *value* is any number of minutes or seconds up to a maximum of 15 minutes, and takes the format ***number***{**m|s**}.

For example, to set **inactivity_timeout** to ten minutes, enter either **10m** or **600s**:

```
(config service modbus_gateway client test_modbus_client)>
inactivity_timeout 600s
(config service modbus_gateway client test_modbus_client)>
```

vi. Set the hostname or IP address of the remote host on which the Modbus server is running:

```
(config service modbus_gateway client test_modbus_client)>
remote_host ip_address|hostname
(config service modbus_gateway client test_modbus_client)>
```

- If **connection_type** is set to **serial**:
  i. Set the serial port:
    i. Use the **?** to determine available serial ports:

```
(config service modbus_gateway client test_modbus_
client)> ... serial port ?

Serial

  Additional Configuration
  ------------------------------------------------------
```

```
------------------------
  port1                     Port 1

(config service modbus_gateway client test_modbus_
client)>
```

ii. Set the port:

```
(config service modbus_gateway client test_modbus_
client)> serial port
(config service modbus_gateway client test_modbus_
client)>
```

ii. Set the packet mode:

```
(config service modbus_gateway client test_modbus_client)>
serial packet_mode value
(config service modbus_gateway client test_modbus_client)>
```

where *value* is either **rtu** or **ascii**. The default is **rtu**.

iii. Set the maximum allowable time between bytes in a packet:

```
(config service modbus_gateway client test_modbus_client)>
serial idle_gap value
(config service modbus_gateway client test_modbus_client)>
```

where *value* is any number between 10 milliseconds and one second, and take the format ***number*{ms|s}**.

For example, to set idle_gap to one second, enter **1000ms** or **1s**.

iv. (Optional) Enable half-duplex (two wire) mode:

```
(config service modbus_gateway client test_modbus_client)>
serial half_duplex true
(config service modbus_gateway client test_modbus_client)>
```

d. (Optional) Enable the gateway to send broadcast messages to this client:

```
(config service modbus_gateway client test_modbus_client)> broadcast
true
(config service modbus_gateway client test_modbus_client)>
```

e. Set the maximum time to wait for a response to a message:

```
(config service modbus_gateway client test_modbus_client)> response_
timeout value
(config service modbus_gateway client test_modbus_client)>
```

Allowed values are between 1 millisecond and 700 milliseconds, and take the format ***number*ms**.

For example, to set response_timeout to 100 milliseconds:

```
(config service modbus_gateway client test_modbus_client)> response_
timeout 100ms
(config service modbus_gateway client test_modbus_client)>
```

The default is **700ms**.

f. Configure the address filter:

This filter is used by the gateway to determine if a message should be forwarded to a destination device. If the Modbus address in the message matches one or more of the filters, the message is forwarded. If it does not match the filters, the message is not forwarded. Allowed values are **1** through **255** or a hyphen-separated range.

For example:

- To have this client filter for incoming messages that contain the Modbus address of 10, set the index **0** entry to **10**:

```
(config service modbus_gateway client test_modbus_client)>
filter 0 10
(config service modbus_gateway client test_modbus_client)>
```

- To filter for all messages with addresses in the range of 20 to 30, set the index **0** entry to **20-30**:

```
(config service modbus_gateway client test_modbus_client)>
filter 0 20-30
(config service modbus_gateway client test_modbus_client)>
```

To add additional filters, increment the index number. For example, to add an additional filter for addresses in the range of 50-100:

```
(config service modbus_gateway client test_modbus_client)> filter 1
50-100
(config service modbus_gateway client test_modbus_client)>
```

g. If request messages handled by this client should always be forwarded to a specific device, , use **fixed_server_address** to set the device's Modbus address:

```
(config service modbus_gateway client test_modbus_client)> fixed_
server_address value
(config service modbus_gateway client test_modbus_client)>
```

Leave at the default setting of **0** to allow messages that match the Modbus address filter to be forwarded to devices based on the Modbuss address in the message.

h. To adjust the Modbus server address downward by the specified value prior to delivering the message, use **adjust_server_address**:

```
(config service modbus_gateway client test_modbus_client)> adjust_
server_address value
(config service modbus_gateway client test_modbus_client)>
```

where *value* is an integer from **0** to **255**. Leave at the default setting of **0** to not adjust the server address.

If a packet contains a Modbus server address above the amount entered here, the address will be adjusted downward by this amount before the packet is delivered. This allows you to configure clients on the gateway that will forward messages to remote devices with the same Modbus address on different buses. For example, if there are two devices on two different buses that have the same Modbus address of 10, you can create two clients on the gateway:

- Client one:
  - **filter** set to **10**.

  This will configure the gateway to deliver all messages that have the Modbus server address of 10 to this device.
- Client two:
  - **filter** set to **20**.
  - **adjust_server_address** set to **10**.

  This will configure the gateway to deliver all messages that have the Modbus server address address of 20 to the device with address 10.

i. Repeat the above instructions for additional clients.

6. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show Modbus gateway status and statistics

You can view status and statistics about location information from either the WebUI or the command line.

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, select **Status** > **Modbus Gateway**.

   The **Modbus Gateway** page appears.

   Statistics related to the Modbus gateway server are displayed. If the message **Server connections not available** is displayed, this indicates that there are no connected clients.

   - To view information about Modbus gateway clients, click **Clients**.
   - To view statistics that are common to both the clients and server, click **Common Statistics**.
   - To view configuration details about the gateway, click the (configuration) icon in the upper right of the gateway's status pane.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Use the show modbus-gateway command at the system prompt:

```
> show modbus-gateway

 Server Connection  IP Address   Port   Uptime
 -----------------  -----------  -----  ------
 modbus_socket      10.45.1.139  49570  6
 modbus_socket      10.45.1.139  49568  13

 Client               Uptime
 -------------------  ------
 modbus_socket_41     0
 modbus_socket_21     0
 modbus_serial_client  428

>
```

   If the message **Server connections not available** is displayed, this indicates that there are no connected clients.

3. Use the show modbus-gateway verbose command at the system prompt to display more information:

```
> show modbus-gateway verbose

 Client               Uptime
 -------------------  ------
 modbus_socket_41     0
 modbus_socket_21     0
 modbus_serial_client  506

 Common Statistics
 -----------------
 Configuration Updates          : 1
 Client Configuration Failure   : 0
 Server Configuration Failure   : 0
 Configuration Load Failure     : 0
 Incoming Connections           : 4
 Internal Error                 : 0
 Resource Shortages             : 0

 Servers
 -------

 modbus_socket
 -------------
```

```
Client Lookup Errors           : 0
Incoming Connections           : 4
Packet Errors                  : 0
RX Broadcasts                  : 0
RX Requests                    : 12
TX Exceptions                  : 0
TX Responses                   : 12

Clients
-------

modbus_socket_41
----------------
Address Translation Errors     : 0
Connection Errors              : 0
Packet Errors                  : 0
RX Responses                   : 4
RX Timeouts                    : 0
TX Broadcasts                  : 0
TX Requests                    : 4

modbus_socket_21
----------------
Address Translation Errors     : 0
Connection Errors              : 0
Packet Errors                  : 0
RX Responses                   : 4
RX Timeouts                    : 0
TX Broadcasts                  : 0
TX Requests                    : 4

modbus_serial_client
--------------------
Address Translation Errors     : 0
Connection Errors              : 0
Packet Errors                  : 0
RX Responses                   : 4
RX Timeouts                    : 0
TX Broadcasts                  : 0
TX Requests                    : 4

>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

System time

By default, the IX15 device synchronizes the system time by periodically connecting to the Digi NTP server, **time.devicecloud.com**. In this mode, the device queries the time server based on following events and schedule:

- At boot time.
- Once a day.

The default configuration has the system time zone set to UTC. No additional configuration is required for the system time if the default configuration is sufficient. However, you can change the default time zone and the default NTP server, as well as configuring additional NTP servers. If multiple servers are configured, a number of time samples are obtained from each of the servers and a subset of the NTP clock filter and selection algorithms are applied to select the best of these. See Configure the system time for details about changing the default configuration.

The IX15 device can also be configured to serve as an NTP server, providing NTP services to downstream devices. See Network Time Protocol for more information about NTP server support.

You can also set the local date and time manually, if there is no access to NTP servers. See Manually set the system date and time for information

Configure the system time

This procedure is optional.

The IX15 device's default system time configuration uses the Digi NTP server, **time.devicecloud.com**, and has the time zone set to **UTC**. You can change the default NTP server and the default time zone. You can also set the local date and time without using an upstream NTP server, as well as configuring additional NTP servers.

Required Configuration Items

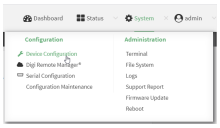
- The time zone for the IX15 device.
- If t least one upstream NTP server for synchronization.

Additional Configuration Options

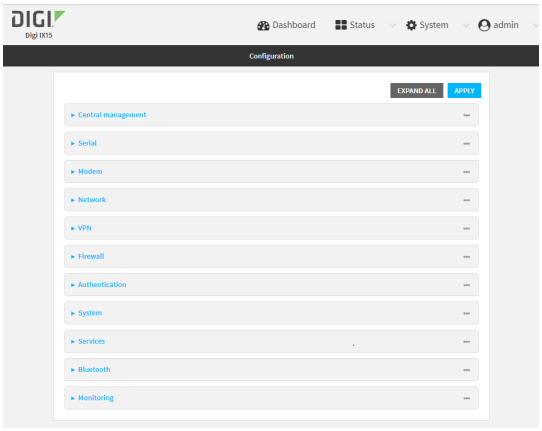
- Additional upstream NTP servers.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **System** > **Time**
4. (Optional) For **Timezone**, select either **UTC** or select the location nearest to your current location to set the timezone for your IX15 device. The default is **UTC**.
5. (Optional) Add upstream NTP servers that the device will use to synchronize its time. The default setting is **time.devicecloud.com**.
   - To change the default value of the NTP server:
     a. Click **NTP servers**.
     b. For **Server**, type a new server name.
   - To add an NTP server:
     a. Click **NTP servers**.
     b. For **Add Server**, click ✚.
     c. For **Server**, enter the hostname of the upstream NTP server that the device will use to synchronize its time.
     d. Click ✚ to add additional NTP servers. If multiple servers are included, servers are tried in the order listed until one succeeds.

Note This list is synchronized with the list of servers included with NTP server configuration, and changes made to one will be reflected in the other. See Configure the device as an NTP server for more information about NTP server configuration.

6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. (Optional) Set the timezone for the location of your IX15 device. The default is **UTC**.

```
(config)> system time timezone value
(config)>
```

Where *value* is the timezone using the format specified with the following command:

```
(config)> system time timezone ?

Timezone: The timezone for the location of this device. This is used to
adjust the time for log
messages. It also affects actions that occur at a specific time of day.
Format:
  Africa/Abidjan
  Africa/Accra
  Africa/Addis_Ababa
  ...

(config)>
```

4. (Optional) Add an upstream NTP server that the device will use to synchronize its time to the appropriate location in the list of NTP servers. The default setting is **time.devicecloud.com**.
   - To delete the default NTP server, **time.devicecloud.com**:

```
(config)> del service ntp server 0
(config)>
```

   - To add the NTP server to the beginning of the list, use the index value of **0** to indicate that it should be added as the first server:

```
(config)> add service ntp server 0 time.server.com
(config)>
```

   - To add the NTP server to the end of the list, use the index keyword **end**:

```
(config)> add service ntp server end time.server.com
(config)>
```

   - To add the NTP server in another location in the list, use an index value to indicate the appropriate position. For example:

```
(config)> add service ntp server 1 time.server.com
(config)>
```

Note This list is synchronized with the list of servers included with NTP server configuration, and changes made to one will be reflected in the other. See Configure the device as an NTP server for more information about NTP server configuration.

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Test the connection to the NTP servers

The following procedure tests the configured NTP servers for connectivity. This test does not affect the device's current local date and time.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Test the configured NTP servers for connectivity:

```
> system time test
Testing NTP server time.devicecloud.com on UDP port 123...
server 52.2.40.158, stratum 2, offset -0.000216, delay 0.05800
server 35.164.164.69, stratum 2, offset -0.000991, delay 0.07188
24 Aug 22:01:20 ntpdate[28496]: adjust time server 52.2.40.158 offset -
0.000216 sec
NTP test sync successful

Testing NTP server time.accns.com on UDP port 123...
server 128.136.167.120, stratum 3, offset -0.001671, delay 0.08455
24 Aug 22:01:20 ntpdate[28497]: adjust time server 128.136.167.120 offset
-0.001671 sec
NTP test sync successful
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Manually synchronize with the NTP server

The following procedure perform a NTP query to the configured servers and set the local time to the first server that responds.

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Synchronize the device's local date and time:

```
> system time synch
24 Aug 22:03:55 ntpdate[2520]: step time server 52.2.40.158 offset -
0.000487 sec
NTP sync to time.devicecloud.com successful
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Manually set the system date and time

If your network restricts access to NTP servers, use this procedure to set the local date and time.

This procedure is available at the Admin CLI only.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Set the device's local date and time:

```
> system time set value
>
```

   where *value* is the The date in year-month-day hour:minute:second format. For example:

```
> system time set 2022-03-03 10:16:23
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Network Time Protocol

Network Time Protocol (NTP) enables devices connected on local and worldwide networks to synchronize their internal software and hardware clocks to the same time source. The IX15 device can be configured as an NTP server, allowing downstream hosts that are attached to the device's Local Area Networks to synchronize with the device.

When the device is configured as an NTP server, it also functions as an NTP client. The NTP client will be consistently synchronized with one or more upstream NTP servers, which means that NTP packets are transferred every few seconds. A minimum of one upstream NTP server is required. Additional NTP servers can be configured. If multiple servers are configured, a number of time samples are obtained from each of the servers and a subset of the NTP clock filter and selection algorithms are applied to select the best of these.

See Configure the device as an NTP server for information about configuring your device as an NTP server.

Configure the device as an NTP server

Required Configuration Items

- Enable the NTP service.
- At least one upstream NTP server for synchronization. The default setting is the Digi NTP server, **time.devicecloud.com**.

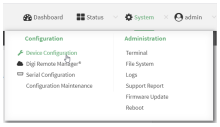
Additional Configuration Options

- Additional upstream NTP servers.
- Access control list to limit downstream access to the IX15 device's NTP service.
- The time zone setting, if the default setting of UTC is not appropriate.

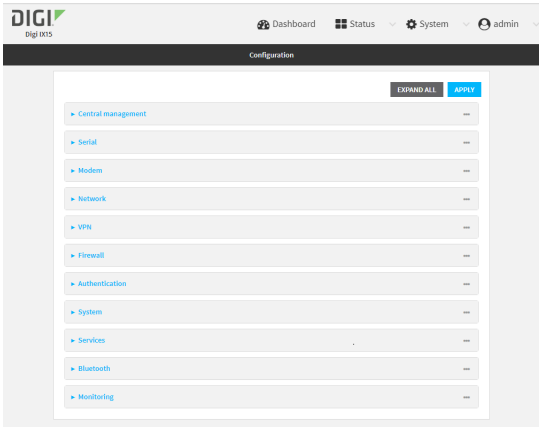
To configure the IX15 device's NTP service:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **NTP**.
4. Enable the IX15 device's NTP service by clicking **Enable**.
5. (Optional) Configure the access control list to limit downstream access to the IX15 device's NTP service.
   - To limit access to specified IPv4 addresses and networks:
     a. Click **IPv4 Addresses**.
     b. For **Add Address**, click ✚.

c. For **Address**, enter the IPv4 address or network that can access the device's NTP service. Allowed values are:
   - A single IP address or host name.
   - A network designation in CIDR notation, for example, 192.168.1.0/24.
   - **any**: No limit to IPv4 addresses that can access the NTP service.

d. Click ✚ again to list additional IP addresses or networks.

- To limit access to specified IPv6 addresses and networks:
  a. Click **IPv6 Addresses**.
  b. For **Add Address**, click ✚.
  c. For **Address**, enter the IPv6 address or network that can access the device's NTP service. Allowed values are:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 2001:db8::/48.
     - **any**: No limit to IPv6 addresses that can access the NTP service.
  d. Click ✚ again to list additional IP addresses or networks.
- To limit access to hosts connected through a specified interface on the IX15 device:
  a. Click **Interfaces**.
  b. For **Add Interface**, click ✚.
  c. For **Interface**, select the appropriate interface from the dropdown.
  d. Click ✚ again to allow access through additional interfaces.
- To limit access based on firewall zones:
  a. Click **Zones**.
  b. For **Add Zone**, click ✚.
  c. For **Zone**, select the appropriate firewall zone from the dropdown.
     See Firewall configuration for information about firewall zones.
  d. Click ✚ again to allow access through additional firewall zones.

Note By default, the access control list for the NTP service is empty, which means that all downstream hosts connected to the IX15 device can use the NTP service.

6. Enable **Fall back to local clock** to allow the device's local system clock to be used as backup time source.
7. (Optional) Add upstream NTP servers that the device will use to synchronize its time. The default setting is **time.devicecloud.com**.
   - To change the default value of the NTP server:
     a. Click **NTP servers**.
     b. For **Server**, type a new server name.
   - To add an NTP server:
     a. Click **NTP servers**.
     b. For **Add Server**, click ✚.
     c. For **Server**, enter the hostname of the upstream NTP server that the device will use to synchronize its time.

d. Click ✚ to add additional NTP servers. If multiple servers are included, servers are tried in the order listed until one succeeds.

Note This list is synchronized with the list of servers included with NTP client configuration, and changes made to one will be reflected in the other. See Configure the system time for more information about NTP client configuration.

8. (Optional) Configure the system time zone. The default is **UTC**.
   a. Click **System** > **Time**
   b. Select the **Timezone** for the location of your IX15 device.
9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable the ntp service:

```
(config)> service ntp enable true
(config)>
```

4. (Optional) Add an upstream NTP server that the device will use to synchronize its time to the appropriate location in the list of NTP servers. The default setting is **time.devicecloud.com**.
   - To delete the default NTP server, **time.devicecloud.com**:

```
(config)> del service ntp server 0
(config)>
```

   - To add the NTP server to the beginning of the list, use the index value of **0** to indicate that it should be added as the first server:

```
(config)> add service ntp server 0 time.server.com
(config)>
```

   - To add the NTP server to the end of the list, use the index keyword **end**:

```
(config)> add service ntp server end time.server.com
(config)>
```

   - To add the NTP server in another location in the list, use an index value to indicate the appropriate position. For example:

```
(config)> add service ntp server 1 time.server.com
(config)>
```

Note This list is synchronized with the list of servers included with NTP client configuration, and changes made to one will be reflected in the other. See Configure the system time for more information about NTP client configuration.

5. Allow the device's local system clock to be used as backup time source:

```
(config)> service ntp local true
(config)>
```

6. (Optional) Configure the access control list to limit downstream access to the IX15 device's NTP service.
   - To limit access to specified IPv4 addresses and networks:

     ```
     (config)> add service ntp acl address end value
     (config)>
     ```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 192.168.1.0/24.
     - **any**: No limit to IPv4 addresses that can access the NTP server agent.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:

     ```
     (config)> add service ntp acl address6 end value
     (config)>
     ```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 2001:db8::/48.
     - **any**: No limit to IPv6 addresses that can access the NTP server agent.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to hosts connected through a specified interface on the IX15 device:

     ```
     (config)> add service ntp acl interface end value
     (config)>
     ```

     Where *value* is an interface defined on your device.

     Display a list of available interfaces:

     Use **... network interface ?** to display interface information:

     ```
     (config)> ... network interface ?

     Interfaces
     ```

```
Additional Configuration
-----------------------------------------
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config)> add service ntp acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.

 Additional Configuration
 ------------------------------------------------------------
-----------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

(config)>
```

Repeat this step to list additional firewall zones.

Note By default, the access control list for the NTP service is empty, which means that all downstream hosts connected to the IX15 device can use the NTP service.

7. (Optional) Set the timezone for the location of your IX15 device. The default is **UTC**.

```
(config)> system time timezone value
(config)>
```

Where *value* is the timezone using the format specified with the following command:

```
(config)> system time timezone ?

Timezone: The timezone for the location of this device. This is used to
adjust the time for log
messages. It also affects actions that occur at a specific time of day.
Format:
  Africa/Abidjan
  Africa/Accra
  Africa/Addis_Ababa
  ...

(config)>
```

8. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show status and statistics of the NTP server

You can display status and statistics for active NTP servers

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **Status**.
3. Under **Services**, click **NTP**.

   The NTP server status page is displayed.

Command line

Show NTP information

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Use the show ntp command at the system prompt:

```
> show ntp

 NTP Status Status
 -----------------
```

```
 Status      : Up
 Sync Status : Up

 Remote            Refid          ST  T  When  Poll  Reach  Delay
Offset  Jitter
 ----------------  -------------  --  -  ----  ----  -----  ------  -----
-  ------
 *ec2-52-2-40-158  129.6.15.32    2   u  191   1024  377    33.570
+1.561  0.991
 128.136.167.120   128.227.205.3  3   u  153   1024  1      43.583  -
1.895  0.382

>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

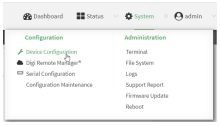
Configure a multicast route

Multicast routing allows a device to transmit data to a single multicast address, which is then distributed to a group of devices that are configured to be members of that group.

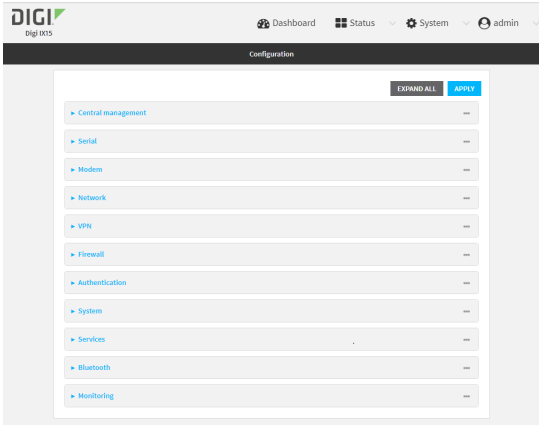
To configure a multicast route:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **Multicast**.

4. For **Add Multicast route**, type a name for the route and click ✚.
5. The new route is enabled by default. To disable, uncheck **Enable**.
6. Type the **Source address** for the route. This must be a multicast IP address between 224.0.0.1 and 239.255.255.255.
7. Select a **Source interface** where multicast packets will arrive.
8. To add one or more destination interface that the IX15 device will send mutlicast packets to:
   a. Click to expand **Destination interfaces**.
   b. Click ✚.
   c. For **Destination interface**, select the interface.
   d. Repeat for additional destination interfaces.
9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add the multicast route. For example, to add a route named **test**:

```
(config)> add service multicast test
(config service multicast test)>
```

4. The multicast route is enabled by default. If it has been disabled, enable the route:

```
(config service multicast test)> enable true
(config service multicast test)>
```

5. Set the source address for the route. This must be a multicast IP address between 224.0.0.1 and 239.255.255.255.

```
(config service multicast test)> dst ip-address
(config service multicast test)>
```

6. Set the source interface for the route where multicast packets will arrive:
   a. Use the **?** to determine available interfaces:
   b. Set the interface. For example:

```
(config service multicast test)> src_interface /network/interface/eth1
(config service multicast test)>
```

7. Set a destination interface that the IX15 device will send mutlicast packets to:
   a. Use the **?** to determine available interfaces:
   b. Set the interface. For example:

```
(config service multicast test)> add interface end
/network/interface/eth1
(config service multicast test)>
```

   c. Repeat for each additional destination interface.
8. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

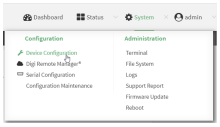
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Enable service discovery (mDNS)

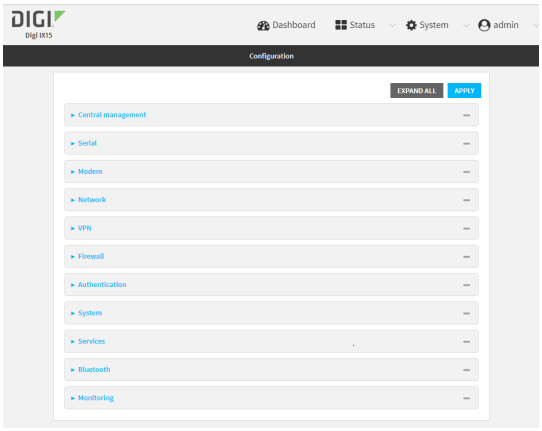
Multicast DNS mDNS is a protocol that resolves host names in small networks that do not have a DNS server. You can enable the IX15 device to use mDNS.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **Service Discovery (mDNS)**.
4. **Enable** the mDNS service.
5. Click **Access control list** to configure access control:
   - To limit access to specified IPv4 addresses and networks:
     a. Click **IPv4 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv4 address or network that can access the device's mDNS service. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 192.168.1.0/24.
        - **any**: No limit to IPv4 addresses that can access the mDNS service.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:
     a. Click **IPv6 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv6 address or network that can access the device's mDNS service. Allowed values are:

- A single IP address or host name.
- A network designation in CIDR notation, for example, 2001:db8::/48.
- **any**: No limit to IPv6 addresses that can access the mDNS service.

d. Click ✚ again to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX15 device:
  a. Click **Interfaces**.
  b. For **Add Interface**, click ✚.
  c. For **Interface**, select the appropriate interface from the dropdown.
  d. Click ✚ again to allow access through additional interfaces.
- To limit access based on firewall zones:
  a. Click **Zones**.
  b. For **Add Zone**, click ✚.
  c. For **Zone**, select the appropriate firewall zone from the dropdown.
  See Firewall configuration for information about firewall zones.
  d. Click ✚ again to allow access through additional firewall zones.

6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable the mDNS service:

```
(config)> service mdns enable true
(config)>
```

4. Configure access control:
   - To limit access to specified IPv4 addresses and networks:

```
(config)> add service mdns acl address end value
(config)>
```

   Where *value* can be:

- A single IP address or host name.
- A network designation in CIDR notation, for example, 192.168.1.0/24.
- **any**: No limit to IPv4 addresses that can access the mDNS service.

Repeat this step to list additional IP addresses or networks.

- To limit access to specified IPv6 addresses and networks:

```
(config)> add service mdns acl address6 end value
(config)>
```

Where *value* can be:

- A single IP address or host name.
- A network designation in CIDR notation, for example, 2001:db8::/48.
- **any**: No limit to IPv6 addresses that can access the mDNS service.

Repeat this step to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX15 device:

```
(config)> add service mdns acl interface end value
(config)>
```

Where *value* is an interface defined on your device.

Display a list of available interfaces:

Use **... network interface ?** to display interface information:

```
(config)> ... network interface ?

Interfaces

Additional Configuration
-----------------------------------------
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config)> add service mdns acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.

 Additional Configuration
 ------------------------------------------------------------
-----------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

(config)>
```

Repeat this step to list additional firewall zones.

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Use the iPerf service

Your IX15 device includes an iPerf3 server that you can use to test the performance of your network.

iPerf3 is a command-line tool that measures the maximum network throughput an interface can handle. This is useful when diagnosing network speed issues, to determine, for example, whether a cellular connection is providing expected throughput.

The IX15 implementation of iPerf3 supports testing with both TCP and UDP.

Note Using iPerf clients that are at a version earlier than iPerf3 to connect to the IX15 device's iPerf3 server may result in unpredictable results. As a result, Digi recommends using an iPerf client at version 3 or newer to connect to the IX15 device's iPerf3 server.

Required configuration items

- Enable the iPerf server on the IX15 device.
- An iPerf3 client installed on a remote host. iPerf3 software can be downloaded at https://iperf.fr/iperf-download.php.

Additional configuration Items

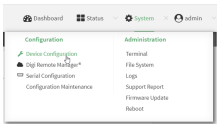
- The port that the IX15 device's iPerf server will use to listen for incoming connections.
- The access control list for the iPerf server.

  When the iPerf server is enabled, the IX15 device will automatically configure its firewall rules to allow incoming connections on the configured listening port. You can restrict access by configuring the access control list for the iPerf server.

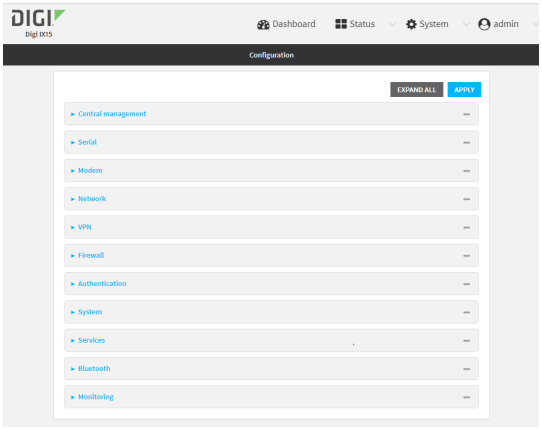
To enable the iPerf3 server:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



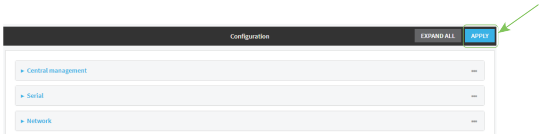
3. Click **Services** > **iPerf**.
4. Click **Enable**.
5. (Optional) For **IPerf Server Port**, type the appropriate port number for the iPerf server listening port.
6. (Optional) Click to expand **Access control list** to restrict access to the iPerf server:
   - To limit access to specified IPv4 addresses and networks:
     a. Click **IPv4 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv4 address or network that can access the device's iperf service. Allowed values are:
        - A single IP address or host name.
        - A network designation in CIDR notation, for example, 192.168.1.0/24.
        - **any**: No limit to IPv4 addresses that can access the iperf service.
     d. Click ✚ again to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:
     a. Click **IPv6 Addresses**.
     b. For **Add Address**, click ✚.
     c. For **Address**, enter the IPv6 address or network that can access the device's iperf service. Allowed values are:

- A single IP address or host name.
- A network designation in CIDR notation, for example, 2001:db8::/48.
- **any**: No limit to IPv6 addresses that can access the iperf service.

d. Click ✚ again to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX15 device:
  a. Click **Interfaces**.
  b. For **Add Interface**, click ✚.
  c. For **Interface**, select the appropriate interface from the dropdown.
  d. Click ✚ again to allow access through additional interfaces.
- To limit access based on firewall zones:
  a. Click **Zones**.
  b. For **Add Zone**, click ✚.
  c. For **Zone**, select the appropriate firewall zone from the dropdown.
  See Firewall configuration for information about firewall zones.
  d. Click ✚ again to allow access through additional firewall zones.

7. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable the iPerf server:

```
(config)> service iperf enable true
(config)>
```

4. (Optional) Set the port number for the iPerf server listening port. The default is 5201.

```
(config)> service iperf port port_number
(config)>
```

5. (Optional) Set the access control list to restrict access to the iPerf server:
   - To limit access to specified IPv4 addresses and networks:

```
(config)> add service iperf acl address end value
(config)>
```

Where *value* can be:

- A single IP address or host name.
- A network designation in CIDR notation, for example, 192.168.1.0/24.
- **any**: No limit to IPv4 addresses that can access the service-type.

Repeat this step to list additional IP addresses or networks.

- To limit access to specified IPv6 addresses and networks:

```
(config)> add service iperf acl address6 end value
(config)>
```

Where *value* can be:

- A single IP address or host name.
- A network designation in CIDR notation, for example, 2001:db8::/48.
- **any**: No limit to IPv6 addresses that can access the service-type.

Repeat this step to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX15 device:

```
(config)> add service iperf acl interface end value
(config)>
```

Where *value* is an interface defined on your device.

Display a list of available interfaces:

Use **... network interface ?** to display interface information:

```
(config)> ... network interface ?

Interfaces

Additional Configuration
-----------------------------------------
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config)> add service iperf acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.

 Additional Configuration
 ------------------------------------------------------------
-----------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

(config)>
```

Repeat this step to list additional firewall zones.

6. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Example performance test using iPerf3

On a remote host with iPerf3 installed, enter the following command:

```
$ iperf3 -c device_ip
```

where *device_ip* is the IP address of the IX15 device. For example:

```
$ iperf3 -c 192.168.2.1
Connecting to host 192.168.2.1, port 5201
[  4] local 192.168.3.100 port 54934 connected to 192.168.1.1 port 5201
[ ID] Interval           Transfer     Bandwidth       Retr  Cwnd
[  4]   0.00-1.00   sec  26.7 MBytes   224 Mbits/sec    8   2.68 MBytes
[  4]   1.00-2.00   sec  28.4 MBytes   238 Mbits/sec   29   1.39 MBytes
[  4]   2.00-3.00   sec  29.8 MBytes   250 Mbits/sec    0   1.46 MBytes
[  4]   3.00-4.00   sec  31.2 MBytes   262 Mbits/sec    0   1.52 MBytes
[  4]   4.00-5.00   sec  32.1 MBytes   269 Mbits/sec    0   1.56 MBytes
[  4]   5.00-6.00   sec  32.5 MBytes   273 Mbits/sec    0   1.58 MBytes
[  4]   6.00-7.00   sec  33.9 MBytes   284 Mbits/sec    0   1.60 MBytes
[  4]   7.00-8.00   sec  33.7 MBytes   282 Mbits/sec    0   1.60 MBytes
[  4]   8.00-9.00   sec  33.5 MBytes   281 Mbits/sec    0   1.60 MBytes
```

```
[  4]   9.00-10.00  sec  33.2 MBytes   279 Mbits/sec    0   1.60 MBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[  4]   0.00-10.00  sec   315 MBytes   264 Mbits/sec   37             sender
[  4]   0.00-10.00  sec   313 MBytes   262 Mbits/sec                  receiver

iperf Done.
$
```

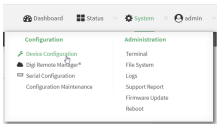
Configure the ping responder service

Your IX15 device's ping responder service replies to ICMP and ICMPv6 echo requests. The service is enabled by default. You can disable the service, or you can configure the service to use an access control list to limit the service to specified IP address, interfaces, and/or zones.

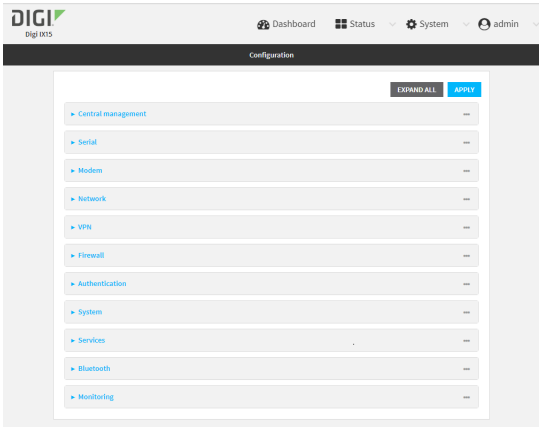
To enable the iPerf3 server:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Services** > **Ping responder**.

   The ping responder service is enabled by default. Click **Enable** to disable all ping responses.
4. Click to expand **Access control list** to restrict ping responses to specified IP address, interfaces, and/or zones:
   - To limit access to specified IPv4 addresses and networks:
     a. Click **IPv4 Addresses**.
     b. For **Add Address**, click ✚.

c. For **Address**, enter the IPv4 address or network that can access the device's ping responder. Allowed values are:
   - A single IP address or host name.
   - A network designation in CIDR notation, for example, 192.168.1.0/24.
   - **any**: No limit to IPv4 addresses that can access the ping responder.

d. Click ✚ again to list additional IP addresses or networks.

- To limit access to specified IPv6 addresses and networks:
  a. Click **IPv6 Addresses**.
  b. For **Add Address**, click ✚.
  c. For **Address**, enter the IPv6 address or network that can access the device's ping responder. Allowed values are:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 2001:db8::/48.
     - **any**: No limit to IPv6 addresses that can access the ping responder.
  d. Click ✚ again to list additional IP addresses or networks.
- To limit access to hosts connected through a specified interface on the IX15 device:
  a. Click **Interfaces**.
  b. For **Add Interface**, click ✚.
  c. For **Interface**, select the appropriate interface from the dropdown.
  d. Click ✚ again to allow access through additional interfaces.
- To limit access based on firewall zones:
  a. Click **Zones**.
  b. For **Add Zone**, click ✚.
  c. For **Zone**, select the appropriate firewall zone from the dropdown.
     See Firewall configuration for information about firewall zones.
  d. Click ✚ again to allow access through additional firewall zones.

5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable the iPerf server:

```
(config)> service iperf enable true
(config)>
```

4. (Optional) Set the port number for the iPerf server listening port. The default is 5201.

```
(config)> service iperf port port_number
(config)>
```

5. (Optional) Set the access control list to restrict access to the iPerf server:
   - To limit access to specified IPv4 addresses and networks:

     ```
     (config)> add service iperf acl address end value
     (config)>
     ```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 192.168.1.0/24.
     - **any**: No limit to IPv4 addresses that can access the service-type.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to specified IPv6 addresses and networks:

     ```
     (config)> add service iperf acl address6 end value
     (config)>
     ```

     Where *value* can be:
     - A single IP address or host name.
     - A network designation in CIDR notation, for example, 2001:db8::/48.
     - **any**: No limit to IPv6 addresses that can access the service-type.

     Repeat this step to list additional IP addresses or networks.
   - To limit access to hosts connected through a specified interface on the IX15 device:

     ```
     (config)> add service iperf acl interface end value
     (config)>
     ```

     Where *value* is an interface defined on your device.

     Display a list of available interfaces:

     Use **... network interface ?** to display interface information:

     ```
     (config)> ... network interface ?

     Interfaces

     Additional Configuration
     ------------------------------------------
      defaultip              Default IP
      defaultlinklocal       Default Link-local IP
     ```

```
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config)> add service iperf acl zone end value
```

Where *value* is a firewall zone defined on your device, or the **any** keyword.

Display a list of available firewall zones:

Type **... firewall zone ?** at the config prompt:

```
(config)> ... firewall zone ?

Zones: A list of groups of network interfaces that can be
referred to by packet
filtering rules and access control lists.

 Additional Configuration
 ------------------------------------------------------------
-----------------------
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup

(config)>
```

Repeat this step to list additional firewall zones.

6. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Example performance test using iPerf3

On a remote host with Iperf3 installed, enter the following command:

```
$ iperf3 -c device_ip
```

where *device_ip* is the IP address of the IX15 device. For example:

```
$ iperf3 -c 192.168.2.1
Connecting to host 192.168.2.1, port 5201
[  4] local 192.168.3.100 port 54934 connected to 192.168.1.1 port 5201
[ ID] Interval           Transfer     Bandwidth       Retr  Cwnd
[  4]   0.00-1.00   sec  26.7 MBytes   224 Mbits/sec    8   2.68 MBytes
[  4]   1.00-2.00   sec  28.4 MBytes   238 Mbits/sec   29   1.39 MBytes
[  4]   2.00-3.00   sec  29.8 MBytes   250 Mbits/sec    0   1.46 MBytes
[  4]   3.00-4.00   sec  31.2 MBytes   262 Mbits/sec    0   1.52 MBytes
[  4]   4.00-5.00   sec  32.1 MBytes   269 Mbits/sec    0   1.56 MBytes
[  4]   5.00-6.00   sec  32.5 MBytes   273 Mbits/sec    0   1.58 MBytes
[  4]   6.00-7.00   sec  33.9 MBytes   284 Mbits/sec    0   1.60 MBytes
[  4]   7.00-8.00   sec  33.7 MBytes   282 Mbits/sec    0   1.60 MBytes
[  4]   8.00-9.00   sec  33.5 MBytes   281 Mbits/sec    0   1.60 MBytes
[  4]   9.00-10.00  sec  33.2 MBytes   279 Mbits/sec    0   1.60 MBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[  4]   0.00-10.00  sec   315 MBytes   264 Mbits/sec   37             sender
[  4]   0.00-10.00  sec   313 MBytes   262 Mbits/sec                  receiver

iperf Done.
$
```

Applications

The IX15 supports Python 3.6 and provides you with the ability to run Python applications on the device interactively or from a file. You can also specify Python applications and other scripts to be run each time the device system restarts, at specific intervals, or at a specified time.

This chapter contains the following topics:

Develop Python applications 541
Set up the IX15 to automatically run your applications 580
Start an interactive Python session 589
Run a Python application at the shell prompt 590
Install third party Python modules 592
Python migration guide 592

Develop Python applications

The IX15 features a standard Python 3.6 distribution. Python is a dynamic, object-oriented language for developing software applications, from simple programs to complex embedded applications. Digi offers the Digi IoT PyCharm Plugin to help you while writing, building, and testing your application. It also provides examples to use as the base for programming your IX15. See Create and test a Python application.

In addition to the standard Python library, the IX15 includes a set of extensions to access its configuration and interfaces. See Python modules.

The IX15 provides you with the ability to:

- Run Python applications on the device interactively or from a file.
- Specify Python applications and other scripts to be run each time the device system restarts, at specific intervals, or at a specified time. See Configure scripts to run automatically.

This section contains the following topics:

Set up the IX15 for Python development
Create and test a Python application
End-to-end demos
Python modules

Set up the IX15 for Python development

1. Access the IX15 local web interface
   a. Use an Ethernet cable to connect the IX15 to your local laptop or PC.
   The factory default IP address is **192.168.2.1**
   b. Log into the IX15 WebUI as a user with full admin access rights.
   The default user name is **admin** and the default password is the unique password printed on the label packaged with your device.
2. Go to the Configuration window
   a. On the menu, click **System**.
   b. Under **Configuration**, click **Device Configuration**. The Configuration window displays.
3. (Optional) If you want to connect to a local network (LAN) that has a DHCP server
   a. Click **Network** > **Interfaces** > **LAN** > **IPv4**.
   b. Select **DHCP address**.

   For LAN configuration, see the following topics in the *IX14 User Guide*:
   - Change the default LAN subnet
   - Change the LAN address type
   - Allow remote access for web administration and SSH
4. Enable service discovery (mDNS)
   a. Click **Services** > **Service Discovery (mDNS)**.
   b. **Enable** the mDNS service.

Note For more information, see Enable service discovery (mDNS).

5. Configure SSH access
   a. Click **Services** > **SSH**.
   b. Click **Enable**.

Note For more information, see the following topics: Configure SSH access, Use SSH with key authentication, and Allow remote access for web administration and SSH.

6. Enable shell access
   a. Click **Authentication** > **Groups** > **admin**.
   b. Click the **Interactive shell access** option.
   c. If this option is not displayed, see Disable shell access.
7. Click **Apply** to save the configuration and apply the changes.
   The **Apply** button is located at the top of the WebUI page. You may need to scroll to the top of the page to locate it.

Create and test a Python application

To develop a Python application for the IX15:

1. Set up the IX15 for Python development.
2. Create and test your application with:

- PyCharm along with Digi XBee PyCharm Plugin. You can create, build, and remotely launch your application in the IX15.
- Your preferred editor and manually transfer the application, install dependencies, and launch in the IX15.

Develop an application in PyCharm

The Digi IoT PyCharm Plugin allows you to write, build and run Python applications for Digi devices in a quick and easy way. See the Digi XBee PyCharm IDE Plugin User Guide for details.

This is what you can do with it:

- Create Python projects from scratch or import one of the available examples.
- Get help while you write your code thanks to the syntax highlight, quick documentation, and code completion features.
- Build and upload Python applications to your Digi device with just one click.
- Add libraries that facilitate the usage of external peripherals or non-standard APIs.
- Communicate with your Digi device through the integrated SSH console to see the application output or execute quick tests.

Manually install and launch an application

To create, build, and launch your application:

1. Write your Python application code. Code can include:

- Any Python 3.6 standard feature.
- Access to the IX15 configuration and hardware with the Python modules.
- Third-party modules included in the IX15:

- pySerial 3.4
- PyModbus 2.3
- Eclipse Paho MQTT Python Client

- Any other third-party module implemented in Python.

2. Generate a wheel (whl) package or a zip package.
   a. If generating a wheel package, see https://packaging.python.org/tutorials/packaging-projects/.
   b. If generating a zip package, the zip must contain a folder with the name of your application and optionally a **requirements.txt** file.

      Example:
      MyDemoFolder

      |---- main.py

      |---- requirements.txt
3. Upload your package to **/tmp**, for instance to **/tmp/MyDemoFolder.zip**.

   See Upload and download files.
4. Install the package. See pyinstall.

   Example:

   > pyinstall package **/tmp/MyDemoFolder.zip**
5. The application is installed in **/etc/config/scripts** directory. Launch your application:
   a. Run your application at the shell prompt.
   b. Configure your application to run automatically.

End-to-end demos

The Digi XBee PyCharm Plugin includes a set of sample applications ready to build and execute in your IX15. You can use these sample applications as a reference to create your own Python application or start developing one from scratch:

- End-to-end sample application

  A good starting point of a complete application to remotely manage XBee networks with a IX15:

  - IX15 Gateway Python application:

  a. Retrieves information—temperature and humidity—from the XBee nodes in the network to upload to Digi Remote Manager.
  b. Receives instructions from Digi Remote Manager to remotely command the same nodes—change a node sampling rate or stop its reporting service.

  - XBee 3 nodes MicroPython application:

  a. Transmits read data—temperature and humidity—to the IX15 and sleeps the configured period of time.

b. Processes incoming commands from the IX15—change the sampling rate or stop the reporting service.

- XBee IoT Smart Agriculture Demo

  A complete example that emulates the management of a smart irrigation system using Digi IoT devices and services. It demonstrates how to use your Digi IX15 Gateway and XBee3 modules to exchange data, communicate with Digi Remote Manager, and use the Bluetooth Low Energy interface to talk to mobile apps applied to the agriculture vertical.

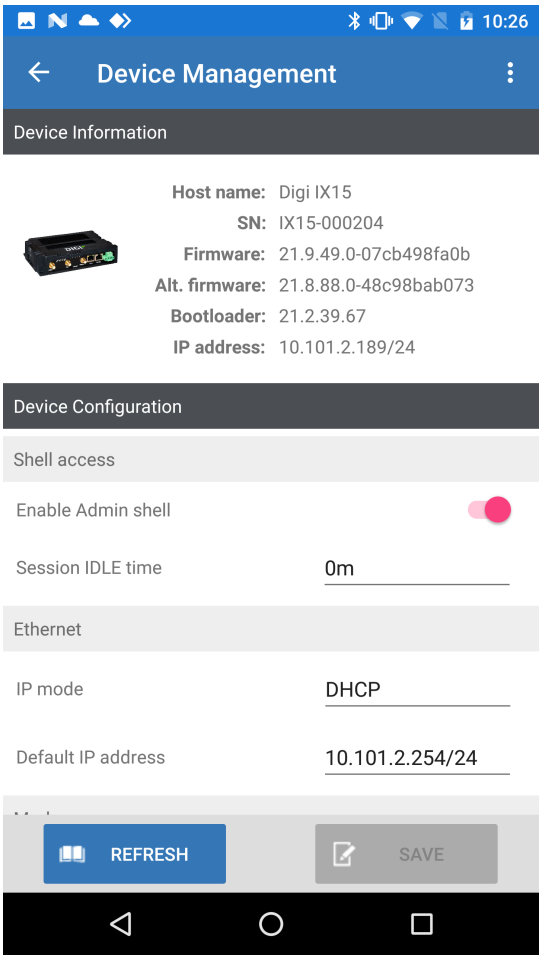
- Network Update from Digi Remote Manager

  An application to update the XBee profile of one or several nodes of your network using Digi Remote Manager. It registers a callback to process device requests coming from Digi Remote Manager to trigger the profile update tasks.

- IX15 BLE Configuration demo

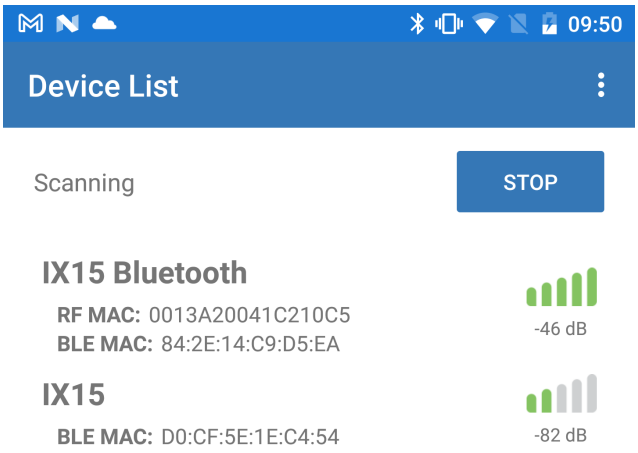
  A demo application for IX15 configuration over Bluetooth Low Energy.

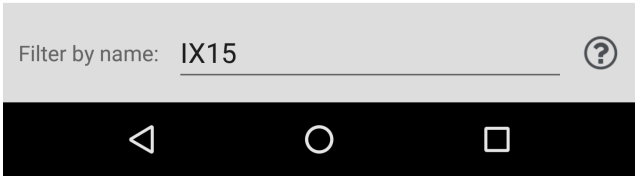
  For more information, see the application README.



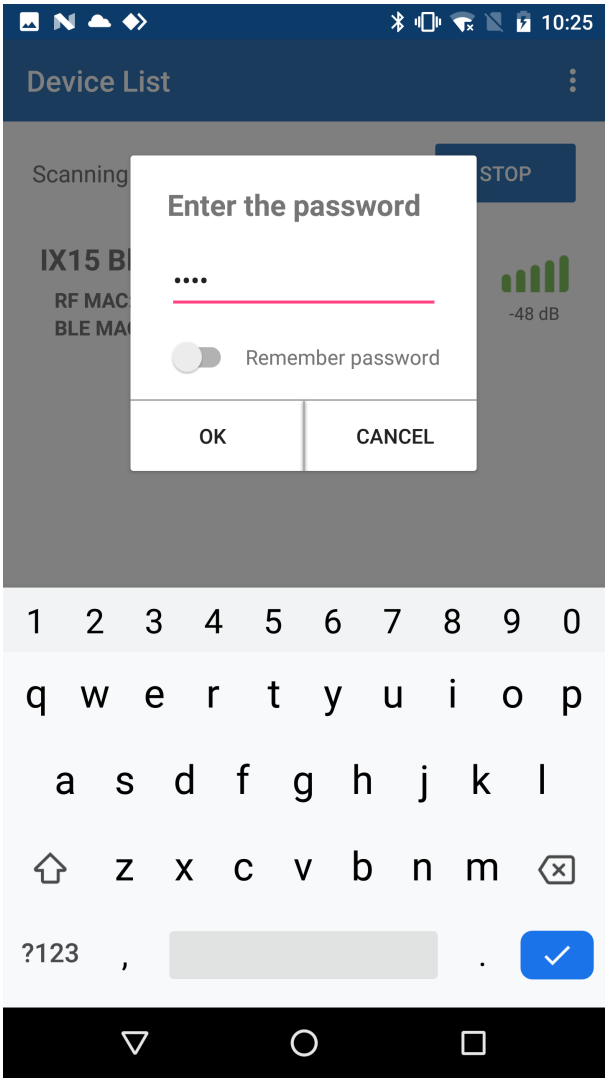
1. Enable CLI over bluetooth. See Configure Bluetooth Low Energy if you need help.
2. Install the **IX15BleConfiguratorDemo** on your mobile phone.
3. Open the app and scan for devices. Your IX15 device will appear in the list.

TIP: You can use **filter by name** if your list is big and you have problems identifying your IX15.

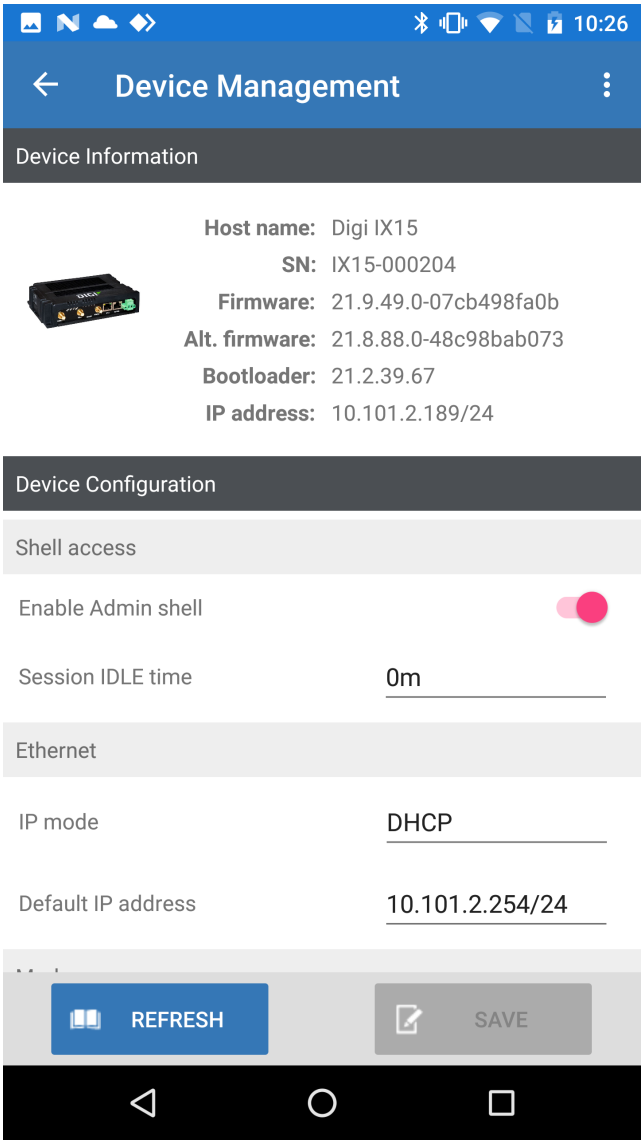




4. Click the IX15 device and enter the device Bluetooth Low Energy password.



5. After passing the password validation, you can see your device information and change some basic configuration settings.



Python modules

The IX15 supports Python 3.6 and provides you with the ability to run Python applications on the device interactively or from a file. It also offers extensions to manage your IX15:

- The **digidevice** module provides platform-specific extensions that allow you to interact with the device's configuration and interfaces.
  The following submodules are included with the **digidevice** module:
  - XBee: **digidevice.xbee**
  - LEDs: **digidevice.led**
  - SMS: **digidevice.sms**
  - GPS: **digidevice.location**

    - Digi Remote Manager:
        - **digidevice.datapoint**
        - **digidevice.device_request**
        - **digidevice.name**
    - Device configuration: **digidevice.config**
    - Command line interface: **digidevice.cli**
    - Access runtime database: **digidevice.runt**
- The XBee Python Library is also integrated so you can work with the local and remote XBee devices in your network.
- Use the Python **serial** module—pySerial—to access the serial ports.
- PyModbus a full Modbus protocol implementation.
- Eclipse Paho MQTT Python client enables applications to connect to an MQTT broker to publish messages, and to subscribe to topics and receive published messages.

Note Module-related documentation is in the Digidevice module section.

Digidevice module

The Python **digidevice** module provides platform-specific extensions that allow you to interact with the device's configuration and interfaces. The following submodules are included with the **digidevice** module:

This section contains the following topics:

XBee API

The **digidevice.xbee** Python module offers the **get_device** function for accessing the internal XBee device of your IX15.

```
def get_device(timeout: int = -1) -> XBeeDevice
```

This function returns a representation of the XBee device in the IX15. This object class is defined in the XBee Python Library, and depends on the protocol your XBee is using:

- **XBeeDevice**: generic XBee device.
- **ZigBeeDevice**: ZigBee protocol-specific XBee device.
- **DigiMeshDevice**: DigiMesh protocol-specific XBee device.
- **Raw802Device**: 802.15.4 protocol-specific XBee device.

Example: Get local XBee instance

```
from digidevice import xbee

local_xbee = xbee.get_device()
```

Once the local XBee of the IX15 is retrieved, you can work with it using the XBee Python Library API that is integrated into the gateway firmware:

- Retrieve and discover the XBee nodes in your network.
- Send and receive data to or from other XBee devices in the network.
- Read and set the IO lines of remote nodes.
- And so forth

Note See the Digi XBee Python Library project online for additional documentation.

Bluetooth Low Energy API

XBee3 devices have the ability to send and receive data from the Bluetooth Low Energy interface of the local XBee device through User Data Relay frames. This can be useful if your IX15 application wants to transmit or receive data from a cellphone connected to it over BLE.

The XBee API of the gateway provides the required methods to communicate with the BLE interface.

Send Bluetooth data

The **XBeeDevice** class and its subclasses provide the following method to send data to the Bluetooth Low Energy interface:

Method	Description
send_bluetooth_data (Bytearray)	Specifies the data to send to the Bluetooth Low Energy interface.

This method is asynchronous, which means that your application does not block during the transmit process.

Send data to Bluetooth

```
from digidevice import xbee

[...]

# Instantiate the XBee device object.
device = xbee.get_device()
device.open()

data = "Bluetooth, are you there?"# Send the data to the Bluetooth interface.
device.send_bluetooth_data(data.encode("utf8"))

[...]
```

The **send_bluetooth_data** method may fail for the following reasons:

- Errors register as **XBeeException**:
  - If the operating mode of the device is not **API** or **ESCAPED_API_MODE**, the method throws an **InvalidOperatingModeException**.
  - If there is an error writing to the XBee interface, the method throws a generic **XBeeException**.

Receive Bluetooth data

You can be notified when new data from the Bluetooth Low Energy interface has been received if you are subscribed or registered to the Bluetooth data reception service by using the **add_bluetooth_data_received_callback** method.

Bluetooth data reception registration

```
from digidevice import xbee

[...]

# Instantiate the XBee device object.
device = xbee.get_device()
device.open()

# Define the callback.
def my_bluetooth_data_callback(data):
    print("Data received from the Bluetooth interface >> '%s'" % data.decode
("utf-8"))

# Add the callback.
device.add_bluetooth_data_received_callback(my_bluetooth_data_callback)

[...]
```

When a new data from the Bluetooth interface is received, your callback is executed providing the data in byte array format as parameter.

To stop listening to new data messages from the Bluetooth interface, use the **del_bluetooth_data_received_callback** method to unsubscribe the already-registered listener.

Deregister Bluetooth data reception

```
[...]

device = [...]
```

```
def my_bluetooth_data_callback(data):
    [...]

device.add_bluetooth_data_received_callback(my_bluetooth_data_callback)

[...]

# Delete the Bluetooth data callback.
device.del_bluetooth_data_received_callback(my_bluetooth_data_callback)

[...]
```

Use digidevice.cli to execute CLI commands

Use the **digidevice.cli** Python module to issue CLI commands from Python to retrieve status and statistical information about the device.

For example, to display the system status and statistics by using an interactive Python session, use the show system command with the **cli** module:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

   ```
   # python
   Python 3.6.13 (default, May  9 2021, 22:49:59)
   [GCC 8.3.0] on linux
   Type "help", "copyright", "credits" or "license" for more information.
   >>>
   ```

3. Import the **cli** submodule:

   ```
   >>> from digidevice import cli
   >>>
   ```

4. Execute a CLI command using the **cli.execute(*command*)** function. For example, to print the system status and statistics to stdout using the **show system** command:

   ```
   >>> response = cli.execute("show system")
   >>>
   >>> print (response)

     Model                    : Digi IX15
     Serial Number            : IX15-000065
     SKU                      : IX15
     Hostname                 : IX15
     MAC Address              : DF:DD:E2:AE:21:18

     Hardware Version         : 50001947-01 1P
     Firmware Version         : 22.2.9.85
     Alt. Firmware Version    : 22.2.9.85
     Alt. Firmware Build Date : Thurs, 03 March 2022 10:16:23
     Bootloader Version       : 19.7.23.0-15f936e0ed
   ```

```
  Current Time             : Thurs, 03 March 2022 10:16:23 +0000
  CPU                      : 1.4%
  Uptime                   : 6 days, 6 hours, 21 minutes, 57 seconds
(541317s)
  Temperature              : 40C

>>>
```

5. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Help for using Python to execute IX15 CLI commands

Get help executing a CLI command from Python by accessing help for **cli.execute**:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **cli** submodule:

```
>>> from digidevice import cli
>>>
```

4. Use the help command with **cli.execute**:

```
>>> help(cli.execute)
Help on function execute in module digidevice.cli:

execute(command, timeout=5)
Execute a CLI command with the timeout specified returning the results.
...
```

5. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Use digidevice.datapoint to upload custom datapoints to Digi Remote Manager

Use the **datapoint** Python module to upload custom datapoints to Digi Remote Manager.

The following characteristics can be defined for a datapoint:

- Stream ID
- Value
- (Optional) Data type
  - integer
  - long
  - float
  - double
  - string
  - binary
- Units (optional)
- Timestamp (optional)
- Location (optional)
  - Tuple of latitude, longitude and altitude

- Description (optional)
- Quality (optional)
  - An integer describing the quality of the data point

For example, to use an interactive Python session to upload datapoints related to velocity, temperature, and the state of the emergency door:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **datapoint** submodule and other necessary modules:

```
>>> from digidevice import datapoint
>>> import time
>>>
```

4. Upload the datapoints to Remote Manager:

```
>>> datapoint.upload("Velocity", 69, units="mph")
>>> datapoint.upload("Temperature", 24, geo_location=(54.409469, -
1.718836, 129))
>>> datapoint.upload("Emergency_Door", "closed", timestamp=time.time())
```

5. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

You can also upload multiple datapoints:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **datapoint** submodule and other necessary modules:

```
>>> from digidevice import datapoint
>>> import time
>>>
```

4. Create datapoint objects:

```
>>> p1 = datapoint.DataPoint("Velocity", 69, units="mph")
>>> p2 = datapoint.DataPoint("Temperature", 24, geo_location=(54.409469,
-1.718836, 129))
>>> p3 = datapoint.DataPoint("Emergency_Door", "closed",
timestamp=time.time())
>>>
```

5. Upload the datapoints to Remote Manager:

```
>>> datapoint.upload_multiple([p1, p2, p3])
>>>
```

6. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Once the datapoints have been uploaded to Remote Manager, they can be viewed via Remote Manager or accessed using Web Services calls. See the *Digi Remote Manager Programmers Guide* for more information on web services and datapoints.

Help for using Python to upload custom datapoints to Remote Manager

Get help for uploading datapoints to your Digi Remote Manager account by accessing help for **datapoint.upload** and **datapoint.upload_multiple**:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **datapoint** submodule and other necessary modules:

```
>>> from digidevice import datapoint
>>>
```

4. Use the help command with **datapoint.upload**:

```
>>> help(datapoint.upload)
Help on function upload in module digidevice.datapoint:

upload(stream_id:str, data, *, description:str=None,
timestamp:float=None, units:str=None,
```

```
geo_location:Tuple[float, float, float]=None, quality:int=None,
data_type:digidevice.datapoint.DataType=None, timeout:float=None)
...
```

5. Use the help command with **datapoint.upload_multiple**:

```
>>> help(datapoint.upload_multiple)
Help on function upload_multiple in module digidevice.datapoint:

upload_multiple(datapoints:List[digidevice.datapoint.DataPoint],
timeout:float=None)
...
```

6. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Use digidevice.config for device configuration

Use the **config** Python module to access and modify the device configuration.

Read the device configuration

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **config** submodule:

```
>>> from digidevice import config
>>>
```

4. Use **config.load()** and the **get()** method to return the device's configuration:
   a. Return the entire configuration:

```
>>> from pprint import pprint # use pprint vs. print to make the
output easier to read
>>> cfg = config.load()
>>> pprint(cfg.dump().splitlines())
```

   This returns the device configuration:

```
...
network.interface.lan1.device=/network/bridge/lan1
network.interface.lan1.enable=true
network.interface.lan1.ipv4.address=192.168.2.1/24
```

```
network.interface.lan1.ipv4.connection_monitor.attempts=3
...
```

b. Print a list of available interfaces:

```
>>> cfg = config.load()
>>> interfaces = cfg.get("network.interface")
>>> print(interfaces.keys())
```

This returns the following:

```
['defaultip', 'defaultlinklocal', 'lan1', 'loopback', 'wan1', 'wwan1',
'wwan2']
```

c. Print the IPv4 address of the LAN interface:

```
>>> cfg = config.load()
>>> interfaces = cfg.get(“network.interfaces”)
>>> print(interfaces.get("lan.ipv4.address"))
```

Which returns:

```
192.168.2.1/24
```

Modify the device configuration

Use the **set()** and **commit()** methods to modify the device configuration:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **config** submodule:

```
>>> from digidevice import config
>>>
```

4. Use **config.load(writable=True)** to enable write mode for the configuration:

```
>>> cfg = config.load(writable=True)
>>>
```

5. Use the **set()** method to make changes to the configuration:

```
>>> cfg.set("system.name", "New-Name")
>>>
```

6. Use the **commit()** method to save the changes:

```
>>> cfg.commit()
True
>>>
```

7. Use the **get()** method to verify the change:

```
>>> print(cfg.get("system.name"))
New-Name
>>>
```

Help for using Python to read and modify device configuration

Get help for reading and modifying the device configuration by accessing help for **digidevice.config**:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **config** submodule:

```
>>> from digidevice import config
>>>
```

4. Use the help command with **config**:

```
>>> help(config)
Help on module acl.config in acl:

NAME
acl.config - Python interface to ACL configuration (libconfig).
...
```

5. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Use Python to respond to Digi Remote Manager SCI requests

The **device_request** Python module allows you to interact with Digi Remote Manager by using Remote Manager's Server Command Interface (SCI), a web service that allows users to access information and perform commands that relate to their devices.

Use Remote Manager's SCI interface to create SCI requests that are sent to your IX15 device, and use the **device_request** module to send responses to those requests to Remote Manager.

See the *Digi Remote Manager Programmers Guide* for more information on SCI.

Task one: Use the device_request module on your IX15 device to create a response

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **device_request** module:

```
>>> from digidevice import device_request
>>>
```

4. Create a function to handle the request from Remote Manager:

```
>>> def handler(target, request):
     print ("received request %s for target %s" % (request, target))
     return "OK"
>>>
```

5. Register a callbackup function that will be called when the device receives a SCI request from Remote Manager:

```
>>> device_request.register("myTarget", handler)
>>>
```

Note Leave the interactive Python session active while completing task two, below. Once you have completed task two, exit the interactive session by using **Ctrl-D**. You can also exit the session using **exit()** or **quit()**.

Task two: Create and send an SCI request from Digi Remote Manager

The second step in using the **device_request** module is to create an SCI request that Remote Manager will forward to the device. For example, you can create in SCI request a the Remote Manager API explorer:

1. In Remote Manager, click **Documentation** > **API Explorer**.
2. Select the device to use as the SCI target:
   a. Click **SCI Targets**.
   b. Click **Add Targets**.
   c. Enter or select the device ID of the device.
   d. Click **Add**.
   e. Click **OK**.

3. Click **Examples** > **SCI** > **Data Service** > **Send Request**.

   Code similar to the following will be displayed in the HTTP message body text box:

```
<sci_request version="1.0">
  <data_service>
  <targets>
    <device id="00000000-00000000-0000FFFF-A83CF6A3"/>
  </targets>
    <requests>
      <device_request target_name="myTarget">
        my payload string
      </device_request>
    </requests>
  </data_service>
</sci_request>
```

   Note The value of the **target_name** parameter in the **device_request** element must correspond to the **target** parameter of the **device_request.register** function in the Python script. In this example, the two are the same.

4. Click **Send**.

   Once that the request has been sent to the device, the handler on the device is executed.

   - On the device, you will receive the following output:

```
>>> received request
       my payload string
      for target myTarget
>>>
```

   - In Remote Manager, you will receive a response similar to the following:

```
<sci_reply version="1.0">
  <data_service>
    <device id="00000000-00000000-0000FFFF-A83CF6A3"/>
      <requests>
        <device_request target_name="myTarget"
status="0">OK</device_request>
      </requests>
    </device>
  </data_service>
</sci_request>
```

Example: Use digidevice.cli with digidevice.device_request

In this example, we will use the **digidevice.cli** module in conjunction with the **digidevice.device_request** module to return information about multiple devices to Remote Manager.

1. Create a Python application, called showsystem.py, that uses the **digidevice.cli** module to create a response containing information about device and the **device_request** module to respond with this information to a request from Remote Manager:

```
from digidevice import device_request
from digidevice import cli
import time

def handler(target, request):
    return cli.execute("show system verbose")

def status_cb(error_code, error_description):
    if error_code != 0:
        print("error handling showSystem device request: %s" % error_
description)

device_request.register("showSystem", handler, status_callback = status_
cb)

# Do not let the process finish so that it handles device requests
while True:
    time.sleep(10)
```

2. Upload the showsystem.py application to the /etc/config/scripts directory on two or more Digi devices. In this example, we will upload it to two devices, and use the same request in Remote Manager to query both devices.

   See Configure scripts to run automatically for information about uploading Python applications to your device. You can also create the script on the device by using the **vi** command when logged in with shell access.

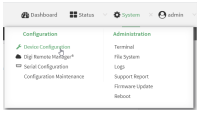
3. For both devices:

   a. Configure the device to automatically run the showsystem.py application on reboot, and to restart the application if it crashes. This can be done from either the WebUI or the command line:

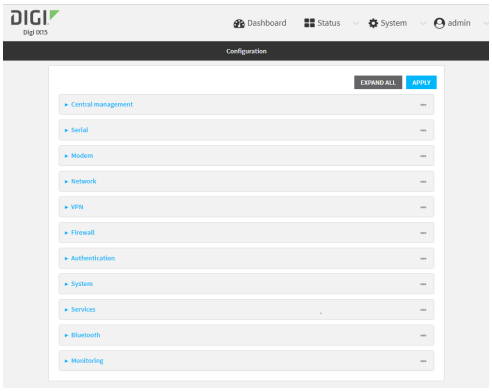
   ### WebUI

   i. Log into the IX15 WebUI as a user with full Admin access rights.

   ii. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

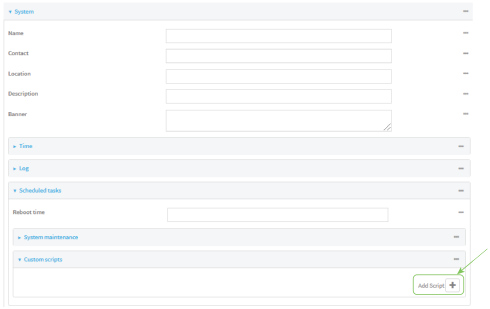
   

   The **Configuration** window is displayed.



iii. Click **System** > **Scheduled tasks** > **Custom scripts**.

iv. Click ✚ to add a custom script.

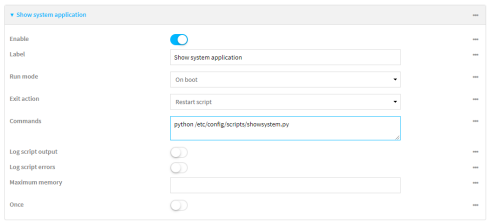


v. For **Label**, type **Show system application**.

vi. For **Run mode**, select **On boot**.

vii. For **Exit action**, select **Restart script**.

viii. For **Commands**, type **python /etc/config/scripts/showsystem.py**.



ix. Click **Apply** to save the configuration and apply the change.



Command line

i. Log into the IX15 command line as a user with full Admin access rights.

Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

ii. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

iii. Add an application entry:

```
(config)> add system schedule script end
(config system schedule script 0)>
```

Scheduled scripts are enabled by default. To disable:

```
(config system schedule script 0)> enable false
(config system schedule script 0)>
```

iv. Provide a label for the script:

```
(config system schedule script 0)> label "Show system application"
```

v. Configure the application to run automatically when the device reboots:

```
(config system schedule script 0)> when boot
(config system schedule script 0)>
```

vi. Configure the application to restart if it crashes:

```
(config system schedule script 0)> exit_action restart
(config system schedule script 0)>
```

vii. Set the command that will execute the application:

```
(config system schedule script 0)> commands "python
/etc/config/scripts/showsystem.py"
(config system schedule script 0)>
```

viii. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

b. Run the showsystem.py application. You can run the application by either rebooting the device, or by running it from the shell prompt.

- To reboot the device:
  i. From the WebUI:
     i. From the main menu, click **System**.
     ii. Click **Reboot**.

i. From the command line, at the Admin CLI prompt, type:

```
> reboot
```

- To run the application from the shell prompt:
  i. Log into the IX15 command line as a user with shell access.

     Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

  ii. Type the following at the shell prompt:

```
# python /etc/config/scripts/showsystem.py &
#
```

  iii. Exit the shell:

```
# exit
```

4. In Remote Manager, click **Documentation** > **API Explorer**.
5. Select the devices to use as the SCI targest:
   a. Click **SCI Targets**.
   b. Click **Add Targets**.
   c. Enter or select the device ID of one of the devices.
   d. Click **Add**.
   e. Enter or select the device ID of the second device and click **Add**.
   f. Click **OK**.
6. Click **Examples** > **SCI** > **Data Service** > **Send Request**.

   Code similar to the following will be displayed in the HTTP message body text box:

```
<sci_request version="1.0">
  <data_service>
  <targets>
    <device id="00000000-00000000-0000FFFF-A83CF6A3"/>
    <device id="00000000-00000000-0000FFFF-485740BC"/>
  </targets>
    <requests>
      <device_request target_name="myTarget">
        my payload string
      </device_request>
    </requests>
  </data_service>
</sci_request>
```

7. For the **device_request** element, replace the value of **target_name** with **showSystem**. This matches the **target** parameter of the **device_request.register** function in the showsystem.py application.

```
      <device_request target_name="showSystem">
```

8. Click **Send**.

You should receive a response similar to the following:

```
<sci_reply version="1.0">
  <data_service>
    <device id="00000000-00000000-0000FFFF-A83CF6A3"/>
      <requests>
        <device_request target_name="showSystem" status="0">Model
           : Digi IX15
          Serial Number            : IX15-000068
          Hostname                 : IX15
          MAC                      : 00:40:D0:13:35:36

          Hardware Version         : 50001959-01 A
          Firmware Version         : 22.2.9.85
          Bootloader Version       : 1
          Firmware Build Date      : Thurs, 03 March 2022 10:16:23
          Schema Version           : 461

          Timezone                 : UTC
          Current Time             : Thurs, 03 March 2022 10:16:23
          CPU                      : 1.1
          Uptime                   : 1 day, 21 hours, 49 minutes, 47
seconds (164987s)
          Temperature              : 39C

          Contact                  : Jane Smith

          Disk
          ----
          Load Average             : 0.10, 0.05, 0.00
          RAM Usage                : 85.176MB/250.484MB(34%)
          Disk /etc/config Usage   : 0.068MB/13.416MB(1%)
          Disk /opt Usage          : 47.724MB/5309.752MB(1%)
          Disk /overlay Usage      : MB/MB(%)
          Disk /tmp Usage          : 0.004MB/40.96MB(0%)
          Disk /var Usage          : 0.820MB/32.768MB(3%)</device_
request>
      </requests>
    </device>
    <device id="00000000-00000000-0000FFFF-485740BC"/>
      <requests>
        <device_request target_name="showSystem" status="0">Model
           : Digi IX15
          Serial Number            : IX15-000023
          Hostname                 : IX15
          MAC                      : 00:40:D0:26:79:1C

          Hardware Version         : 50001959-01 A
          Firmware Version         : 22.2.9.85
          Bootloader Version       : 1
          Firmware Build Date      : Thurs, 03 March 2022 10:16:23
          Schema Version           : 461

          Timezone                 : UTC
          Current Time             : Thurs, 03 March 2022 10:16:23
          CPU                      : 1.1
          Uptime                   : 4 day, 13 hours, 43 minutes, 22
seconds (395002s)
```

```
          Temperature              : 37C

          Contact                  : Omar Ahmad
          Disk
          ----
          Load Average             : 0.10, 0.05, 0.00
          RAM Usage                : 85.176MB/250.484MB(34%)
          Disk /etc/config Usage   : 0.068MB/13.416MB(1%)
          Disk /opt Usage          : 47.724MB/5309.752MB(1%)
          Disk /overlay Usage      : MB/MB(%)
          Disk /tmp Usage          : 0.004MB/40.96MB(0%)
          Disk /var Usage          : 0.820MB/32.768MB(3%)</device_
request>
      </requests>
    </device>
  </data_service>
</sci_request>
```

Help for using Python to respond to Digi Remote Manager SCI requests

Get help for respond to Digi Remote Manager Server Command Interface (SCI) requests by accessing help for **digidevice.device_request**:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

   ```
   # python
   Python 3.6.13 (default, May  9 2021, 22:49:59)
   [GCC 8.3.0] on linux
   Type "help", "copyright", "credits" or "license" for more information.
   >>>
   ```

3. Import the **device_request** submodule:

   ```
   >>> from digidevice import device_request
   >>>
   ```

4. Use the help command with **device_request**:

   ```
   >>> help(device_request)
   Help on module digidevice.device_request in digidevice:

   NAME
   digidevice.device_request - APIs for registering device request handlers
   ...
   ```

   You can also use the help command with available **device_request** functions:

   - Use the help command with **device_request.register**:

     ```
     >>> help(device_request.register)
     Help on function register in module digidevice.device_request:
     ```

```
register(target:str, response_callback:Callable[[str, str], str],
status_callback:Callable[[int, str], NoneType]=None, xml_
encoding:str='UTF-8')
...
```

- Use the help command with **device_request.unregister**:

```
>>> help(device_request.unregister)
Help on function unregister in module digidevice.device_request:

unregister(target:str) -> bool
...
```

5. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Use digidevice runtime to access the runtime database

Use the **runt** submodule to access and modify the device runtime database.

Read from the runtime database

Use the **keys()** and **get()** methods to read the device configuration:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **runt** submodule:

```
>>> from digidevice import runt
>>>
```

4. Use the **start()** method to open the runtime database:

```
>>> runt.start()
>>>
```

5. Use the **keys()** method to display available keys in the runtime database, and use the **get()** method to print information from the runtime database:
   a. Print available keys:

```
>>> print(runt.keys(""))
```

   This returns available keys:

```
['advanced', 'drm', 'firmware', 'location', 'manufacture', 'metrics',
'mm', 'network', 'pam', 'serial', 'system']
```

b. Print available keys for the system key:

```
>>> print(runt.keys("system"))
```

This will return the following:

```
['boot_count', 'chassis', 'cpu_temp', 'cpu_usage', 'disk', 'load_avg',
'local_time', 'mac', 'mcu', 'model', 'ram', 'serial', 'uptime']
```

c. Use the **get()** method to print the device's MAC address:

```
>>> print(runt.get("system.mac"))
```

This will return the MAC address of the device.

6. Use the **stop()** method to close the runtime database:
7. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Modify the runtime database

Use the **set()** method to modify the runtime database:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **runt** submodule:

```
>>> from digidevice import runt
>>>
```

4. Use **start()** method to open the runtime database:

```
>>> runt.start()
>>>
```

5. Use the **set()** method to make changes to the runtime database:

```
>>> runt.set("my-variable", "my-value")
>>>
```

6. Use the **get()** method to verify the change:

```
>>> print(runt.get("my-variable"))
my-variable
>>>
```

7. Close the runtime database:

```
>>> runt.stop()
>>>
```

8. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Help for using Python to access the runtime database

Get help for reading and modifying the device runtime database by accessing help for **digidevice.runt**:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **runt** submodule:

```
>>> from digidevice import runt
>>>
```

4. Use the help command with **runt**:

```
>>> help(runt)

Help on module acl.runt in digidevice:

NAME
acl.runt - Python interface to ACL runtime database (runtd).
...
```

5. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Use Python to upload the device name to Digi Remote Manager

The **name** submodule can be used to upload a custom name for your device to Digi Remote Manager.

When you use the **name** submodule to upload a custom device name to Remote Manager, the following issues apply:

- If the name is being used by to another device in your Remote Manager account, the name will be removed from the previous device and added to the new device.

- If Remote Manager is configured to apply a profile to a device based on the device name, changing the name of the device may cause Remote Manager to automatically push a profile onto the device.

Together, these two features allow you to swap one device for another by using the **name** submodule to change the device name, while guaranteeing that the new device will have the same configuration as the previous one.

Note Because causing a profile to be automatically pushed from Remote Manager may change the behavior of the device, including overwriting existing usernames and passwords, the **name** submodule should be used with caution. As a result, support for this functionality is disabled by default on Remote Manager.

Enable support on Digi Remote Manager for uploading custom device names

1. In Remote Manager, click **API Explorer**.
2. For the HTTP method, select **PUT**.
3. For **Enter and API or select an example**, type **/ws/v1/settings/inventory/AllowDeviceToSetOwnNameEnabled**.
4. In the HTTP message body text box, type the following:

```
{
   "name" : "AllowDeviceToSetOwnNameEnabled",
   "value" : "true"
}
```

5. Click **Send**.

Upload a custom name

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **name** submodule:

```
>>> from digidevice import name
```

4. Upload the name to Remote Manager:

```
>>> name.upload("my_name")
```

5. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Help for uploading the device name to Digi Remote Manager

Get help for uploading the device name to Digi Remote Managerby accessing help for **digidevice.name**:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **name** submodule:

```
>>> from digidevice import name
>>>
```

4. Use the help command with **name**:

```
>>> help(name)

Help on module digidevice.name in digidevice:

NAME
digidevice.name - API for uploading name from the device
...
```

5. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Use Python to set the maintenance window

The **maintenance** Python module allows you to set the service state of a device. When the module sets the device to out of service, this can be used as trigger to begin maintenance activity. See Schedule system maintenance tasks for more details.

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **maintenance** module:

```
>>> from digidevice import maintenance
>>>
```

4. To determine the current service state of the device:

```
>>> maintenance.state()
'IN_SERVICE'
>>>
```

5. To set the device to out of service:

```
>>> maintenance.out_of_service()
>>> maintenance.state()
'OUT_OF_SERVICE'
>>>
```

6. To set the device to in service:

```
>>> maintenance.in_service()
>>> maintenance.state()
'IN_SERVICE'
>>>
```

Note Leave the interactive Python session active while completing task two, below. Once you have completed task two, exit the interactive session by using **Ctrl-D**. You can also exit the session using **exit()** or **quit()**.

Help for the digidevice maintenance module

Get help for the digidevice maintenance module:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Import the **maintenance** submodule:

```
>>> from digidevice import maintenance
>>>
```

4. Use the help command with **maintenance** :

```
>>> help(maintenance )
Help on module digidevice.maintenance in digidevice:

NAME
    digidevice.maintenance

DESCRIPTION
    API for setting the device's service state. The service state is
stored
    in runt.
...
```

5. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

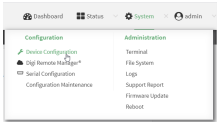
Use Python to send and receive SMS messages

You can create Python scripts that send and receive SMS message in tandem with the Digi Remote Manager or Digi aView by using the digidevice.sms module. To use a script to send or receive SMS messages, you must also enable the ability to schedule SMS scripting.

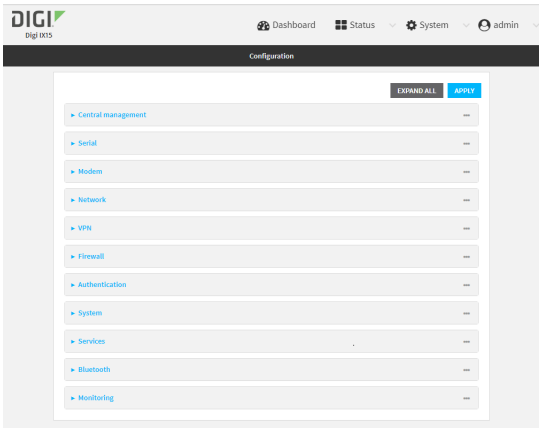
Enable the ability to schedule SMS scripting

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

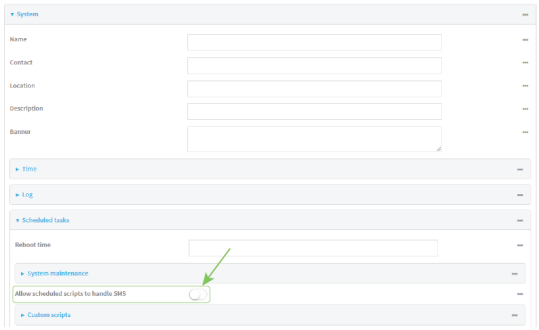


The **Configuration** window is displayed.



3. Click **System** > **Scheduled tasks**.

4. Click to enable **Allow scheduled scripts to handle SMS**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type:

```
(config)> system schedule sms_script_handling true
(config)>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

See Configure scripts to run automatically for more information about scheduling scripts.

Example digidevice.sms code

The following example code receives an SMS message and sends a response:

```
#!/usr/bin/python3.6
```

```
import os
import threading
import sys
from digidevice.sms import Callback, send
COND = threading.Condition()

def sms_test_callback(sms, info):
    print(f"SMS message from {info['content.number']} received")
    print(sms)
    print(info)
    COND.acquire()
    COND.notify()
    COND.release()

def send_sms(destination, msg):
    print("sending SMS message", msg)
    if len(destination) == 10:
        destination = "+1" + destination
    send(destination, msg)

if __name__ == '__main__':
    if len(sys.argv) > 1:
        dest = sys.argv[1]
    else:
        dest = '+15005550006'
    my_callback = Callback(sms_test_callback, metadata=True)
    send_sms(dest, 'Hello World!')
    print("Please send an SMS message now.")
    print("Execution halted until a message is received or 60 seconds have
passed.")
    # acquire the semaphore and wait until a callback occurs
    COND.acquire()
    try:
        COND.wait(60.0)
    except Exception as err:
        print("exception occured while waiting")
        print(err)
    COND.release()
    my_callback.unregister_callback()
```

Use Python to access serial ports

You can use the Python **serial** module to access serial ports on your IX15 device that are configured to be in Application mode. See Configure Application mode for information about configuring a serial port in Application mode.

To use Python to access serial ports:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

2. Determine the path to the serial port:

   ```
   # ls /dev/serial/
   by-id   by-path   by-usb   port1
   #
   ```

3. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

4. Import the **serial** module:

```
>>> import serial
>>>
```

5. You can now perform operations on the serial port. For example, to write a message to the serial port:

```
>>> s = serial.Serial("/dev/serial/port1", 115200)
>>> s.write(b"Hello from serial port")
26
>>>
```

6. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Use the Paho MQTT python library

Your IX15 device includes support for the Paho MQTT python library. MQTT is a lightweight messaging protocol used to communicate with various applications including cloud-based applications such as Amazon Web Services and Microsoft Azure. The following is example code that reads CPU and RAM usage on the device, updates the device firmware, then publishes information about DHCP clients and system information to the MQTT server at 192.168.1.100. The MQTT server IP is configurable.

```
"""
MQTT client example:
- Reporting some device metrics from runt
- Reporting DHCP clients
- Firmware update feature (simple implementation, read TODO in cmd_fwupdate)
"""

import sys
import time
import paho.mqtt.client as mqtt
import json
from acl import runt, config
from http import HTTPStatus
import urllib.request
import tempfile
import os
from digidevice import cli

POLL_TIME = 60

def cmd_reboot(params):
    print("Rebooting unit...")
    try:
        cli.execute("reboot", 10)
```

```
        except:
            print("Failed to run 'reboot' command")
            return HTTPStatus.INTERNAL_SERVER_ERROR

    return HTTPStatus.OK

    def cmd_fwupdate(params):
        try:
            fw_uri = params["uri"]
        except:
            print("Firmware file URI not passed")
            return HTTPStatus.BAD_REQUEST

        print("Request to update firmware with URI: {}".format(fw_uri))

           try:
            fd, fname = tempfile.mkstemp()
            os.close(fd)
            try:
                urllib.request.urlretrieve(fw_uri, fname)
            except:
                print("Failed to download FW file from URI {}".format(fw_uri))
                return HTTPStatus.NOT_FOUND

            try:
                ret = cli.execute("system firmware update file " + fname, 60)
            except:
                print("Failed to run firmware update command")
                return HTTPStatus.INTERNAL_SERVER_ERROR

            if not "Firmware update completed" in ret:
                print("Failed to update firmware")
                return HTTPStatus.INTERNAL_SERVER_ERROR
        finally:
            os.remove(fname)

        print("Firmware update finished")

        return HTTPStatus.OK

    CMD_HANDLERS = {
        "reboot": cmd_reboot,
        "fw-update": cmd_fwupdate
    }


    def send_cmd_reply(client, cmd_path, cid, cmd, status):
        if not status or not cid:
            return

        if cmd_path.startswith(PREFIX_CMD):
            path = cmd_path[len(PREFIX_CMD):]
        else:
            print("Invalid command path ({}), cannot send reply".format(cmd_path))
            return

        reply = {
            "cmd": cmd,
            "status": status
        }
```

```
    client.publish(PREFIX_RSP + path + "/" + cid, json.dumps(reply, separators=
(',',':')))

def on_connect(client, userdata, flags, rc):
    print("Connected to MQTT server")
    client.subscribe(PREFIX_CMD + "/system")

def on_message(client, userdata, msg):
    """ Supporting only a single topic for now, no need for filters
    Expects the following message format:
    {
        "cid": "<client-id>",
        "cmd": "<command>",
        "params": {
           <optional_parameters>
        }
    }

    Supported commands:
    - "fw-update"
        params:
            - "uri": "<firmware_file_URL>"
    - "reboot"
        params:
    """

    try:
        m = json.loads(msg.payload)
        cid = m["cid"]
        cmd = m["cmd"]
        try:
            payload = m["params"]
        except:
            payload = None
    except:
        print("Invalid command format: {}".format(msg.payload))
        if not cid:
            # Return if client-ID not passed
            return None
        send_cmd_reply(client, msg.topic, cid, cmd, HTTPStatus.BAD_REQUEST)

    try:
        status = CMD_HANDLERS[cmd](payload)
    except:
        print("Invalid command: {}".format(cmd))
        status = HTTPStatus.NOT_IMPLEMENTED

    send_cmd_reply(client, msg.topic, cid, cmd, status)


def publish_dhcp_leases():
    leases = []
    try:
        with open('/etc/config/dhcp.leases', 'r') as f:
            for line in f:
                elems = line.split()
                if len(elems) != 5:
                    continue
                leases.append({"mac": elems[1], "ip": elems[2], "host": elems
```

```
[3]})
        if leases:
            client.publish(PREFIX_EVENT + "/leases", json.dumps(leases,
separators=(',',':')))
    except:
        print("Failed to open DHCP leases file")

def publish_system():
    avg1, avg5, avg15 = runt.get("system.load_avg").split(', ')
    ram_used = runt.get("system.ram.per")
    disk_opt = runt.get("system.disk./opt.per")
    disk_config = runt.get("system.disk./etc/config.per")

    msg = json.dumps({
        "load_avg": {
            "1min": avg1,
            "5min": avg5,
            "15min": avg15
        },
        "disk_usage": {
            "/opt": disk_opt,
            "/etc/config:": disk_config,
            "ram": ram_used
        }
    })

    client.publish(PREFIX_EVENT + "/system", json.dumps(msg))

runt.start()
serial = runt.get("system.serial")

PREFIX = "router/" + serial
PREFIX_EVENT = "event/" + PREFIX
PREFIX_CMD = "cmd/" + PREFIX
PREFIX_RSP = "rsp/" + PREFIX

client = mqtt.Client()
client.on_connect = on_connect
client.on_message = on_message

try:
    client.connect("192.168.1.100", 1883, 60)
    client.loop_start()
except:
    print("Failed to connect to MQTT server")
    sys.exit(1)

while True:
    publish_dhcp_leases()
    publish_system()
    time.sleep(POLL_TIME)
```

Set up the IX15 to automatically run your applications

This section contains the following topics:

- Configure scripts to run automatically
- Show script information
- Stop a script that is currently running

Configure scripts to run automatically

You can configure a script or a python application to run automatically when the system restarts, at specific intervals, or at a specified time. By default, scripts execute in a "sandbox," which restricts access to the file system and available commands that can be used by the script.

Required configuration items

- Upload or create the script.
- Enable the application to be run script .
- Select whether the script should run:
  - When the device boots.
  - At a specified time.
  - At a specified interval.
  - During system maintenance.

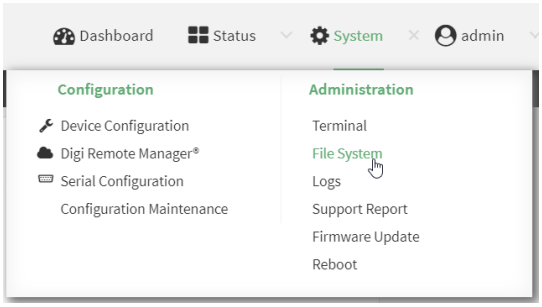
Additional configuration items

- A label used to identify the script .
- The action to take if the script finishes. The actions that can be taken are:
  - None.
  - Restart the script.
  - Reboot the device.
- Whether to write the script output and errors to the system log.
- If the script is set to run at a specified interval, whether another instance of the script should be run at the specified interval if the previous instance is still running.
- The memory available to be used by the script .
- Whether the script should run one time only.

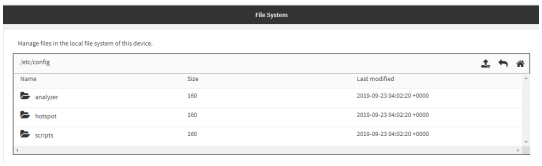
Task one: Upload the application

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Administration**, click **File System**.



The **File System** page appears.



3. Highlight the **scripts** directory and click ↱ to open the directory.
4. Click ⬆ (upload).
5. Browse to the location of the script on your local machine. Select the file and click **Open** to upload the file.

The uploaded file is uploaded to the **/etc/config/scripts** directory.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, use the scp command to upload the Python application script to the IX15 device:

```
> scp host hostname-or-ip user username remote remote-path local local-path to local
```

   where:

   - *hostname-or-ip* is the hostname or ip address of the remote host.
   - *username* is the name of the user on the remote host.
   - *remote-path* is the path and filename of the file on the remote host that will be copied to the IX15 device.
   - *local-path* is the location on the IX15 device where the copied file will be placed.

   For example:

   To upload a script from a remote host with an IP address of 192.168.4.1 to the /etc/config/scripts directory on the IX15 device, issue the following command:

```
> scp host 192.168.4.1 user admin remote /home/admin/bin/test.py local
/etc/config/scripts/ to local
admin@192.168.4.1's password: adminpwd
test.py                          100%    36MB    11.1MB/s        00:03
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

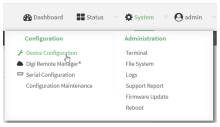
Note You can also create scripts by using the **vi** command when logged in with shell access.

Task two: Configure the application to run automatically

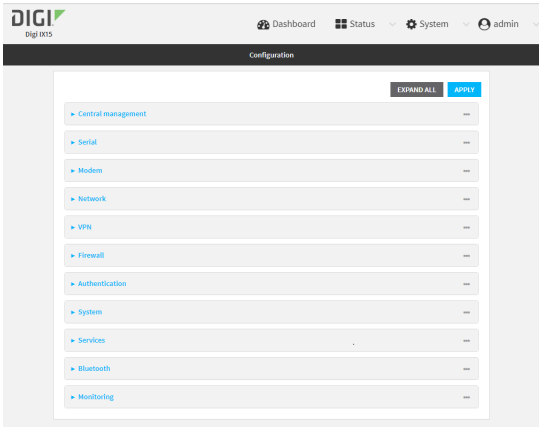
Note This feature does not provide syntax or error checking. Certain commands can render the device inoperable. Use with care.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

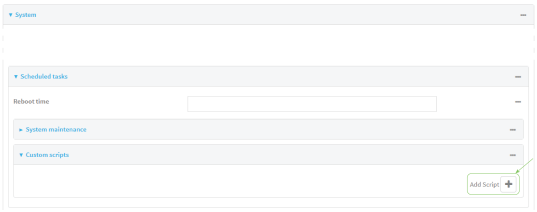


   The **Configuration** window is displayed.

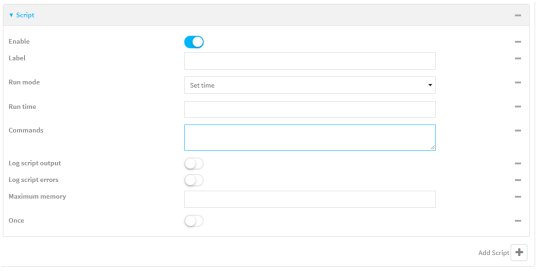


3. Click **System** > **Scheduled tasks** > **Custom scripts**.

4. For **Add Script**, click ✚.



The script configuration window is displayed.



Custom scripts are enabled by default. To disable, click **Enable** to toggle off.

5. (Optional) For **Label**, provide a label for the script.
6. For **Run mode**, select the mode that will be used to run the script. Available options are:
   - **On boot**: The script will run once each time the device boots.
     - If **On boot** is selected, select the action that will be taken when the script completes in **Exit action**. Available options are:
       - **None**: Action taken when the script exits.
       - **Restart script**: Runs the script repeatedly.
       - **Reboot**: The device will reboot when the script completes.
   - **Interval**: The script will start running at the specified interval, within 30 seconds after the configuration change is saved.
     - If **Interval** is selected, in **Interval**, type the interval.

       Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

       For example, to set **Interval** to ten minutes, enter **10m** or **600s**.
     - Click to enable **Run single** to run only a single instance of the script at a time.

       If **Run single** is not enabled, a new instance of the script will be started at every interval, regardless of whether the script is still running from a previous interval.
   - **Set time**: Runs the script at a specified time of the day.
     - If **Set Time** is selected, specify the time that the script should run in **Run time**, using the format *HH*:*MM*.
   - **During system maintenance**: The script will run during the system maintenance time window.
7. For **Commands**, enter the commands that will execute the script.

   If the script begins with **#!**, then the script will be invoked in the location specified by the path for the script command. Otherwise, the default shell will be used (equivalent to **#!/bin/sh**).

8. Script logging options:
   a. Click to enable **Log script output** to log the script's output to the system log.
   b. Click to enable **Log script errors** to log script errors to the system log.

   If neither option is selected, only the script's exit code is written to the system log.
9. For **Maximum memory**, enter the maximum amount of memory available to be used by the script and its subprocesses, using the format *number*{**b**|**bytes**|**KB**|**k**|**MB**|**MB**|**M**|**GB**|**G**|**TB**|**T**}.
10. **Sandbox** is enabled by default, which restricts access to the file system and available commands that can be used by the script. This option protects the script from accidentally destroying the system it is running on.
11. Click to enable **Once** to configure the script to run only once at the specified time.

    If **Once** is enabled, rebooting the device will cause the script to not run again. The only way to re-run the script is to:
    - Remove the script from the device and add it again.
    - Make a change to the script.
    - Uncheck **Once**.
12. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a script:

```
(config)> add system schedule script end
(config system schedule script 0)>
```

   Scheduled scripts are enabled by default. To disable:

```
(config system schedule script 0)> enable false
(config system schedule script 0)>
```

4. (Optional) Provide a label for the script.

```
(config system schedule script 0)> label value
(config system schedule script 0)>
```

   where *value* is any string. if spaces are used, enclose *value* within double quotes.

5. Set the mode that will be used to run the script:

```
(config system schedule script 0)> when mode
(config system schedule script 0)>
```

where *mode* is one of the following:

- **boot**: The script will run once each time the device boots.
  - If **boot** is selected, set the action that will be taken when the script completes:

```
(config system schedule script 0)> exit_action action
(config system schedule script 0)>
```

    where *action* is one of the following:

    - **none**: Action taken when the script exits.
    - **restart**: Runs the script repeatedly.
    - **reboot**: The device will reboot when the script completes.
- **interval**: The script will start running at the specified interval, within 30 seconds after the configuration change is saved. If **interval** is selected:
  - Set the interval:

```
(config system schedule script 0)> on_interval value
(config system schedule script 0)>
```

    where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

    For example, to set **on_interval** to ten minutes, enter either **10m** or **600s**:

```
(config system schedule script 0)> on_interval  600s
(config system schedule script 0)>
```

  - (Optional) Configure the script to run only a single instance at a time:

```
(config system schedule script 0)> once true
(config system schedule script 0)>
```

    If **once** is set to **false**, a new instance of the script will be started at every interval, regardless of whether the script is still running from a previous interval.
- **set_time**: Runs the script at a specified time of the day.
  - If **set_time** is set, set the time that the script should run, using the format *HH*:*MM*:

```
(config system schedule script 0)> run_time HH:MM
(config system schedule script 0)>
```

- **maintenance_time**: The script will run during the system maintenance time window.

6. Set the commands that will execute the script:

```
(config system schedule script 0)> commands filename
(config system schedule script 0)>
```

where *filename* is the path and filename of the script, and any related command line information.

If the script begins with **#!**, then the script will be invoked in the location specified by the path for the script command. Otherwise, the default shell will be used (equivalent to **#!/bin/sh**).

7. Script logging options:
   - To log the script's output to the system log:

     ```
     (config system schedule script 0)> syslog_stdout true
     (config system schedule script 0)>
     ```

   - To log script errors to the system log:

     ```
     (config system schedule script 0)> syslog_stderr true
     (config system schedule script 0)>
     ```

   If **syslog_stdout** and **syslog_stderr** are not enabled, only the script's exit code is written to the system log.
8. Set the maximum amount of memory available to be used by the script and its subprocesses:

   ```
   (config system schedule script 0)> max_memory value
   (config system schedule script 0)>
   ```

   where *value* uses the syntax ***number***{**b**|**bytes**|**KB**|**k**|**MB**|**M**|**GB**|**G**|**TB**|**T**}.
9. To run the script only once at the specified time:

   ```
   (config system schedule script 0)> once true
   (config system schedule script 0)>
   ```

   If **once** is enabled, rebooting the device will cause the script to run again. The only way to re-run the script is to:
   - Remove the script from the device and add it again.
   - Make a change to the script.
   - Disable **once**.
10. **Sandbox** is enabled by default. This option protects the script from accidentally destroying the system it is running on.

    ```
    (config system schedule script 0)> sandbox true
    (config system schedule script 0)>
    ```

11. Save the configuration and apply the change:

    ```
    (config)> save
    Configuration saved.
    >
    ```

12. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

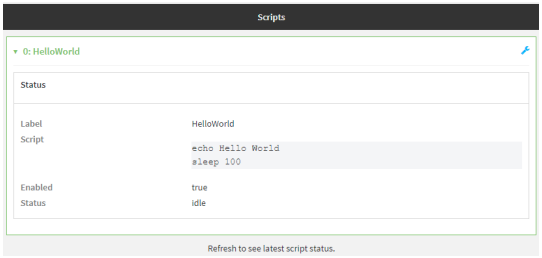
Show script information

You can view status and statistics about location information from either the WebUI or the command line.

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. At the **Status** page, click **Scripts**.

   The **Scripts** page displays:



Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Use the show scripts command at the system prompt:

```
> show scripts

 Index  Label        Enabled  Status  Run time
 -----  -----------  -------  ------  --------
 0      script1      true     active
 1      script2      true     idle     01:00
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

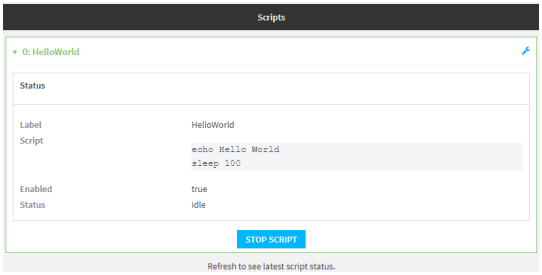
Stop a script that is currently running

You can stop a script that is currently running.

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. At the **Status** page, click **Scripts**.

   The **Scripts** page displays:



3. For scripts that are currently running, click **Stop Script** to stop the script.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Determine the name of scripts that are currently running:

```
> show scripts

 Index  Label        Enabled  Status  Run time
 -----  -----------  -------  ------  --------
 0      script1      true     active
 1      script2      true     idle     01:00
>
```

   Scripts that are currently running have the status of **active**.
3. Stop the appropriate script:

```
)> system script stop script1
>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Start an interactive Python session

Use the **python** command without specifying any parameters to start an interactive Python session. The Python session operates interactively using REPL (Read Evaluate Print Loop) to allow you to write Python code on the command line.

Note The Python interactive session is not available from the Admin CLI. You must access the device shell in order to run Python applications from the command line. See Authentication groups for information about configuring authentication groups that include shell access.

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the shell prompt, use the **python** command with no parameters to enter an interactive Python session:

```
# python
Python 3.6.13 (default, May  9 2021, 22:49:59)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

3. Type Python commands at the Python prompt. For example, to view help for the digidevice module, type:

```
>>> help("digidevice")
Help on package digidevice:

NAME
    digidevice - Digi device python extensions

DESCRIPTION
    This module includes various extensions that allow Python
    to interact with additional features offered by the device.
...
```

4. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Run a Python application at the shell prompt

Python applications can be run from a file at the shell prompt. The Python application will run until it completes, displaying output and prompting for additional user input if needed. To interrupt the application, enter **CTRL-C**.

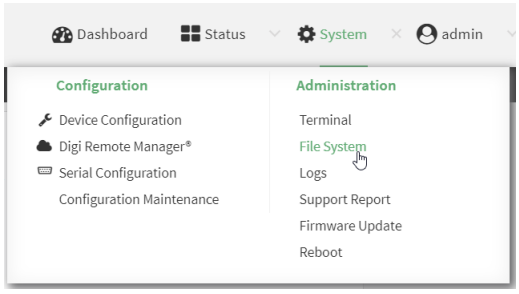
Note Python applications cannot be run from the Admin CLI. You must access the device shell in order to run Python applications from the command line. See Authentication groups for information about configuring authentication groups that include shell access.

1. Upload the Python application to the IX15 device:

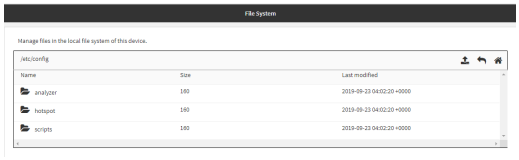
   WebUI

a. Log into the IX15 WebUI as a user with Admin access.

b. On the menu, click **System**. Under **Administration**, click **File System**.



The **File System** page appears.



c. Highlight the **scripts** directory and click to open the directory.

d. Click (upload).

e. Browse to the location of the script on your local machine. Select the file and click **Open** to upload the file.

The uploaded file is uploaded to the **/etc/config/scripts** directory.

Command line

a. Log into the IX15 command line as a user with Admin access.

Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

b. At the command line, use the scp command to upload the Python application script to the IX15 device:

```
> scp host hostname-or-ip user username remote remote-path local
local-path to local
```

where:

- *hostname-or-ip* is the hostname or ip address of the remote host.
- *username* is the name of the user on the remote host.
- *remote-path* is the path and filename of the file on the remote host that will be copied to the IX15 device.
- *local-path* is the location on the IX15 device where the copied file will be placed.

For example:

To upload a script from a remote host with an IP address of 192.168.4.1 to the /etc/config/scripts directory on the IX15 device, issue the following command:

```
> scp host 192.168.4.1 user admin remote /home/admin/bin/test.py local
/etc/config/scripts/ to local
admin@192.168.4.1's password: adminpwd
test.py                         100%    36MB    11.1MB/s        00:03
>
```

c. Type **exit** to exit the Admin CLI.

Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Note You can also create scripts by using the **vi** command when logged in with shell access.

2. Log into the IX15 command line as a user with shell access.

Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

3. Use the **python** command to run the Python application. In the following example, the Python application, **test.py**, takes 3 parameters: **120**, **ports** and **storage**:

```
# python /etc/config/scripts/test.py 120 ports storage
```

Install third party Python modules

The IX15 offers a set of Python modules to manage your device interfaces, its configuration or to connect to the cloud. But your application may require of other Python libraries not already pre-installed. Use **pip** to install from PyPI Python third-party modules.

Install the latest version of "SomeProject"

```
pip install "SomeProject"
```

Install a specific version of "SomeProject"

```
pip install "SomeProject==1.4"
```

Note Only modules completely implemented in Python can be installed.

Python migration guide

This section describes the Python programing differences between previous generations of Digi XBee gateways and the new IX15 Gateway and explains how to properly migrate Python code.

Note This section does not cover the **DIA framework** and how to port DIA applications. For these cases, contact Digi for additional information about DIA migration.

Programming IDE

The first main difference while working with the IX15 Gateway is the Python application development IDE.

In previous generations, **Digi ESP for Python** was the IDE used to create and work with Python projects for Digi XBee gateways. The new generation of XBee gateways starting with the IX15 use

Pycharm as the selected IDE to develop Python applications. Digi has created a set of plugins that allows you to easily create new Python projects, import existing samples, transfer the application to the gateway, and execute it remotely with only a few clicks. You can find more information about this new IDE as well as a getting started guide in the *Digi XBee PyCharm IDE Plugin User Guide*.

Note Using the PyCharm IDE to develop Python applications for the gateway is totally optional. You can develop Python applications using any software and then transfer them to the IX15 Gateway.

Python version

Another big difference of the IX15 compared to previous XBee gateways is the Python version used to program applications. Previous Digi XBee gateways use Python 2.6, while the new generation of gateways use Python 3.6.

The new Python environment running in the IX15 Gateway is based on a more standard and supported set of APIs and libraries, making the Python code more generic and portable than ever. This is achieved with the use of pip to install all the required Python libraries and dependencies for your application in the gateway. This is an automatic process when using Pycharm.

The following topics describe how to use Python to access the different gateway interfaces and capabilities:

- Hardware access
- CLI
- Interaction with Digi Remote Manager is key for the XBee gateways. For this reason, a set of APIs are still available to perform the most common Digi Remote Manager operations, although they are implemented in a different way:
- Power management
- SMS
- XBee

Hardware access

Some Python applications may require to interact with the gateway hardware interfaces to perform specific actions. Some of the most common accessed hardware interfaces in the gateways are:

- LEDs
- Watchdog
- Temperature sensor
- GPS

LEDs

In previous generations, the **digihw** Python module is used to control the gateway user LEDs:

1. Invoke the **user_led_set(value, number)** function specifying the new LED value and the LED number.

Previous API

```
import digihw

# Turn on user LED's
digihw.user_led_set(1, 1)
```

```
# Add a 1-second delay delay
time.sleep(1.0)

# Turn off the LED's
digihw.user_led_set(0, 1)
```

In the new XBee gateways all the LEDs are controlled by the firmware. That means that there is not a user-reserved LED to use by the applications. However, you can request ownership of any LED and make use of it as long as the Python application is running.

The **digidevice.led** Python module helps you in this process:

1. Request LED control by invoking the **acquire(led)** function specifying the LED name to control.
2. After that, use the **set(led, state)** function to change the LED behavior.
3. When done, invoke the **release(led)** function to return the LED control to the system.

New API

```
from digidevice import led
from digidevice.led import Led, State
import time

led.acquire(Led.XBEE1)
led.set(Led.XBEE1, State.ON)
time.sleep(1)
led.set(Led.XBEE1, State.FLASH)
time.sleep(5)
led.set(Led.XBEE1, State.OFF)
time.sleep(1)
led.release(Led.XBEE1)
```

Watchdog

In previous generations, the **digiwdog** Python module was used to control the gateway watchdog:

1. Create a watchdog object with **Watchdog(timeout, name)** constructor specifying the watchdog timeout and name.
2. Invoke the **stroke()** method to update the watchdog periodically.

Previous API

```
import digiwdog
import time

wd = digiwdog.Watchdog(10, "ForceReset")
while True:
    wd.stroke()
    time.sleep(1)
```

In the new XBee gateways, there is no access to the system watchdog.

Temperature sensor

In previous generations, the **digihw** Python module was used to read the device temperature.:

1. Invoke the **temperature()** function to retrieve the current temperature.

Previous API

```
import digihw

sample = digihw.temperature()
print "Celsius: ", sample
```

In the new XBee gateways, there is not a specific API to access the device temperature:

1. It is possible to read it from the file system.

New API

```
TEMPERATURE_FILE = "/sys/class/thermal/thermal_zone0/temp"

with open(TEMPERATURE_FILE, 'r') as file:
    sample = file.read()

print("Millicelsius: %s" % sample)
```

GPS

In previous generations, the **digihw** Python module was used to read the GPS location:

1. Invoke the **gps_location()** function to retrieve the current GPS location.

Previous API

```
import digihw

gps_data = digihw.gps_location()
latitude, longitude, altitude, timestamp = gps_data
print "Latitude  : %s" % latitude
print "Longitude : %s" % longitude
print "Altitude  : %d" % altitude
```

The new XBee gateways do not have a GPS interface, so there is no specific API to access the device location.

CLI

In previous generations, the **digicli** Python module was used to access the system CLI:

1. Invoke the **digicli(command)** function with the CLI command to execute.

Previous API

```
import digicli

status, output = digicli.digicli('show net')
if status:
    for line in output:
        if line.find('MAC Address') >= 0:
            l = line.split(':')
            print "".join(l[1:]).strip()
```

In the new XBee gateways, the **digidevice.cli** Python module is used to access the system CLI:

1. Invoke the **execute(command)** function with the CLI command to execute.

New API

```
from digidevice import cli

response = cli.execute("show system")
print(response)
```

Digi Remote Manager

Interaction with Digi Remote Manager is key for the XBee gateways. For this reason, a set of APIs are still available to perform the most common Digi Remote Manager operations, although they are implemented in a different way:

- Upload datapoints
- Subscribe device request
- Remove device request

Upload datapoints

In previous generations, the **idigidata** Python module was used to upload data points to Digi Remote Manager:

1. Create the full contents of the request in CSV format.
2. Pass it to the **send_to_idigi(csv_string, url)** function.

Previous API

```
import idigidata

doc = """\
#TIMESTAMP,DATA,DATATYPE,STREAMID
1441108800000,75,INTEGER,temperature
"""
success, code, msg = idigidata.send_to_idigi(doc, "DataPoint/upload.csv")

if not success:
    print "Got error code %d (%s) uploading data" % (code, msg)
```

In the new XBee gateways, the **digidevice.datapoint** Python module is used to upload data points to Digi Remote Manager:

1. Call the **upload(stream, data)** function with the datapoint information. Invoke the method for each datapoint you want to upload.

```
from digidevice import datapoint
import time

datapoint.upload("Velocity", 69, units="mph")
datapoint.upload("Temperature", 24, geo_location=(54.409469, -1.718836,
```

```
129))
datapoint.upload("Emergency_Door", "closed", timestamp=time.time())
```

Subscribe device request

In previous generations, the **idigidata** Python module was used to register callbacks from Digi Remote Manager:

1. Use the **register_callback(target, handler)** function passing the target name and callback handler as arguments.

Previous API

```
import idigidata
import time

def handler(target, data):
    # Remove leading and trailing whitespace
    data = data.strip()
    # Print out what we got
    print("received request %s for target %s" % (data, target))
    return "OK"

# Register the callback function for the target name "myTarget".
handle = idigidata.register_callback("myTarget", handler)
```

In the new XBee gateways, the **digidevice.device_request** Python module is used to register callbacks from Digi Remote Manager:

1. Invoke the **register(target, handler)** function passing the target name and callback handler as arguments.

New API

```
from digidevice import device_request

def handler(target, data):
    # Remove leading and trailing whitespace
    data = data.strip()
    # Print out what we got
    print("received request %s for target %s" % (data, target))
    return "OK"# Register the callback function for the target name
"myTarget".
device_request.register("myTarget", handler)
```

Remove device request

In previous generations, the **idigidata** Python module was used to unregister callbacks from Digi Remote Manager:

1. Use the **unregister_callback(handler)** function passing the registration handler as argument.

Previous API

```
import idigidata
```

```
def handler ...

handle = idigidata.register_callback("myTarget", handler)
idigidata.unregister_callback(handle)
```

In the new XBee gateways, the **digidevice.device_request** Python module is used to unregister callbacks from Digi Remote Manager:

1. Invoke the **unregister(target)** function passing the target name argument.

New API

```
from digidevice import device_request

def handler ...

device_request.register("myTarget", handler)
device_request.unregister("myTarget")
```

Power management

When the IX15 is powered by batteries, the power consumption is a very important factor to consider. For this reason, Digi created a set of power management APIs to control the device power functions.

In previous generations, the **digipowercontrol** Python module was used to control the power management features of the gateway. You could suspend—put the device in sleep mode—and wake up after a configured period of time or using an external GPIO—wake up button—as trigger:

1. Use the **system_power_set(0, duration)** function to suspend the device for the specified amount of seconds.
2. Use the **wake_on_external_irq_set(gpio_num, mode)** function to configure the provided GPIO as wake up source.

Previous API

```
import digipowercontrol

DURATION = 5.0

# Go to sleep, wake-up and reset after the number of seconds entered
# This function will not return
digipowercontrol.system_power_set(0, DURATION)
```

Previous API

```
import digipowercontrol

# The GPIO described in the Hardware Reference Manual for your board
# In this case GPIO4 is connected to the wake up button on the board.
WAKE_UP_GPIO = 0x04

# Wake up when the button is pressed, for the this board
# the GPIO4 is pulled low when the button is pressed, so we
# wakeup on the falling edge.
digipowercontrol.wake_on_external_irq_set(WAKE_UP_GPIO,
```

```
digipowercontrol.WAKE_ON_IRQ_FALLING_EDGE)

# Go to sleep, reset when a trigger occurs, this routine does not return
digipowercontrol.system_power_set(0)
```

The IX15 does not include a specific API for power management control. You can access the different power management options using the **digidevice.config** Python module:

- Configure power profile
- Configure wake up sources
- Put the IX15 in sleep mode

Configure power profile

A Power profile is a set of settings that determine how the system will behave in terms of power consumption during standard operating mode. You can choose to preserve power, performance or to balance both:

- **Performance**: The CPU clock frequency is scaled up to work in the highest available frequency and provide a better system performance.
- **Auto**: The CPU clock frequency is dynamically scaled up and down to provide better performance during high demanding conditions and also to save power during inactivity periods.
- **Power save**: The CPU clock frequency is scaled down to work in the lowest available frequency and save power.
- **Manual**: Allows you to manually set the working frequency of the CPU. When this option is selected, the setting **Custom frequency** is available to set the CPU working frequency manually:
  - 198 MHz
  - 396 MHz
  - 528 MHz
  - 792 MHz

To change the active power profile, use **digidevice.config** Python module as follows:

```
from digidevice import config

# Load configuration.
cfg = config.load(writable=True)

# Set the active power profile:
cfg.set("system.power.profile", "auto")
#cfg.set("system.power.profile", "performance")
#cfg.set("system.power.profile", "powersave")
#cfg.set("system.power.profile", "manual")

# When the power profile is set to manual, you can specify the working
frequency:
#cfg.set("system.power.custom_freq", 198000)
#cfg.set("system.power.custom_freq", 396000)
#cfg.set("system.power.custom_freq", 528000)
#cfg.set("system.power.custom_freq", 792000)
```

```
# Apply and save configuration
cfg.commit()
```

Configure wake up sources

The wake up sources refer to the different mechanisms and triggers used to wake up the device and put it in normal operating mode again when the device is in suspend mode. There are three available wake up sources:

- **XBee**: wake up the device when any data is received in the XBee interface.
- **Serial port**: wake up the device when any data is received in the Serial Port.
- **RTC alarm**: configure an alarm and wake up the device when the alarm triggers.

To configure the different wake up sources, use the **digidevice.config** Python module as follows:

New API

```
from digidevice import config

# Load configuration.
cfg = config.load(writable=True)

# Configure the wake up sources:
cfg.set("system.power.wakeup_sources.xbee", True)
#cfg.set("system.power.wakeup_sources.xbee", False)
cfg.set("system.power.wakeup_sources.serial", True)
#cfg.set("system.power.wakeup_sources.serial", False)
cfg.set("system.power.wakeup_sources.rtc", True)
#cfg.set("system.power.wakeup_sources.rtc", False)

# When the RTC alarm wake up source is configured, you have to configure
the date/time of the alarm using this format: [YYYY-MM-DD hh:mm:ss]
cfg.set("system.power.wakeup_sources.rtc_time", "YYYY-MM-DD hh:mm:ss")

# Apply and save configuration
cfg.commit()
```

Put the IX15 in sleep mode

The sleep or suspend mode is a special state where the CPU, most of the RAM and most of the digital peripherals are powered off to save as much power as possible. The IX15 Gateway can be commanded to go to suspend mode at any time.

To put the device in sleep mode use the following Python code:

New API

```
import os

# This method blocks until the device wakes up
os.system("suspend")
```

SMS

One common feature of cellular capable gateways is the use of SMS messages for communication. The IX15 Gateways have a cellular interface, so these are the SMS related actions that can be

executed:

- Receive an SMS
- Send a SMS

Receive an SMS

In previous generations, the **digisms** Python module was used to receive SMS messages:

1. Specify a handler to receive the SMS text message.
2. Register a callback by invoking the **Callback(handler)** function.

Previous API

```
import digisms

def sms_callback(sms):
    print """\

Message from: %s
at: %s
====================
%s
====================
""" % (sms.source_addr, sms.timestamp, sms.message)

# Register the SMS callback
my_callback = digisms.Callback(sms_callback)
# Leave the program executing
input()
my_callback.unregister_callback()
```

In the new XBee gateways, the **digidevice.sms** Python module is used to receive SMS messages.:

1. Specify a handler to receive the SMS text message.
2. Register a callback by invoking the **Callback(handler)** function.

New API

```
from digidevice import sms

def sms_callback(sms):
    print("\nMessage from: %s" % sms['from'])
    print("====================")
    print(sms['message'])
    print("====================")

# Register the SMS callback
my_callback = sms.Callback(sms_callback)
# Leave the program executing
input()
my_callback.unregister_callback()
```

Send a SMS

In previous generations, the **digisms** Python module was used to send SMS messages:

1. Invoke the **send(address, message)** function with the SMS destination and the message to send.

Previous API

```
import digisms
import time

ADDRESS = "15551234567"
MESSAGE = "Hello, World!"

cur_time = time.strftime("%a, %d %b %Y %H:%M:%S", time.gmtime())
msg = "%s: Message %s" % (cur_time, MESSAGE)
digisms.send(address, msg)
```

In the new XBee gateways, the **digidevice.sms** Python module is used to send SMS messages:

1. Invoke the **send(address, message)** function with the SMS destination and the message to send.

New API

```
from digidevice import sms
import time

ADDRESS = "15551234567"
MESSAGE = "Hello, World!"

cur_time = time.strftime("%a, %d %b %Y %H:%M:%S", time.gmtime())
msg = "%s: Message %s" % (cur_time, MESSAGE)
sms.send(address, msg)
```

XBee

XBee device access from Python code is the part that has undergone more changes when comparing the IX15 Gateway with previous gateways.

In previous generations, the **zigbee** Python module was used as the entry point to every XBee related operation in the gateway.

In the new XBee gateways, all the XBee operations are based in the open source Digi Python Library that is already installed in the gateway. This library is protocol-agnostic and provides access to every XBee related operation using a object oriented philosophy. Find more information about this library at:

- Source code: https://github.com/digidotcom/xbee-python
- Documentation: https://xbplib.readthedocs.io/en/latest/

This is how the most common XBee operations are executed:

- List/discover nodes
- Node information
- Read XBee settings
- Write XBee settings
- Execute XBee commands

- Receive data
- Send data
- Send broadcast data
- Register device in trustcenter
- Unregister device from trustcenter

List/discover nodes

In previous generations, the **zigbee** Python module was used to list the available XBee nodes of the network:

1. Invoke the **getnodelist()** method to perform a discovery of XBee nodes. It returns the list of discovered nodes.

Previous API

```
import zigbee

# Perform a node discovery:
node_list = zigbee.getnodelist()
```

In the new XBee gateways, the **digidevice.xbee** Python module is used to list and discover the available XBee nodes of a network. The node list and node discovery are two separated operations.

List nodes

To list the nodes of the network:

1. List the XBee of the gateway with the **get_device()** function.
2. Once you have the local instance, get the XBee network using the **get_network()** method.
3. List the available nodes with **get_devices()**.

New API

```
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    xbee_network = device.get_network()
    node_list = xbee_network.get_devices()
finally:
    if device.is_open():
        device.close()
```

Discover nodes

To perform an XBee network discover:

1. Use the XBee network instance to register a network discovery callback.
2. Invoke the **start_discovery()** method.

New API

```
import time

from digi.xbee.models.status import NetworkDiscoveryStatus
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    xbee_network = device.get_network()
    xbee_network.set_discovery_timeout(15)  # 15 seconds.

    # Callback for discovered devices.
    def callback_device_discovered(remote):
        print("Device discovered: %s" % remote)

    # Callback for discovery finished.
    def callback_discovery_finished(status, desc=None):
        if status == NetworkDiscoveryStatus.SUCCESS:
            print("Discovery process finished successfully")
        else:
            info = status.description
            if desc:
                info = "%s (%s)" % (desc, status.description)
            print("There was an error discovering devices: %s" % info)

    xbee_network.add_device_discovered_callback(callback_device_
discovered)
    xbee_network.add_discovery_process_finished_callback(callback_
discovery_finished)
    xbee_network.start_discovery_process()
    while xbee_network.is_discovery_running():
        time.sleep(0.1)
    node_list =  xbee_network.get_devices()
finally:
    if device.is_open():
        device.close()
```

Node information

In previous generations, the **zigbee** Python module was used to list the available XBee nodes of the network. With the node list, you could iterate the different nodes and print their information:
Previous API

```
import zigbee

# Perform a node discovery:
node_list = zigbee.getnodelist()

# For each node, print its information
for node in node_list:
    print("Node Identifier: %s" % node.label)
    print("64-bit address:  %s" % node.addr_extended)
    print("16-bit address:  %s" % node.addr_short)
    print("Role:            %s" % node.type)
```

In the new XBee gateways, the **digidevice.xbee** Python module is used to list and discover the available XBee nodes of the network. When you have the XBee node list, iterate the nodes and print their information:

New API

```
from digi.xbee.util import utils
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    xbee_network = device.get_network()
    node_list =  xbee_network.get_devices()
    # For each node, print its information
    for node in node_list:
        print("Node Identifier:  %s" % node.get_node_id())
        print("64-bit address:   %s" % node.get_64bit_addr())
        print("16-bit address:   %s" % node.get_16bit_addr())
        print("Role:             %s" % node.get_role().description)
        print("Firmware version: %s" % utils.hex_to_string(node.get_
firmware_version()))
        print("Hardware version: %s" % node.get_hardware_version
().description)
finally:
    if device.is_open():
        device.close()
```

Read XBee settings

In previous generations, the **zigbee** Python module was used to read any parameter from XBee nodes in the network:

1. Invoke **ddo_get_param(ext_addr, parameter)** function with the 64-bit address of the XBee to read from and the setting ID to read.

Previous API

```
import zigbee

DESTINATION="[00:13:a2:00:40:0a:07:8d]!"

value = zigbee.ddo_get_param(DESTINATION, 'NI')

print "Read NI value: %s" % (value)
```

In the new XBee gateways, the **digidevice.xbee** Python module is used to read any parameter from XBee nodes in the network:

1. When you have the XBee instance, invoke the **get_parameter(parameter)** method specifying the setting ID to read.
2. You can execute this action in local and remote XBee devices. Note that value is returned as a **byte array**.

Local device

New API

```
from digidevice import xbee
```

```
device = xbee.get_device()
try:
    device.open()
    value = device.get_parameter("NI")
    print("Read NI value: %s" % value.decode())
finally:
    if device.is_open():
        device.close()
```

Remote devices

New API

```
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    xbee_network = device.get_network()
    node_list = xbee_network.get_devices()
    # For each node, read the NI setting
    for node in node_list:
        value = node.get_parameter("NI")
        print("Read NI value for node %s: %s" % (node, value.decode()))
finally:
    if device.is_open():
        device.close()
```

Write XBee settings

In previous generations, the **zigbee** Python module was used to write any parameter to XBee nodes in the network:

1. Invoke **ddo_set_param(ext_addr, parameter, value)** function with the 64-bit address of the XBee to write to, the setting ID to write, and the value.

Previous API

```
import zigbee

DESTINATION="[00:13:a2:00:40:0a:07:8d]!"

zigbee.ddo_set_param(DESTINATION, 'D1', 1)
```

In the new XBee gateways, the **digidevice.xbee** Python module is used to write any parameter to any XBee node in the network:

1. When you have the XBee instance, invoke the **set_parameter(parameter, value)** method specifying the setting ID to write and the value.
2. You can execute this action in local and remote XBee devices. Note that value must be provided as a **byte array**.

Local device

```
>from digi.xbee.util import utils
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    device.set_parameter("D1", bytearray([5]))
finally:
    if device.is_open():
        device.close()
```

Remote devices

```
from digi.xbee.util import utils
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    xbee_network = device.get_network()
    node_list = xbee_network.get_devices()
    # For each node, set the D1 setting to 5
    for node in node_list:
        node.set_parameter("D1", bytearray([5]))
finally:
    if device.is_open():
        device.close()
```

Execute XBee commands

In previous generations, the **zigbee** Python module was used to execute commands in any XBee node of the network.:

1. Invoke the **ddo_command(ext_addr, command)** function specifying the 64-bit address of the destination XBee and the command ID to execute.

Previous API

```
import zigbee

DESTINATION="[00:13:a2:00:40:0a:07:8d]!"

zigbee.ddo_command(DESTINATION, 'RE')
```

In the new XBee gateways, the **digidevice.xbee** Python module is used to execute commands in any XBee node of the network:

1. When you have the XBee instance, invoke the **execute_command(command)** method with the command ID to execute.
2. You can execute this action in local and remote XBee devices.

Local device

New API

```
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    device.execute_command("RE")
finally:
    if device.is_open():
        device.close()
```

Remote devices

```
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    xbee_network = device.get_network()
    node_list = xbee_network.get_devices()
    # Execute the "RE" command in each node
    for node in node_list:
        node.execute_command("RE")
finally:
    if device.is_open():
        device.close()
```

Receive data

In previous generations, the **socket** Python module was used to read data from any XBee node of the network:

1. Create a socket with the required XBee options.
2. Bind it to the desired end point, cluster ID, and profile ID.
3. All the data is received as "explicit data", so you have to bind to the proper end point and cluster ID.
4. Reading from the socket is a blocking operation, but you can use **select** for an non-blocking read.

Blocking read
Previous API

```
import xbee
from socket import *

# Create the socket, datagram mode, proprietary transport:
sd = socket(AF_XBEE, SOCK_DGRAM, XBS_PROT_TRANSPORT)

# Bind to endpoint 0xe8 (232) for ZB/DigiMesh, but 0x00 for 802.15.4
s.bind(("", end_point, profile_id, cluster_id))

# Block until a single frame is received, up to 255 bytes:
payload, src_addr = sd.recvfrom(255)
```

Non-blocking read
Previous API

```
import xbee
from socket import *
from select import *

# Create the socket, datagram mode, proprietary transport:
sd = socket(AF_XBEE, SOCK_DGRAM, XBS_PROT_TRANSPORT)
# Bind to endpoint 0xe8 (232) for ZB/DigiMesh, but 0x00 for 802.15.4
sd.bind(("", 0xe8, 0, 0))
# Configure the socket for non-blocking operation:
sd.setblocking(0)

try:
    # Initialize state variables:
    payload = ""    src_addr = ()
    # Forever:
    while 1:
        # Reset the ready lists:
        rlist, wlist = ([], [])
        if len(payload) == 0:
            # If the payload buffer is empty,
            # add socket to read list:
            rlist = [sd]
        else:
            # Otherwise, add the socket to the
            # write list:
            wlist = [sd]

        # Block on select:
        rlist, wlist, xlist = select(rlist, wlist, [])
        # Is the socket readable?
        if sd in rlist:
            # Receive from the socket:
            payload, src_addr = sd.recvfrom(72)
            # If the packet was "quit", then quit:
            if payload == "quit":
                raise Exception, "quit received"        # Is the socket
writable?
        if sd in wlist:
            # Send to the socket:
            count = sd.sendto(payload, 0, src_addr)
            # Slice off count bytes from the buffer,
            # useful for if this was a partial write:
            payload = payload[count:]

except Exception, e:
    # upon an exception, close the socket:
    sd.close()
```

In the new XBee gateways, the way data is read from XBee devices in the network is totally different. Using the **digidevice.xbee** Python module, you can read data by polling—blocking the execution until a message is received—or asynchronously using the local XBee device instance. For asynchronous reads, you can specifically read explicit data too.

Asynchronous data read

Once you have the local XBee device instance:

1. Define a handler for the received data.
2. Register a data received callback using the **add_data_received_callback(data_handler)** method with the handler as argument.
3. Every time the local XBee receives data, the registered handler is executed with the received XBee message as argument. It contains information about the sender and the data received.

New API

```
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()

    def data_receive_callback(xb_msg):
        print("From %s >> %s" % (xb_msg.remote_device, xb_msg.data.decode
()))

    device.add_data_received_callback(data_receive_callback)
    print("Waiting for data...\n")
    # Keep the program executing until a key is pressed
    input()
finally:
    if device.is_open():
        device.close()
```

Asynchronous explicit data read

Once you have the local XBee instance:

1. Define a handler for the received data.
2. Register a data received callback using the **add_expl_data_received_callback(explicit_data_handler)** method with the handler as argument.
3. Every time the local XBee receives explicit data, the registered handler is executed with the received explicit XBee message as argument. It contains information about the sender and the explicit data received.

New API

```
from digi.xbee.models.mode import APIOutputModeBit
from digi.xbee.util import utils
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    # Configure device te receive data as explicit data.
    device.set_api_output_mode_value(APIOutputModeBit.calculate_api_
output_mode_value(device.get_protocol(), {APIOutputModeBit.EXPLICIT}))

    def explicit_data_callback(expl_xb_msg):
        print("From %s >> %s" % (expl_xb_msg.remote_device, expl_xb_
msg.data.decode()))
        print(" - Source endpoint:        %s" % utils.hex_to_string(expl_
xb_msg.source_endpoint.to_bytes(1, byteorder='big')))
```

```
        print(" - Destination endpoint:   %s" % utils.hex_to_string(expl_
xb_msg.dest_endpoint.to_bytes(1, byteorder='big')))
        print(" - Cluster ID:             %s" % utils.hex_to_string(expl_
xb_msg.cluster_id.to_bytes(1, byteorder='big')))
        print(" - Profile ID:             %s" % utils.hex_to_string(expl_
xb_msg.profile_id.to_bytes(1, byteorder='big')))

    device.add_expl_data_received_callback(explicit_data_callback)
    print("Waiting for explicit data...\n")
    # Keep the program executing until a key is pressed
    input()
finally:
    if device.is_open():
        device.close()
```

Polling data read

Once you have the local XBee device instance:

1. Invoke the **read_data(timeout)** or **read_data_from(remote_node, timeout)** methods to wait for incoming data.
2. If no timeout is specified, these methods return immediately if there is no available data.

New API

```
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    # Wait a maximum of 10 seconds for incoming data
    xbee_message = read_data(timeout=10)
    if xbee_message:
        print("From %s >> %s" % (xbee_message.remote_device,
                                 xbee_message.data.decode()))
    else:
        print("No data available")
finally:
    if device.is_open():
        device.close()
```

Send data

In previous generations, the **socket** Python module was used to send data to any XBee node of the network:

1. Create a socket with the required XBee options
2. Bind it to the desired end point, cluster ID, and profile ID.
3. Only explicit data can be sent as you have to specify end point, cluster ID, and profile ID when building the destination address.

Previous API

```
import xbee
from socket import *
```

```
DESTINATION=("00:0d:6f:00:00:06:89:29!", 0xe8, 0xc105, 0x11)

# Create the socket, datagram mode, proprietary transport:
sd = socket(AF_XBEE, SOCK_DGRAM, XBS_PROT_TRANSPORT)

# Bind to endpoint 0xe8 (232) for ZB/DigiMesh, but 0x00 for 802.15.4
s.bind(("", end_point, profile_id, cluster_id))

# Send "Hello, World!" to the destination node, endpoint,
# using the profile_id and cluster_id specified in DESTINATION:
sd.sendto("Hello, World!", 0, DESTINATION)
```

In the new XBee gateways, the way data is sent to XBee devices in the network is totally different. Using the **digidevice.xbee** Python module, you can send data to any remote XBee device synchronously or asynchronously using the local XBee instance. You can also send explicit data to a specific end point, cluster ID, and profile ID.

Send data synchronously

Once you have the local XBee instance:

1. Get the destination node from the XBee network.
2. Invoke the **send_data(remote_device, data)** method specifying the destination node and the payload to send.
3. This method blocks until the data is successfully sent, an error occurs, or the timeout elapses.

New API

```
from digidevice import xbee

DATA_TO_SEND = "Hello, World!"
REMOTE_NODE_ID = "REMOTE"

device = xbee.get_device()
try:
    device.open()
    # Obtain the remote XBee device from the XBee network.
    xbee_network = device.get_network()
    remote_device = xbee_network.get_device_by_node_id(REMOTE_NODE_ID)
    if remote_device:
        print("Sending data to %s % remote_device)
        device.send_data(remote_device, DATA_TO_SEND)
        print("Success")
    else:
        print("Remote device not found")
finally:
    if device.is_open():
        device.close()
```

Send data asynchronously

Once you have the local XBee device instance:

1. Get the destination node from the XBee network.
2. Invoke the **send_data_async(remote_device, data)** method specifying the destination node and the payload to send.
3. This method returns immediately.

New API

```
from digidevice import xbee

DATA_TO_SEND = "Hello, World!"
REMOTE_NODE_ID = "REMOTE"

device = xbee.get_device()
try:
    device.open()
    # Obtain the remote XBee device from the XBee network.
    xbee_network = device.get_network()
    remote_device = xbee_network.get_device_by_node_id(REMOTE_NODE_ID)
    if remote_device:
        print("Sending data to %s % remote_device)
        device.send_data_async(remote_device, DATA_TO_SEND)
        print("Success")
    else:
        print("Remote device not found")
finally:
    if device.is_open():
        device.close()
```

Send explicit data synchronously

Once you have the local XBee instance:

1. Get the destination node from the XBee network.
2. Invoke the **send_expl_data(remote_device, data, source_end_point, dest_end_point, cluster_id, profile_id)** method specifying the destination node, the payload to send, the end points, the cluster ID, and the profile ID.
3. This method blocks until the data is successfully sent, an error occurs, or send timeout elapses.

New API

```
from digidevice import xbee

DATA_TO_SEND = "Hello, World!"
REMOTE_NODE_ID = "REMOTE"
SRC_ENDPOINT = 0xA0
DEST_ENDPOINT = 0xA1
CLUSTER_ID = 0x1554
PROFILE_ID = 0x1234

device = xbee.get_device()
try:
    device.open()
    # Obtain the remote XBee device from the XBee network.
    xbee_network = device.get_network()
    remote_device = xbee_network.get_device_by_node_id(REMOTE_NODE_ID)
    if remote_device:
```

```
        print("Sending explicit data to %s % remote_device)
        device.send_expl_data(remote_device, DATA_TO_SEND, SRC_ENDPOINT,
DEST_ENDPOINT, CLUSTER_ID, PROFILE_ID)
        print("Success")
    else:
        print("Remote device not found")
finally:
    if device.is_open()
```

Send explicit data asynchronously

Once you have the local XBee instance:

1. Get the destination node from the XBee network.
2. **Iinvoke** the **send_expl_data_async(remote_device, data, source_end_point, dest_end_point, cluster_id, profile_id)** method specifying the destination node, the payload to send, the end points, the cluster ID, and the profile ID.
3. This method returns immediately.

New API

```
from digidevice import xbee

DATA_TO_SEND = "Hello, World!"
REMOTE_NODE_ID = "REMOTE"
SRC_ENDPOINT = 0xA0
DEST_ENDPOINT = 0xA1
CLUSTER_ID = 0x1554
PROFILE_ID = 0x1234

device = xbee.get_device()
try:
    device.open()
    # Obtain the remote XBee device from the XBee network.
    xbee_network = device.get_network()
    remote_device = xbee_network.get_device_by_node_id(REMOTE_NODE_ID)
    if remote_device:
        print("Sending explicit data to %s % remote_device)
        device.send_expl_data_async(remote_device, DATA_TO_SEND, SRC_
ENDPOINT, DEST_ENDPOINT, CLUSTER_ID, PROFILE_ID)
        print("Success")
    else:
        print("Remote device not found")
finally:
    if device.is_open():
        device.close()
```

Send broadcast data

In previous generations, the **socket** Python module was used to send broadcast data to all the XBee nodes of the network:

1. Create a socket with the required XBee options.
2. Bind it to the desired end point, cluster ID, and profile ID.

3. Only explicit data can be sent as you have to specify end point, cluster ID, and profile ID when building the destination address. Also, you must specify the **broadcast address** in the destination address.

Previous API

```
import xbee
from socket import *

# Broadcast address is "[00:00:00:00:00:00:FF:FF]!"
DESTINATION=("[00:00:00:00:00:00:FF:FF]!", 0xe8, 0xc105, 0x11)

# Create the socket, datagram mode, proprietary transport:
sd = socket(AF_XBEE, SOCK_DGRAM, XBS_PROT_TRANSPORT)

# Bind to endpoint 0xe8 (232) for ZB/DigiMesh, but 0x00 for 802.15.4
s.bind(("", end_point, profile_id, cluster_id))

# Send "Hello, World!" to the destination node, endpoint,
# using the profile_id and cluster_id specified in DESTINATION:
sd.sendto("Hello, World!", 0, DESTINATION)
```

In the new XBee gateways, the way data is broadcast to all XBee devices in the network is totally different. Using the **digidevice.xbee** Python module, you can send data to any remote XBee using the local XBee instance. You can also broadcast explicit data to a specific end point, cluster ID, and profile ID.

Standard data broadcast

Once you have the local XBee instance:

1. Invoke the **send_data_broadcast(data)** method specifying the data to broadcast.

New API

```
from digidevice import xbee

DATA_TO_SEND = "Hello, World!"

device = xbee.get_device()
try:
    device.open()
    device.send_data_broadcast(DATA_TO_SEND)
finally:
    if device.is_open():
        device.close()
```

Explicit data broadcast

Once you have the local XBee instance:

1. Invoke the **send_expl_data_broadcast(data, source_end_point, dest_end_point, cluster_id, profile_id)** method specifying the data to broadcast, the end points, the cluster ID, and the profile ID.

New API

```
from digidevice import xbee

DATA_TO_SEND = "Hello, World!"
SRC_ENDPOINT = 0xA0
DEST_ENDPOINT = 0xA1
CLUSTER_ID = 0x1554
PROFILE_ID = 0x1234

device = xbee.get_device()
try:
    device.open()
    device.send_expl_data_broadcast(DATA_TO_SEND, SRC_ENDPOINT, DEST_
ENDPOINT, CLUSTER_ID, PROFILE_ID)
finally:
    if device.is_open():
        device.close()
```

Register device in trustcenter

In previous generations, the **zigbee** Python module was used to register joining devices in an encrypted Zigbee network:

1. Invoke the **register_joining_device(addr_extended, key)** function specifying the 64-bit address of the XBee to register and the registration key.

Previous API

```
import zigbee

zigbee.register_joining_device("[00:0d:6f:00:00:06:89:29!]", "1234")
```

In the new XBee gateways, this action is available only when the XBee gateway is configured with Zigbee protocol. The **digidevice.xbee** Python module allows you to register joining devices in an encrypted Zigbee network using the local XBee instance:

1. Invoke the **register_joining_device(registrant_address, options, key)** method specifying the 64-bit address of the joining device, the registration options, and the key.

New API

```
from digi.xbee.models.address import XBee64BitAddress
from digi.xbee.models.options import RegisterKeyOptions
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    device.register_joining_device(XBee64BitAddress("000d6f0000068929"),
RegisterKeyOptions.LINK_KEY, "1234".decode())
finally:
    if device.is_open():
        device.close()
```

Unregister device from trustcenter

In previous generations, the **zigbee** Python module was used to unregister joining devices in an encrypted Zigbee network:

1. Invoke the **unregister_joining_device(addr_extended)** method specifying the 64-bit address of the XBee to unregister.

Previous API

```
import zigbee

zigbee.unregister_joining_device("[00:0d:6f:00:00:06:89:29!]")
```

In the new XBee gateways, this action is available only when the XBee gateway is configured with the Zigbee protocol. The **digidevice.xbee** Python module allows you to unregister joining devices in an encrypted Zigbee network using the local XBee instance:

1. Invoke the **unregister_joining_device(registrant_address, options, key)** method specifying the 64-bit address of the device to unregister.

New API

```
from digi.xbee.models.address import XBee64BitAddress
from digidevice import xbee

device = xbee.get_device()
try:
    device.open()
    device.unregister_joining_device(XBee64BitAddress("000d6f0000068929"))
finally:
    if device.is_open():
        device.close()
```

Deployment

Deploying a Python application in the gateway involves the process of transferring the application, all the required resources, and libraries, and configure the application to automatically start under predetermined circumstances.

- Transfer the application
- Configure application start

Transfer the application

When deploying an application in the gateway, all the source code, resources, and application required libraries must be transferred to the device.

In previous XBee gateways, this process is handled by the Digi ESP for Python IDE. The IDE automatically builds and transfers the application to the gateway. If your application requires additional libraries, you are responsible of either copying them to the project or manually uploading them to the gateway.

The IX15 Gateway uses Pycharm as the application development IDE with a set of Digi created plugins to ease the development process. These plugins automatically build and transfer the application code and resources to the gateway as in the past. The IDE also automatically installs any required Python library so that you do not have to manually install/copy them.

See the Digi XBee PyCharm IDE plugin guide to learn more on transferring application and dependencies to the gateway: Build and run the project.

Configure application start

Once your application, required resources, and dependencies are transferred to the gateway, the last step is to decide when the application should start. In most cases it should be once the device boots, but there are more options.

In previous generations, the Python application automatic start was configured from the gateway web interface:
Old deployment instructions

1. Login to the web interface.
2. Select the **Python** link on the left, then the **Auto-Start Settings**. You will be shown the four scripts you can auto-start.
3. Enter the program name, including the **.py** extension. Do not include the command: **Python**!
4. Click **Apply**.
5. Reboot the device.

The IX15 Gateway also uses the web interface to configure the application automatic start. The main difference is that there are new configuration options including the full start command. You can read how Configure scripts to run automatically.

User authentication

This chapter contains the following topics:

IX15 user authentication 620
User authentication methods 620
Authentication groups 628
Local users 637
Terminal Access Controller Access-Control System Plus (TACACS+) 651
Remote Authentication Dial-In User Service (RADIUS) 658
LDAP 664
Configure serial authentication 671
Disable shell access 673
Set the idle timeout for IX15 users 675
Example user configuration 678

IX15 user authentication

User authentication on the IX15 has the following features and default configuration:

Feature	Description	Default configuration
Idle timeout	Determines how long a user session can be idle before the system automatically disconnects.	■ 10 minutes.
Allow shell	If disabled, prevents all authentication prohibits access to the shell prompt for all authentication groups. This does not prevent access to the Admin CLI. **Note** If shell access is disabled, re-enabling it will erase the device's configuration and perform a factory reset.	■ Enabled.
Methods	Determines how users are authenticated for access: **local users**, **TACACS+**, or **RADIUS**.	■ **local users**.
Groups	Associates access permissions for a group. . You can modify the released groups and create additional groups as needed for your site. A user can be assigned to more than one group.	■ **admin**: Provides the logged-in user with administrative and shell access. ■ **serial**: Provides the logged-in user with access to serial ports.
Users	Defines local users for the IX15.	■ **admin**: Belongs to both the **admin** and **serial** groups.
TACACS+	Configures support for TACACS+ (Terminal Access Controller Access-Control System Plus) servers and users.	■ Not configured.
RADIUS	Configures support for RADIUS (Remote Authentication Dial-In User Service) servers and users.	■ Not configured.
LDAP	Configures support for LDAP (Lightweight Directory Access Protocol) servers and users.	■ Not configured.
Serial	Configures authentication for serial TCP and autoconnect services.	■ Not configured.

User authentication methods

Authentication methods determine how users of the IX15 device are authenticated. Available authentication methods are:

- **Local users**: User are authenticated on the local device.
- **RADIUS**: Users authenticated by using a remote RADIUS server for authentication.

  See Remote Authentication Dial-In User Service (RADIUS) for information about configuring RADIUS authentication.
- **TACACS+**: Users authenticated by using a remote TACACS+ server for authentication.

  See Terminal Access Controller Access-Control System Plus (TACACS+) for information about configuring TACACS+ authentication.
- **LDAP**: Users authenticated by using a remote LDAP server for authentication.

  See LDAP for information about configuring LDAP authentication.

Add a new authentication method

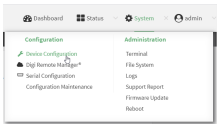
Required configuration items

- The types of authentication method to be used:

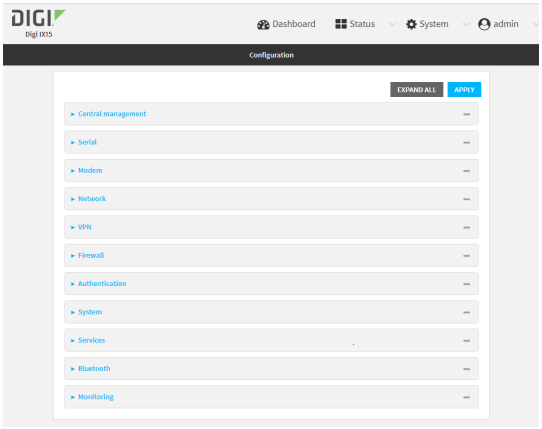
To add an authentication method:

WebUI

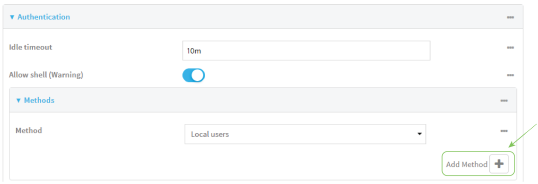
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



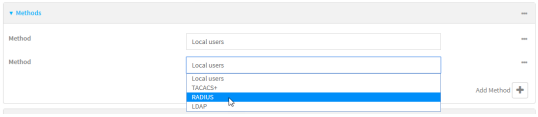
The **Configuration** window is displayed.



3. Click **Authentication** > **Methods**.
4. For **Add Method**, click ✚.



5. Select the appropriate authentication type for the new method from the **Method** drop-down.



Note Authentication methods are attempted in the order they are listed until the first successful authentication result is returned. See Rearrange the position of authentication methods for information about how to reorder the authentication methods.

6. Repeat these steps to add additional methods.
7. Click **Apply** to save the configuration and apply the change.



Command line

Authentication methods are attempted in the order they are listed until the first successful authentication result is returned. This procedure describes how to add methods to various places in the list.

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add the new authentication method to the appropriate location in the list:
   - To determine the current list of authentication methods:
     a. Log into the IX15 command line as a user with full Admin access rights.

        Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
     b. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

     c. Use the **show auth method** command to display the current authentication methods configuration:

```
(config)> show auth method
0 local
(config)>
```

   - To add the new authentication method to the beginning of the list, use the index value of **0** to indicate that it should be added as the first method:

```
(config)> add auth method 0 auth_type
(config)>
```

     where *auth_type* is one of **local**, **radius**, **tacacs+**, or **ldap**.

- To add the new authentication method to the end of the list, use the index keyword **end**:

```
(config)> add auth method end auth_type
(config)>
```

  where *auth_type* is one of **local**, **radius**, **tacacs+**, or **ldap**.

- To add the new authentication in another location in the list, use an index value to indicate the appropriate position. For example:

```
(config)> add auth method 1 auth_type
(config)>
```

  where *auth_type* is one of **local**, **radius**, **tacacs+**, or **ldap**.

- You can also use the **move** command to rearrange existing methods. See Rearrange the position of authentication methods for information about how to reorder the authentication methods.

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

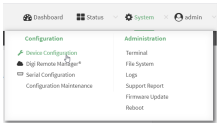
5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

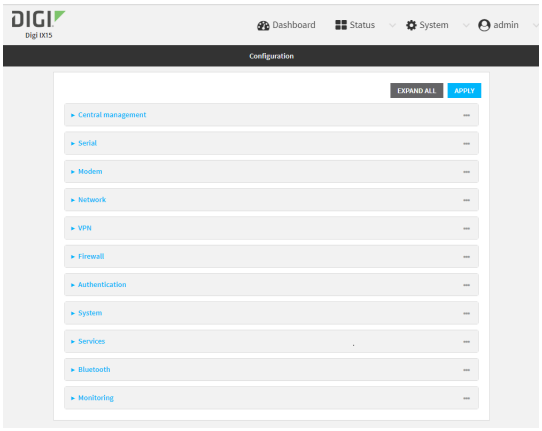
Delete an authentication method

WebUI

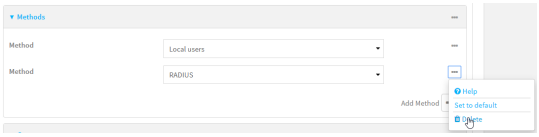
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **Authentication** > **Methods**.
4. Click the menu icon (**...**) next to the method and select **Delete**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Use the **show auth method** command to determine the index number of the authentication method to be deleted:

```
(config)> show auth method
0 local
1 radius
2 tacacs+
(config)>
```

4. Delete the appropriate authentication method:

```
(config)> del auth method n
```

Where *n* is index number of the authentication method to be deleted. For example, to delete the TACACS+ authentication method as displayed by the example **show** command, above:

```
(config)> del auth method 2
```

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

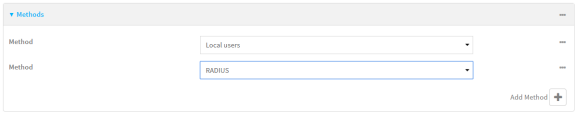
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Rearrange the position of authentication methods

WebUI

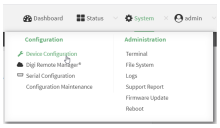
Authentication methods are reordered by changing the method type in the **Method** drop-down for each authentication method to match the appropriate order.

For example, the following configuration has **Local users** as the first method, and **RADIUS** as the second.

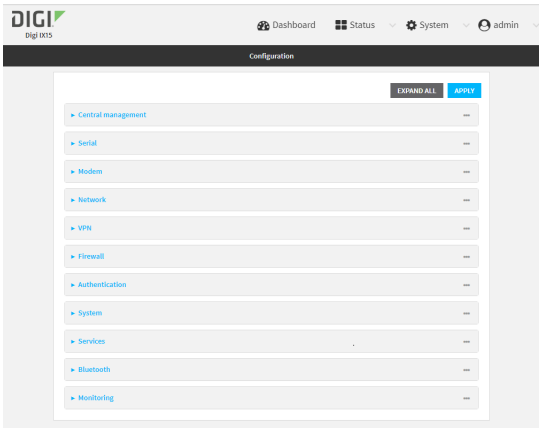


To reorder these so that **RADIUS** is first and **Local users** is second:

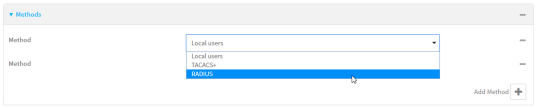
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

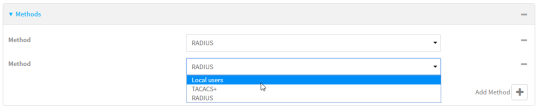
   The **Configuration** window is displayed.



3. Click to expand the first **Method**.
4. In the **Method** drop-down, select **RADIUS**.



5. Click to expand the second **Method**.
6. In the **Method** drop-down, select **Local users**.



7. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Use the **show** command to display current configuration:

```
(config)> show auth method
0 local
```

```
1 radius
(config)>
```

4. Use the **move** command to rearrange the methods:

```
(config)> move auth method 1 0
(config)>
```

5. Use the **show** command again to verify the change:

```
(config)> show auth method
0 radius
1 local
(config)>
```

6. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Authentication groups

Authentication groups are used to assign access rights to IX15 users. Three types of access rights can be assigned:

- **Admin access**: Users with Admin access can be configured to have either:
  - The ability to manage the IX15 device by using the WebUI or the Admin CLI.
  - Read-only access to the WebUI and Admin CLI.
- **Shell access**: Users with Shell access have the ability to access the shell when logging into the IX15 via ssh, telnet, or the serial console.

  Shell access is not available if the **Allow shell** parameter has been disabled. See Disable shell access for more information about the **Allow shell** parameter.
- **Serial access**: Users with Serial access have the ability to log into the IX15 device by using the serial console.

Preconfigured authentication groups

The IX15 device has two preconfigured authentication groups:

- The **admin** group is configured by default to have full **Admin access**.
- The **serial** group is configured by default to have **Serial access**.

The preconfigured authentication groups cannot be deleted, but the access rights defined for the group are configurable.

This section contains the following topics:

Change the access rights for a predefined group 629

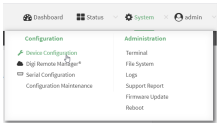
Add an authentication group 631
Delete an authentication group 635

Change the access rights for a predefined group

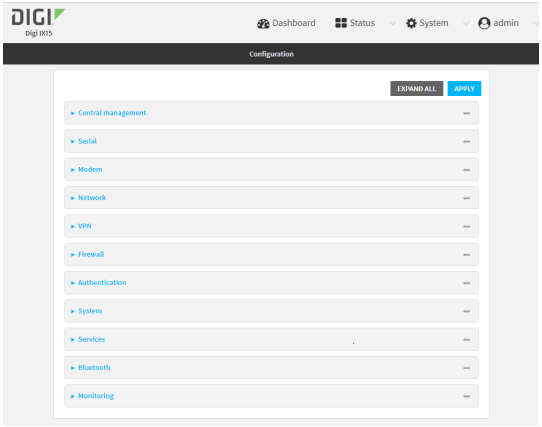
By default, two authentication groups are predefined: **admin** and **serial**. To change the access rights of the predefined groups:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Authentication** > **Groups**.
4. Click the authentication group to be changed, either **admin** or **serial**, to expand its configuration node.
5. Click the box next to the following options, as appropriate, to enable or disable access rights for each:
   - **Admin access**

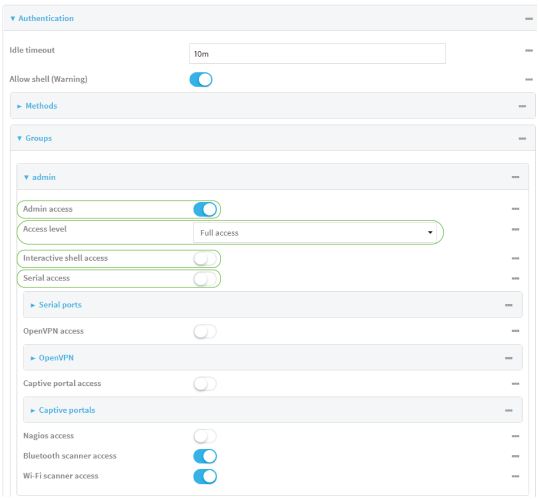
     For groups assigned Admin access, you can also determine whether the **Access level** should be **Full access** or **Read-only access**.

     - **Full access** provides users of this group with the ability to manage the IX15 device by using the WebUI or the Admin CLI.
     - **Read-only access** provides users of this group with read-only access to the WebUI and Admin CLI.

     The default is **Full access**.
   - **Interactive shell access**

Shell access is not available if the **Allow shell** parameter has been disabled. See Disable shell access for more information about the **Allow shell** parameter.

- **Serial access**



6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable or disable access rights for the group. For example:
   - Admin access:
     - To set the access level for Admin access of the **admin** group:

```
(config)> auth group admin acl admin level value
(config)>
```

       where *value* is either:
       - **full**: provides users of this group with the ability to manage the IX15 device by using the WebUI or the Admin CLI.
       - **read-only**: provides users of this group with read-only access to the WebUI and Admin CLI.

The default is **full**.

- To disable Admin access for the **admin** group:

```
(config)> auth group admin acl admin enable false
(config)>
```

- Shell access:
  - To enable Shell access for the **serial** group:

```
(config)> auth group serial acl shell enable true
(config)>
```

Shell access is not available if the **Allow shell** parameter has been disabled. See Disable shell access for more information about the **Allow shell** parameter.

- Serial access:
  - To enable Serial access for the **admin** group:

```
(config)> auth group admin acl serial enable true
(config)>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Add an authentication group

Required configuration items

- The access rights to be assigned to users that are assigned to this group.

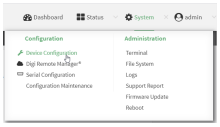
Additional configuration items

- Access rights to OpenVPN tunnels, and the tunnels to which they have access.
- Access rights to captive portals, and the portals to which they have access.
- Access rights to query the device for Nagios monitoring.

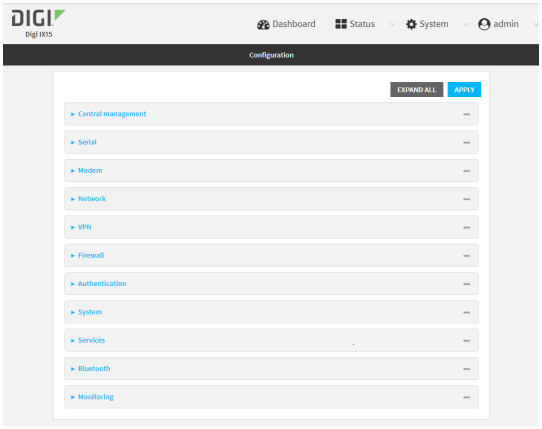
To add an authentication group:

WebUI

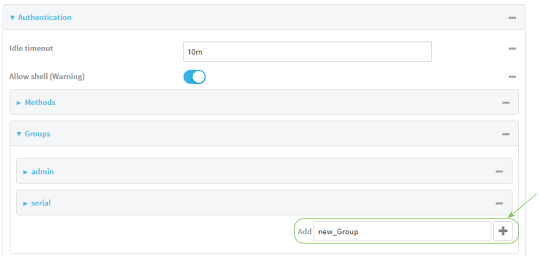
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Authentication** > **Groups**.
4. For **Add**, type a name for the group and click **+**.



The group configuration window is displayed.



5. Click the following options, as appropriate, to enable or disable access rights for each:
   - **Admin access**

     For groups assigned Admin access, you can also determine whether the **Access level** should be **Full access** or **Read-only access**.

     where *value* is either:

- **Full access full**: provides users of this group with the ability to manage the IX15 device by using the WebUI or the Admin CLI.
- **Read-only access read-only**: provides users of this group with read-only access to the WebUI and Admin CLI.

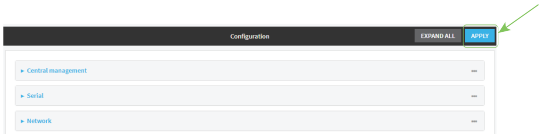
The default is **Full access full**.

- **Shell access**

  Shell access is not available if the **Allow shell** parameter has been disabled. See Disable shell access for more information about the **Allow shell** parameter.

- **Serial access**

6. (Optional) Configure the serial ports to which users of this group have access:
   a. Click **Serial ports** to expand the **Serial ports** node.
   b. For **Add Port**, click ✚.
   c. In the **Port** dropdown, select a port.
   d. Click ✚ again to add additional serial ports.
7. (Optional) Configure OpenVPN access. See for further information.
8. (Optional) Configure captive portal access:
   a. Enable captive portal access rights for users of this group by checking the box next to **Captive portal access**.
   b. Click **Captive portals** to expand the **Captive portal** node.
   c. For **Add Captive portal**, click ✚.
   d. In the **Captive portal** dropdown, select a captive portal to which users of this group will have access.
   e. Click ✚ again to add additional captive portals.
9. (Optional) Enable users that belong to this group to query the device for Nagios monitoring by checking the box next to **Nagios access**.
10. (Optional) Enable users that belong to this group to access the Bluetooth scanning service by checking the box next to **Bluetooth scanner access**.
11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Use the **add auth group** command to add a new authentication. For example, to add a group named **test**:

```
(config)> add auth group test
(config auth group test)>
```

4. Enable access rights for the group:
   - Admin access:

```
(config auth group test)> acl admin enable true
(config)>
```

   - Set the access level for Admin access:

```
(config)> auth group admin acl admin level value
(config)>
```

     where *value* is either:
     - **full**: provides users of this group with the ability to manage the IX15 device by using the WebUI or the Admin CLI.
     - **read-only**: provides users of this group with read-only access to the WebUI and Admin CLI.

     The default is **full**.
   - Shell access:

```
(config auth group test)> acl shell enable true
(config)>
```

     Shell access is not available if the **Allow shell** parameter has been disabled. See Disable shell access for more information about the **Allow shell** parameter.
   - Serial access:

```
(config auth group test)> acl serial enable true
(config)>
```

5. (Optional) Configure captive portal access:
   a. Return to the config prompt by typing three periods (**...**):

```
(config auth group test)> ...
(config)>
```

   b. Enable captive portal access rights for users of this group:

```
(config)> auth group test acl portal enable true
(config)>
```

   c. Add a captive portal to which users of this group will have access:
      i. Determine available portals:

```
(config)> show firewall portal
portal1
```

```
        auth none
        enable true
        http redirect
        no interface
        no message
        no redirect_url
        no terms
        timeout 24h
        no title
(config)>
```

ii. Add a captive portal:

```
(config)> add auth group test acl portal portals end portal1
(config)>
```

6. (Optional) Configure Nagios monitoring:

```
(config)> auth group test acl nagios enable true
(config)>
```

7. (Optional) Enable users that belong to this group to access the Bluetooth scanning service:

```
(config)> auth group test acl bluetooth_scanner enable true
(config)>
```

8. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

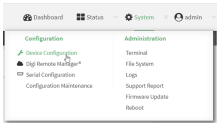
Delete an authentication group

By default, the IX15 device has two preconfigured authentication groups: **admin** and **serial**. These groups cannot be deleted.

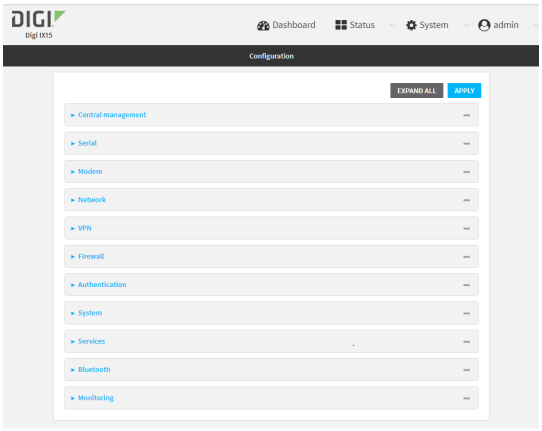
To delete an authentication group that you have created:

WebUI

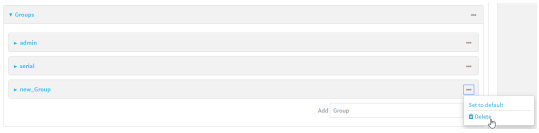
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **Authentication** > **Groups**.
4. Click the menu icon **(...)** next to the group to be deleted and select **Delete**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type:

```
(config)> del auth group groupname
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Local users

Local users are authenticated on the device without using an external authentication mechanism such as TACACS+ or RADIUS. Local user authentication is enabled by default, with one preconfiged default user.

Default user

At manufacturing time, each IX15 device comes with a default user configured as follows:

- Username: **admin**.
- Password: The default password is displayed on the label on the bottom of the device.

> **Note** The default password is a unique password for the device, and is the most critical security feature for the device. If you reset the device to factory defaults, you must log in using the default user and password, and you should immediately change the password to a custom password. Before deploying or mounting the IX15 device, record the default password, so you have the information available when you need it even if you cannot physically access the label on the bottom of the device.

The default **admin** user is preconfigured with both Admin and Serial access. You can configure the **admin** user account to fit with the needs of your environment.

This section contains the following topics:

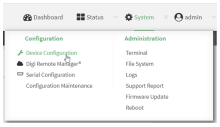
Change a local user's password 638
Configure a local user 640
Delete a local user 648

Change a local user's password

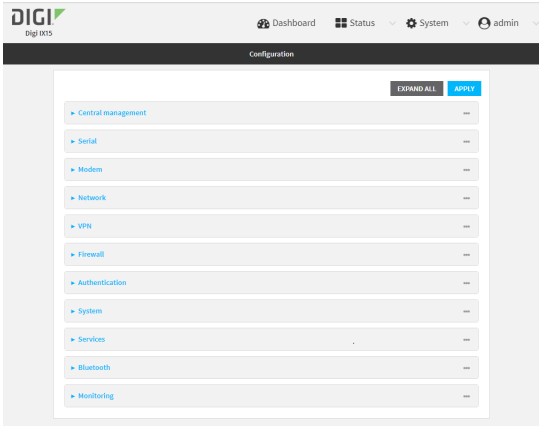
To change a user's password:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



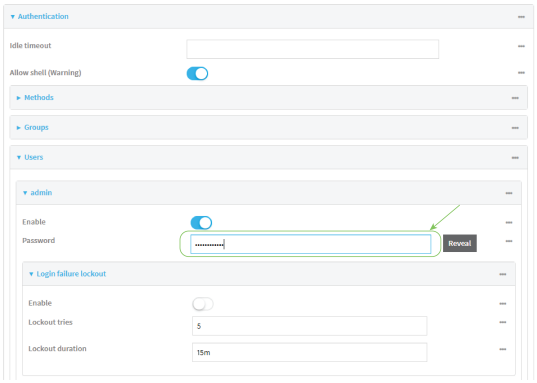
The **Configuration** window is displayed.



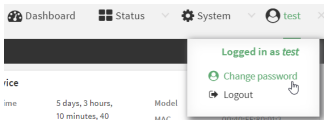
3. Click **Authentication** > **Users**.
4. Click the username to expand the user's configuration node.
5. For **Password**, enter the new password. The password must be at least eight characters long and must contain at least one uppercase letter, one lowercase letter, one number, and one special character.

   For the **admin** user, the password field can be left blank:

   - If the password field for the **admin** user is left blank, the **admin** user's password will be the default password printed on the device's label.
   - If the **admin** user's password has been changed from the default and the configuration saved, if you then clear the password field for the **admin** user, this will result in the device device's configuration being erased and reset to the default configuration.



You can also change the password for the active user by clicking the user name in the menu bar:



The active user must have full Admin access rights to be able to change the password.

6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

   ```
   > config
   (config)>
   ```
3. At the config prompt, type:

   ```
   (config)> auth user username password pwd
   ```

   Where:

   - *username* is the name of the user.
   - *pwd* is the new password for the user. The password must be at least eight characters long and must contain at least one uppercase letter, one lowercase letter, one number, and one special character.
4. Save the configuration and apply the change:

   ```
   (config)> save
   Configuration saved.
   >
   ```
5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure a local user

Required configuration items

- A username.
- A password. The password must be at least eight characters long and must contain at least one uppercase letter, one lowercase letter, one number, and one special character. For security reasons, passwords are stored in hash form. There is no way to get or display passwords in clear-text form, although prior to saving the configuration, the password can be shown by clicking **Reveal**.
- The authentication group or groups from which the user will inherit access rights. See Authentication groups for information about configuring groups.

Additional configuration items

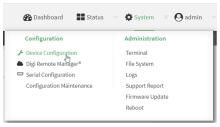
- An alias for the user. Because the username cannot contain any special characters, such as hyphens (-) or periods (.), an alias allows the user to log in using a name that contains special characters.
- The number of unsuccessful login attempts before the user is locked out of the system.

- The amount of time that the user is locked out of the system after the specified number of unsuccessful login attempts.
- An optional public ssh key, to authenticate the user when using passwordless SSH login.
- Two-factor authentication information for user login over SSH, telnet, and the serial console:
  - The verification type for two-factor authentication: Either time-based or counter-based.
  - The security key.
  - Whether to allow passcode reuse (time based verification only).
  - The passcode refresh interval (time based verification only).
  - The valid code window size.
  - The login limit.
  - The login limit period.
  - One-time use eight-digit emergency scratch codes.

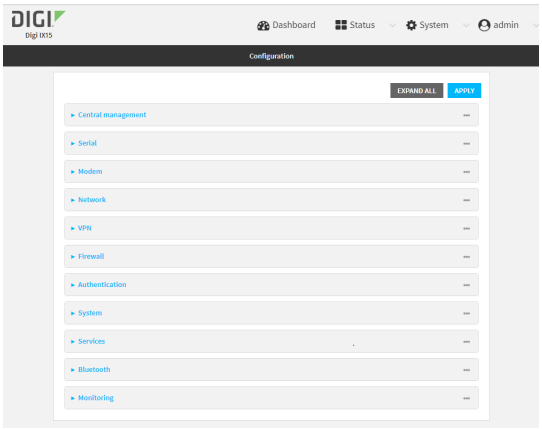
To configure a local user:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

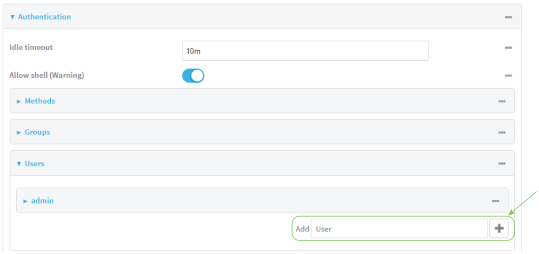


The **Configuration** window is displayed.

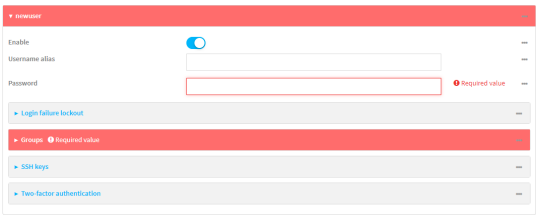


3. Click **Authentication** > **Users**.

4. In **Add User**, type a name for the user and click ✚.



The user configuration window is displayed.



The user is enabled by default. To disable, click to toggle off **Enable**.

5. (Optional) For **Username alias**, type an alias for the user.

   Because the name used to create the user and cannot contain special characters such as hyphens (-) or periods (.), an alias allows the user to log in using a name that contains special characters. For security purposes, if two users have the same alias, the alias will be disabled.

6. Enter a password for the user. The password must be at least eight characters long and must contain at least one uppercase letter, one lowercase letter, one number, and one special character.

7. Click to expand **Login failure lockout**.

   The login failure lockout feature is enabled by default. To disable, click to toggle off **Enable**.

   a. For **Lockout tries**, type the number of unsuccessful login attempts before the user is locked out of the device. The default is **5**.

   b. For **Lockout duration**, type the amount of time that the user is locked out after the number of unsuccessful login attempts defined in **Lockout tries**.

      Allowed values are any number of minutes, or seconds, and take the format ***number***{**m**|**s**}.

      For example, to set **Lockout duration** to ten minutes, enter **10m** or **600s**.

      The minimum value is 1 second, and the maximum is 15 minutes. The default is 15 minutes.

8. Add groups for the user.

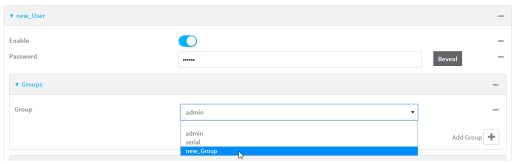
   Groups define user access rights. See Authentication groups for information about configuring groups.

a. Click to expand **Groups**.

b. For **Add Group**, click ✚.

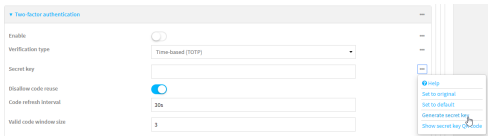


c. For **Group**, select an appropriate group.



Note Every user must be configured with at least one group. You can add multiple groups to a user by clicking **Add** again and selecting the next group.

9. (Optional) Add SSH keys for the user to use passwordless SSH login:
   a. Click **SSH keys**.
   b. In **Add SSH key**, paste or type a public encryption key that this user can use for passwordless SSH login and click ✚.
10. (Optional) Configure two-factor authentication for SSH, telnet, and serial console login:
   a. Click **Two-factor authentication**.
   b. Check **Enable** to enable two-factor authentication for this user.
   c. Select the **Verification type**:
      - **Time-based (TOTP)**: Time-based One-Time Password (TOTP) authentication uses the current time to generate a one-time password.
      - **Counter-based (HOTP)**: HMAC-based One-Time Password (HOTP) uses a counter to validate a one-time password.
   d. Generate a **Secret key**:
      i. Click **...** next to the field label and select **Generate secret key**.



      ii. Copy the secret key for use with an application or mobile device to generate passcodes.
   e. For time-based verification only, select **Disallow code reuse** to prevent a code from being used more than once during the time that it is valid.
   f. For time-based verification only, in **Code refresh interval**, type the amount of time that a code will remain valid.

      Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}. For example, to set **Code refresh interval** to ten minutes, enter **10m** or **600s**.

g. In **Valid code window size**, type the allowed number of concurrently valid codes. In cases where TOTP is being used, increasing the **Valid code window size** may be necessary when the clocks used by the server and client are not synchronized.

h. For **Login limit**, type the number of times that the user is allowed to attempt to log in during the **Login limit period**. Set **Login limit** to **0** to allow an unlimited number of login attempts during the **Login limit period**.

i. For **Login limit period**, type the amount of time that the user is allowed to attempt to log in.

   Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}. For example, to set **Login limit period** to ten minutes, enter **10m** or **600s**.

j. Scratch codes are emergency codes that may be used once, at any time. To add a scratch code:

   i. Click **Scratch codes**.

   ii. For **Add Code**, click ✚.

   iii. For **Code**, enter the scratch code. The code must be eight digits, with a minimum of 10000000.

   iv. Click ✚ again to add additional scratch codes.

11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a user. For example, to create a user named **new_user**:

```
(config)> add auth user new_user
(config auth user new_user)>
```

   The user is enabled by default. To disable the user, type:

```
(config auth user new_user)> enable false
(config auth user new_user)>
```

4. (Optional) Create a username alias for the user.

Because the name to create the user cannot contain special characters such as hyphens (-) or periods (.), an alias allows the user to log in using a name that contains special characters. For security purposes, if two users have the same alias, the alias will be disabled.

```
(config auth user new_user> username username_alias
(config auth user new_user)>
```

5. Set the user's password. The password must be at least eight characters long and must contain at least one uppercase letter, one lowercase letter, one number, and one special character.

```
(config auth user new_user> password pwd
(config auth user new_user)>
```

6. Configure login failure lockout settings:

   The login failure lockout feature is enabled by default. To disable:

```
(config auth user new_user> lockout enable false
(config auth user new_user)>
```

   a. Set the number of unsuccessful login attempts before the user is locked out of the device.

      where *value* is any integer. The minimum value is **1**, and the default value is **5**.

   b. Set the amount of time that the user is locked out after the number of unsuccessful login attempts defined in **lockout tries**:

```
(config auth user new_user> lockout duration value
(config auth user new_user)>
```

      where *value* is any number of minutes, or seconds, and takes the format ***number***{**m**|**s**}.

      For example, to set **duration** to ten minutes, enter either **10m** or **600s**:

```
(config auth user new_user)> lockout duration 600s
(config auth user new_user)>
```

      The minimum value is 1 second, and the maximum is 15 minutes. The default is 15 minutes.

7. Add groups for the user.

   Groups define user access rights. See Authentication groups for information about configuring groups.

   a. Add a group to the user. For example, to add the admin group to the user:

```
(config auth user new_user> add group end admin
(config auth user new_user)>
```

      Note Every user must be configured with at least one group.

   b. (Optional) Add additional groups by repeating the add group command:

```
(config auth user new_user> add group end serial
(config auth user new_user)>
```

To remove a group from a user:

a. Use the **show** command to determine the index number of the group to be deleted:

```
(config auth user new_user> show group
0 admin
1 serial
(config auth user new_user>
```

b. Type the following:

```
(config auth user new_user)> del group n
(config auth user new_user)>
```

Where *n* is index number of the authentication method to be deleted. For example, to delete the serial group as displayed by the example **show** command, above:

```
(config auth user new_user)> del group 1
(config auth user new_user)>
```

8. (Optional) Add SSH keys for the user to use passwordless SSH login:

a. Change to the user's ssh_key node:

```
(config auth user new_user)> ssh_key
(config auth user new_user ssh_key)>
```

b. Add the key by using the ssh_key command and pasting or typing a public encryption key that this user can use for passwordless SSH login:

```
(config auth user new_user ssh_key)> ssh_key key
(config auth user new_user ssh_key)>
```

9. (Optional) Configure two-factor authentication for SSH, telnet, and serial console login:

a. Change to the user's two-factor authentication node:

```
(config auth user new_user)> 2fa
(config auth user new_user 2fa)>
```

b. Enable two-factor authentication for this user:

```
(config auth user new_user 2fa)> enable true
(config auth user new_user 2fa)>
```

c. Configure the verification type. Allowed values are:

- **totp**: Time-based One-Time Password (TOTP) authentication uses the current time to generate a one-time password.
- **hotp**: HMAC-based One-Time Password (HOTP) uses a counter to validate a one-time password.

The default value is **totp**.

```
(config auth user new_user 2fa)> type totp
(config auth user new_user 2fa)>
```

d. Add a secret key:

```
(config auth user new_user 2fa)> secret key
(config auth user new_user 2fa)>
```

This key should be used by an application or mobile device to generate passcodes.

e. For time-based verification only, enable **disallow_reuse** to prevent a code from being used more than once during the time that it is valid.

```
(config auth user new_user 2fa)> disallow_reuse true
(config auth user new_user 2fa)>
```

f. For time-based verification only, configure the code refresh interval. This is the amount of time that a code will remain valid.

```
(config auth user new_user 2fa)> refresh_interval value
(config auth user new_user 2fa)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **refresh_interval** to ten minutes, enter either **10m** or **600s**:

```
(config auth user name 2fa)> refresh_interval 600s
(config auth user name 2fa)>
```

The default is **30s**.

g. Configure the valid code window size. This represents the allowed number of concurrently valid codes. In cases where TOTP is being used, increasing the valid code window size may be necessary when the clocks used by the server and client are not synchronized.

```
(config auth user new_user 2fa)> window_size 3
(config auth user new_user 2fa)>
```

h. Configure the login limit. This represents the number of times that the user is allowed to attempt to log in during the Login limit period. Set to 0 to allow an unlimited number of login attempts during the Login limit period

```
(config auth user new_user 2fa)> login_limit 3
(config auth user new_user 2fa)>
```

i. Configure the login limit period. This is the amount of time that the user is allowed to attempt to log in.

```
(config auth user new_user 2fa)> login_limit_period value
(config auth user new_user 2fa)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **login_limit_period** to ten minutes, enter either **10m** or **600s**:

```
(config auth user name 2fa)> login_limit_period 600s
(config auth user name 2fa)>
```

The default is **30s**.

j. Scratch codes are emergency codes that may be used once, at any time. To add a scratch code:

i. Change to the user's scratch code node:

```
(config auth user new_user 2fa)> scratch_code
(config auth user new_user 2fa scratch_code)>
```

ii. Add a scratch code:

```
(config auth user new_user 2fa scratch_code)> add end code
(config auth user new_user 2fa scratch_code)>
```

Where code is an digit number, with a minimum of 10000000.

iii. To add additional scratch codes, use the **add end *code*** command again.

10. Save the configuration and apply the change:

```
(config auth user new 2fa scratch_code)> save
Configuration saved.
>
```

11. Type **exit** to exit the Admin CLI.

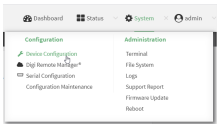
Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Delete a local user

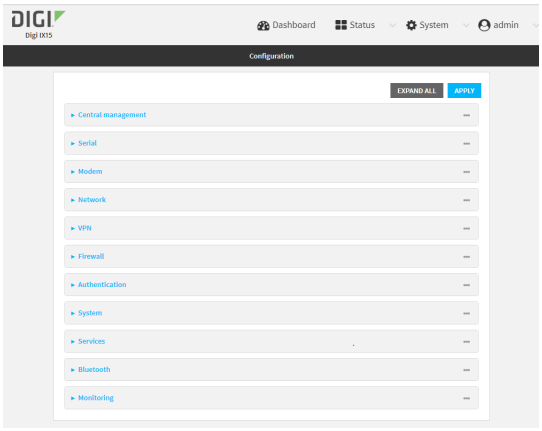
To delete a user from your IX15:

WebUI

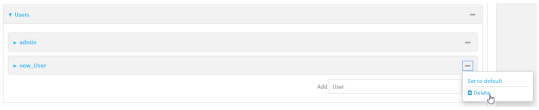
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Authentication** > **Users**.
4. Click the menu icon (**...**) next to the name of the user to be deleted and select **Delete**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type:

```
(config)> del auth user username
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Terminal Access Controller Access-Control System Plus (TACACS+)

Your IX15 device supports Terminal Access Controller Access-Control System Plus (TACACS+), a networking protocol that provides centralized authentication and authorization management for users who connect to the device. With TACACS+ support, the IX15 device acts as a TACACS+ client, which sends user credentials and connection parameters to a TACACS+ server over TCP. The TACACS+ server then authenticates the TACACS+ client requests and sends back a response message to the device.

When you are using TACACS+ authentication, you can have both local users and TACACS+ users able to log in to the device. To use TACACS+ authentication, you must set up a TACACS+ server that is accessible by the IX15 device prior to configuration. The process of setting up a TACACS+ server varies by the server environment.

This section contains the following topics:

TACACS+ user configuration 652
TACACS+ server failover and fallback to local authentication 653
Configure your IX15 device to use a TACACS+ server 653

TACACS+ user configuration

When configured to use TACACS+ support, the IX15 device uses a remote TACACS+ server for user authentication (password verification) and authorization (assigning the access level of the user). Additional TACACS+ servers can be configured as backup servers for user authentication.

This section outlines how to configure a TACACS+ server to be used for user authentication on your IX15 device.

Example TACACS+ configuration

With TACACS+, users are defined in the server configuration file. On Ubuntu, the default location and filename for the server configuration file is **/etc/tacacs+/tac_plus.conf**.

Note TACACS+ configuration, including filenames and locations, may vary depending on your platform and installation. This example assumes a Ubuntu installation.

To define users:

1. Open the TACACS+ server configuration file in a text editor. For example:

```
$ sudo gedit /etc/tacacs+/tac_plus.conf
```

2. Add users to the file using the following format. This example will create two users, one with admin and serial access, and one with only serial access.

```
user = user1 {
       name ="User1 for IX15"
       pap = cleartext password1
       service = system {
         groupname = admin,serial
       }
}
user = user2 {
       name ="User2 for IX15"
       pap = cleartext password2
       service = system {
         groupname = serial
       }
}
```

   The **groupname** attribute is optional. If used, the value must correspond to authentication groups configured on your IX15. Alternatively, if the user is also configured as a local user on the IX15 device and the LDAP server authenticates the user but does not return any groups, the local configuration determines the list of groups. See Authentication groups for more information about authentication groups. The **groupname** attribute can contain one group or multiple groups in a comma-separated list.

3. Save and close the file.
4. Verify that your changes did not introduce any syntax errors:

```
$ sudo tac_plus -C /etc/tacacs+/tac_plus.conf -P
```

   If successful, this command will echo the configuration file to standard out. If the command encounters any syntax errors, a message similar to this will display:

```
Error: Unrecognised token on line 1
```

5. Restart the TACACS+ server:

```
$ sudo /etc/init.d/tacacs_plus restart
```

TACACS+ server failover and fallback to local authentication

In addition to the primary TACACS+ server, you can also configure your IX15 device to use backup TACACS+ servers. Backup TACACS+ servers are used for authentication requests when the primary TACACS+ server is unavailable.

Falling back to local authentication

With user authentication methods, you can configure your IX15 device to use multiple types of authentication. For example, you can configure both TACACS+ authentication and local authentication, so that local authentication can be used as a fallback mechanism if the primary and backup TACACS+ servers are unavailable. Additionally, users who are configured locally but are not configured on the TACACS+ server are still able to log into the device. Authentication methods are attempted in the order they are listed until the first successful authentication result is returned; therefore if you want to ensure that users are authenticated first through the TACACS+ server, and only authenticated locally if the TACACS+ server is unavailable or if the user is not defined on the TACACS+ server, then you should list the TACACS+ authentication method prior to the Local users authentication method.

See User authentication methods for more information about authentication methods.

If the TACACS+ servers are unavailable and the IX15 device falls back to local authentication, only users defined locally on the device are able to log in. TACACS+ users cannot log in until the TACACS+ servers are brought back online.

Configure your IX15 device to use a TACACS+ server

This section describes how to configure a IX15 device to use a TACACS+ server for authentication and authorization.

Required configuration items

- Define the TACACS+ server IP address or domain name.
- Define the TACACS+ server shared secret.
- The group attribute configured in the TACACS+ server configuration.
- The service field configured in the TACACS+ server configuration.
- Add TACACS+ as an authentication method for your IX15 device.

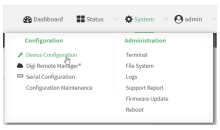
Additional configuration items

- Whether other user authentication methods should be used in addition to the TACACS+ server, or if the TACACS+ server should be considered the authoritative login method.
- Enable command authorization, so that the device will communicate with the TACACS+ server to determine if the user is authorized to execute a specific command.
- Enable command accounting, so that the device will communicate with the TACACS+ server to log commands that the user executes.

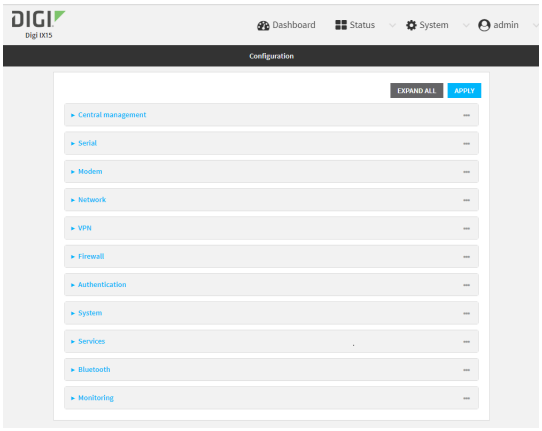
- The TACACS+ server port. It is configured to 49 by default.
- Add additional TACACS+ servers in case the first TACACS+ server is unavailable.

WebUI

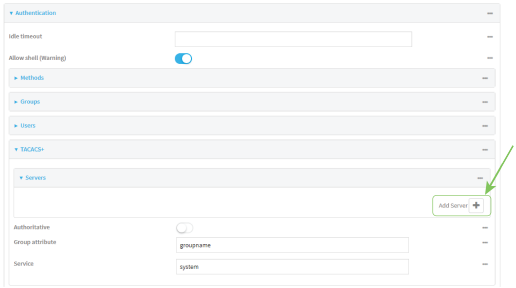
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Authentication** > **TACACS+** > **Servers**.
4. Add TACACS+ servers:
   a. For **Add server**, click ✚.

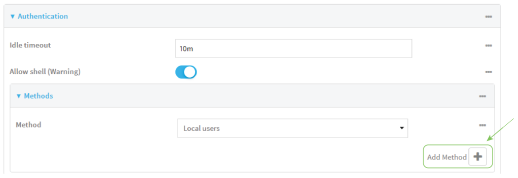


   b. For **Hostname**, type the hostname or IP address of the TACACS+ server.
   c. (Optional) Change the default **Port** setting to the appropriate port. Normally this should be left at the default setting of port 49.
   d. For **Secret**, type the TACACS+ server's shared secret. This is configured in the key parameter of the TACACS+ server's tac_plus.conf file, for example:

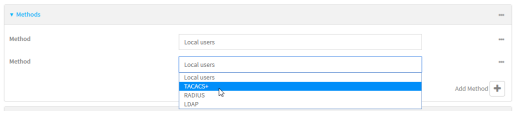
```
key = testing123
```

   e. (Optional) Click ✚ again to add additional TACACS+ servers.

5. (Optional) Enable **Authoritative** to prevent other authentication methods from being used if TACACS+ authentication fails. Other authentication methods will only be used if the TACACS+ server is unavailable.
6. (Optional) For **Group attribute**, type the name of the attribute used in the TACACS+ server's configuration to identify the IX15 authentication group or groups that the user is a member of. For example, in TACACS+ user configuration, the group attribute in the sample tac_plus.conf file is **groupname**, which is also the default setting in the IX15 configuration.
7. (Optional) For **Service**, type the value of the **service** attribute in the the TACACS+ server's configuration. For example, in TACACS+ user configuration, the value of the **service** attribute in the sample tac_plus.conf file is **system**, which is also the default setting in the IX15 configuration.
8. (Optional) Enable **Command authorization**, which instructs the device to communicate with the TACACS+ server to determine if the user is authorized to execute a specific command. Only the first configured TACACS+ server will be used for command authorization.
9. (Optional) Enable **Command accounting**, which instructs the device to communicate with the TACACS+ server to log commands that the user executes. Only the first configured TACACS+ server will be used for command accounting.
10. Add TACACS+ to the authentication methods:
    a. Click **Authentication** > **Methods**.
    b. For **Add method**, click ✚.



    c. Select **TACACS+** for the new method from the **Method** drop-down.



    Authentication methods are attempted in the order they are listed until the first successful authentication result is returned. See Rearrange the position of authentication methods for information about rearranging the position of the methods in the list.
11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. (Optional) Prevent other authentication methods from being used if TACACS+ authentication fails. Other authentication methods will only be used if the TACACS+ server is unavailable.

```
(config)> auth tacacs+ authoritative true
(config)>
```

4. (Optional) Configure the group_attribute. This is the name of the attribute used in the TACACS+ server's configuration to identify the IX15 authentication group or groups that the user is a member of. For example, in TACACS+ user configuration, the group attribute in the sample tac_plus.conf file is **groupname**, which is also the default setting for the group_attribute in the IX15 configuration.

```
(config)> auth tacacs+ group_attribute attribute-name
(config)>
```

5. (Optional) Configure the type of service. This is the value of the **service** attribute in the the TACACS+ server's configuration. For example, in TACACS+ user configuration, the value of the **service** attribute in the sample tac_plus.conf file is **system**, which is also the default setting in the IX15 configuration.

```
(config)> auth tacacs+ service service-name
(config)>
```

6. (Optional) Enable command authorization, which instructs the device to communicate with the TACACS+ server to determine if the user is authorized to execute a specific command. Only the first configured TACACS+ server will be used for command authorization.

```
(config)> auth tacacs+ command_authorization true
(config)>
```

7. (Optional) Enable command accounting, which instructs the device to communicate with the TACACS+ server to log commands that the user executes. Only the first configured TACACS+ server will be used for command accounting.

```
(config)> auth tacacs+ command_accounting true
(config)>
```

8. Add a TACACS+ server:
   a. Add the server:

```
(config)> add auth tacacs+ server end
(config auth tacacs+ server 0)>
```

   b. Enter the TACACS+ server's IP address or hostname:

```
(config auth tacacs+ server 0)> hostname hostname|ip-address
(config auth tacacs+ server 0)>
```

c. (Optional) Change the default port setting to the appropriate port:

```
(config auth tacacs+ server 0)> port port
(config auth tacacs+ server 0)>
```

d. (Optional) Repeat the above steps to add additional TACACS+ servers.

9. Add TACACS+ to the authentication methods. Authentication methods are attempted in the order they are listed until the first successful authentication result is returned. This example will add TACACS+ to the end of the list. See User authentication methods for information about adding methods to the beginning or middle of the list.

```
(config)> add auth method end tacacs+
(config)>
```

10. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

11. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Remote Authentication Dial-In User Service (RADIUS)

Your IX15 device supports Remote Authentication Dial-In User Service (RADIUS), a networking protocol that provides centralized authentication and authorization management for users who connect to the device. With RADIUS support, the IX15 device acts as a RADIUS client, which sends user credentials and connection parameters to a RADIUS server over UDP. The RADIUS server then authenticates the RADIUS client requests and sends back a response message to the device.

When you are using RADIUS authentication, you can have both local users and RADIUS users able to log in to the device. To use RADIUS authentication, you must set up a RADIUS server that is accessible by the IX15 device prior to configuration. The process of setting up a RADIUS server varies by the server environment. An example of a RADIUS server is FreeRADIUS.

This section contains the following topics:

RADIUS user configuration 659
RADIUS server failover and fallback to local configuration 659
Configure your IX15 device to use a RADIUS server 660

RADIUS user configuration

When configured to use RADIUS support, the IX15 device uses a remote RADIUS server for user authentication (password verification) and authorization (assigning the access level of the user). Additional RADIUS servers can be configured as backup servers for user authentication.

This section outlines how to configure a RADIUS server to be used for user authentication on your IX15 device.

Example FreeRADIUS configuration

With FreeRADIUS, users are defined in the **users** file in your FreeRADIUS installation. To define users:

1. Open the FreeRadius user file in a text editor. For example:

```
$ sudo gedit /etc/freeradius/3.0/users
```

2. Add users to the file using the following format:

```
user1 Cleartext-Password := "user1"
       Unix-FTP-Group-Names := "admin"

user2 Cleartext-Password := "user2"
       Unix-FTP-Group-Names := "serial"
```

   The **Unix-FTP-Group-Names** attribute is optional. If used, the value must correspond to authentication groups configured on your IX15. Alternatively, if the user is also configured as a local user on the IX15 device and the RADIUS server authenticates the user but does not return any groups, the local configuration determines the list of groups. See Authentication groups for more information about authentication groups. The **Unix-FTP-Group-Names** attribute can contain one group or multiple groups in a comma-separated list.

3. Save and close the file.
4. Verify that your changes did not introduce any syntax errors:

```
$ sudo freeradius -CX
```

   This should return a message that completes similar to:

```
...
Configuration appears to be OK
```

5. Restart the FreeRADIUS server:

```
$ sudo /etc/init.d/freeradius restart
```

RADIUS server failover and fallback to local configuration

In addition to the primary RADIUS server, you can also configure your IX15 device to use backup RADIUS servers. Backup RADIUS servers are used for authentication requests when the primary RADIUS server is unavailable.

Falling back to local authentication

With user authentication methods, you can configure your IX15 device to use multiple types of authentication. For example, you can configure both RADIUS authentication and local authentication, so that local authentication can be used as a fallback mechanism if the primary and backup RADIUS

servers are unavailable. Additionally, users who are configured locally but are not configured on the RADIUS server are still able to log into the device. Authentication methods are attempted in the order they are listed until the first successful authentication result is returned; therefore if you want to ensure that users are authenticated first through the RADIUS server, and only authenticated locally if the RADIUS server is unavailable or if the user is not defined on the RADIUS server, then you should list the RADIUS authentication method prior to the Local users authentication method.

See User authentication methods for more information about authentication methods.

If the RADIUS servers are unavailable and the IX15 device falls back to local authentication, only users defined locally on the device are able to log in. RADIUS users cannot log in until the RADIUS servers are brought back online.

Configure your IX15 device to use a RADIUS server

This section describes how to configure a IX15 device to use a RADIUS server for authentication and authorization.

Required configuration items

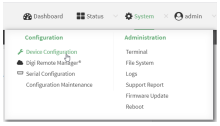
- Define the RADIUS server IP address or domain name.
- Define the RADIUS server shared secret.
- Add RADIUS as an authentication method for your IX15 device.

Additional configuration items

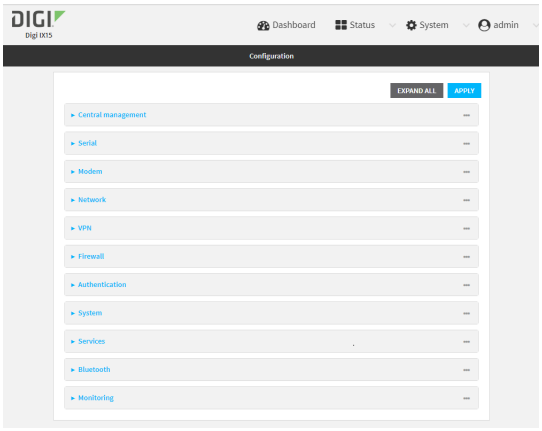
- Whether other user authentication methods should be used in addition to the RADIUS server, or if the RADIUS server should be considered the authoritative login method.
- The RADIUS server port. It is configured to 1812 by default.
- Add additional RADIUS servers in case the first RADIUS server is unavailable.
- The server NAS ID. If left blank, the default value is used:
  - If you are access the IX15 device by using the WebUI, the default value is for NAS ID is **httpd**.
  - If you are access the IX15 device by using ssh, the default value is **sshd**.
- Time in seconds before the request to the server times out. The default is 3 seconds and the maximum possible value is 60 seconds.
- Enable additional debug messages from the RADIUS client.

WebUI

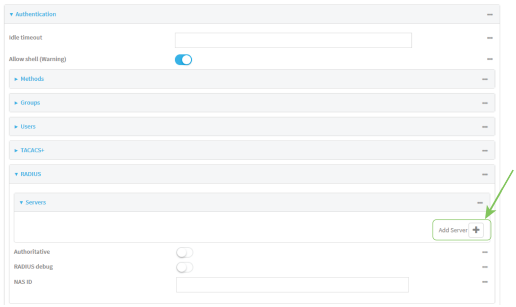
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Authentication** > **RADIUS** > **Servers**.
4. Add RADIUS servers:
   a. For **Add server**, click ✚.



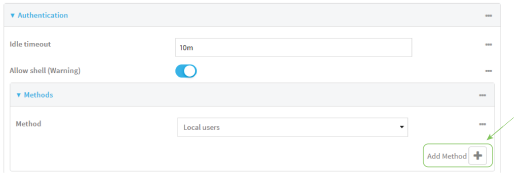
   b. For **Hostname**, type the hostname or IP address of the RADIUS server.
   c. (Optional) Change the default **Port** setting to the appropriate port. Normally this should be left at the default setting of port 1812.
   d. For **Secret**, type the RADIUS server's shared secret. This is configured in the secret parameter of the RADIUS server's client.conf file, for example:

```
secret=testing123
```

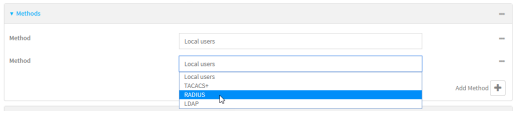
   e. For **Timeout**, type or select the amount of time in seconds to wait for the RADIUS server to respond. Allowed value is any integer from **3** to **60**. The default value is **3**.
   f. (Optional) Click ✚ again to add additional RADIUS servers.
5. (Optional) Enable **Authoritative** to prevent other authentication methods from being used if RADIUS authentication fails. Other authentication methods will only be used if the RADIUS server is unavailable.
6. (Optional) Click **RADIUS debug** to enable additional debug messages from the RADIUS client.
7. (Optional) For **NAS ID**, type the unique identifier for this network access server (NAS). You can use the fully-qualified domain name of the NAS or any arbitrary string. If not set, the default value is used:

- If you are accessing the IX15 device by using the WebUI, the default value is for NAS ID is **httpd**.
- If you are accessing the IX15 device by using ssh, the default value is **sshd**.

8. Add RADIUS to the authentication methods:
   a. Click **Authentication** > **Methods**.
   b. For **Add method**, click ✚.



   c. Select **RADIUS** for the new method from the **Method** drop-down.



   Authentication methods are attempted in the order they are listed until the first successful authentication result is returned. See Rearrange the position of authentication methods for information about rearranging the position of the methods in the list.

9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. (Optional) Prevent other authentication methods from being used if RADIUS authentication fails. Other authentication methods will only be used if the RADIUS server is unavailable.

```
(config)> auth radius authoritative true
(config)>
```

4. (Optional) Enable debug messages from the RADIUS client:

```
(config)> auth radius debug true
(config)>
```

5. (Optional) Configure the NAS ID. This is a unique identifier for this network access server (NAS). You can use the fully-qualified domain name of the NAS or any arbitrary string. If not set, the default value is used:
   - If you are accessing the IX15 device by using the WebUI, the default value is for NAS ID is **httpd**.
   - If you are accessing the IX15 device by using ssh, the default value is **sshd**.

```
(config)> auth radius nas_id id
(config)>
```

6. Add a RADIUS server:
   a. Add the server:

```
(config)> add auth radius server end
(config auth radius server 0)>
```

   b. Enter the RADIUS server's IP address or hostname:

```
(config auth radius server 0)> hostname hostname|ip-address
(config auth radius server 0)>
```

   c. (Optional) Change the default port setting to the appropriate port:

```
(config auth radius server 0)> port port
(config auth radius server 0)>
```

   d. Configure the amount of time in seconds to wait for the RADIUS server to respond. Allowed value is any integer from **3** to **60**. The default value is **3**.

```
(config auth radius server 0)> timeout value
(config auth radius server 0)>
```

   e. (Optional) Repeat the above steps to add additional RADIUS servers.
7. Add RADIUS to the authentication methods. Authentication methods are attempted in the order they are listed until the first successful authentication result is returned. This example will add RADIUS to the end of the list. See User authentication methods for information about adding methods to the beginning or middle of the list.

```
(config)> add auth method end radius
(config)>
```

8. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

LDAP

Your IX15 device supports LDAP (Lightweight Directory Access Protocol), a protocol used for directory information services over an IP network. LDAP can be used with your IX15 device for centralized authentication and authorization management for users who connect to the device. With LDAP support, the IX15 device acts as an LDAP client, which sends user credentials and connection parameters to an LDAP server. The LDAP server then authenticates the LDAP client requests and sends back a response message to the device.

When you are using LDAP authentication, you can have both local users and LDAP users able to log in to the device. To use LDAP authentication, you must set up a LDAP server that is accessible by the IX15 device prior to configuration. The process of setting up a LDAP server varies by the server environment.

This section contains the following topics:

LDAP user configuration 665
LDAP server failover and fallback to local configuration 666
Configure your IX15 device to use an LDAP server 666

LDAP user configuration

When configured to use LDAP support, the IX15 device uses a remote LDAP server for user authentication (password verification) and authorization (assigning the access level of the user). Additional LDAP servers can be configured as backup servers for user authentication.

This section outlines how to configure a LDAP server to be used for user authentication on your IX15 device.

There are several different implementations of LDAP, including Microsoft Active Directory. This section uses OpenLDAP as an example configuration. Other implementations of LDAP will have different configuration methods.

Example OpenLDAP configuration

With OpenLDAP, users can be configured in a text file using the LDAP Data Interchange Format (LDIF). In this case, we will be using a file called **add_user.ldif**.

1. Create the **add_user.ldif** file in a text editor. For example:

```
$ gedit ./add_user.ldif
```

2. Add users to the file using the following format:

```
dn: uid=john,dc=example,dc=com
objectClass: inetOrgPerson
cn: John Smith
sn: Smith
uid: john
userPassword: password
ou: admin serial
```

   - The value of **uid** and **userPassword** must correspond to the username and password used to log into the IX15 device.
   - The **ou** attribute is optional. If used, the value must correspond to authentication groups configured on your IX15. Alternatively, if the user is also configured as a local user on the IX15 device and the LDAP server authenticates the user but does not return any groups, the local configuration determines the list of groups. See Authentication groups for more information about authentication groups.

   Other attributes may be required by the user's objectClass. Any objectClass may be used as long it allows the **uid**, **userPassword**, and **ou** attributes.

3. Save and close the file.
4. Add the user to the OpenLDAP server:

```
$ ldapadd -x -H 'ldap:///' -D 'cn=admin,dc=example,dc=com' -W -f add_
user.ldif
adding new entry "uid=john,dc=example,dc=com"
```

5. Verify that the user has been added by performing an LDAP search:

```
$ ldapsearch -x -LLL -H 'ldap:///' -b 'dc=example,dc=com'
uid=john
dn: uid=john,dc=example,dc=com
objectClass: inetOrgPerson
```

```
cn: John Smith
sn: Smith
uid: john
ou: admin serial
```

LDAP server failover and fallback to local configuration

In addition to the primary LDAP server, you can also configure your IX15 device to use backup LDAP servers. Backup LDAP servers are used for authentication requests when the primary LDAP server is unavailable.

Falling back to local authentication

With user authentication methods, you can configure your IX15 device to use multiple types of authentication. For example, you can configure both LDAP authentication and local authentication, so that local authentication can be used as a fallback mechanism if the primary and backup LDAP servers are unavailable. Additionally, users who are configured locally but are not configured on the LDAP server are still able to log into the device. Authentication methods are attempted in the order they are listed until the first successful authentication result is returned; therefore if you want to ensure that users are authenticated first through the LDAP server, and only authenticated locally if the LDAP server is unavailable or if the user is not defined on the LDAP server, then you should list the LDAP authentication method prior to the Local users authentication method.

See User authentication methods for more information about authentication methods.

If the LDAP servers are unavailable and the IX15 device falls back to local authentication, only users defined locally on the device are able to log in. LDAP users cannot log in until the LDAP servers are brought back online.

Configure your IX15 device to use an LDAP server

This section describes how to configure a IX15 device to use an LDAP server for authentication and authorization.

Required configuration items

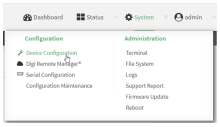
- Define the LDAP server IP address or domain name.
- Add LDAP as an authentication method for your IX15 device.

Additional configuration items

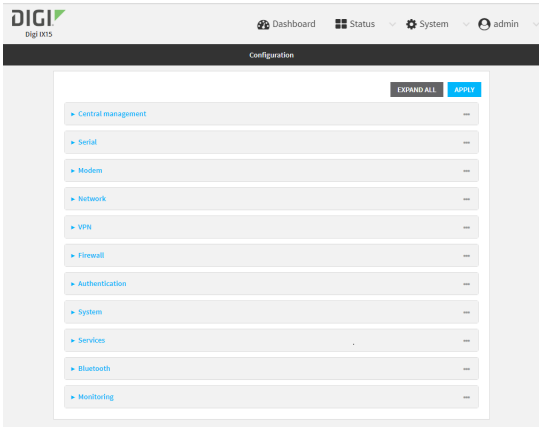
- Whether other user authentication methods should be used in addition to the LDAP server, or if the LDAP server should be considered the authoritative login method.
- The LDAP server port. It is configured to 389 by default.
- Whether to use Transport Layer Security (TLS) when communicating with the LDAP server.
- The distinguished name (DN) and password used to communicate with the server.
- The distinguished name used to search to user base.
- The group attribute.
- The number of seconds to wait to receive a message from the server.
- Add additional LDAP servers in case the first LDAP server is unavailable.

WebUI

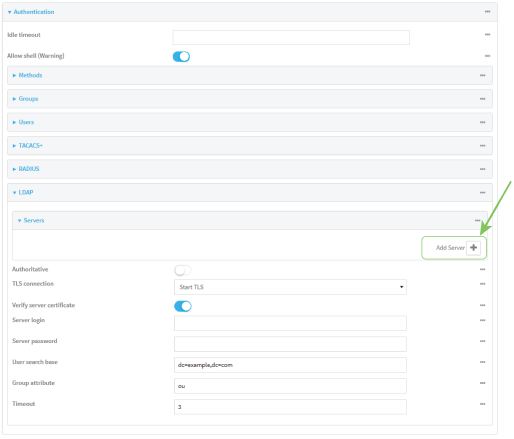
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.

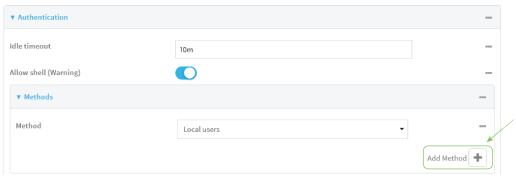


3. Click **Authentication** > **LDAP** > **Servers**.
4. Add LDAP servers:
   a. For **Add server**, click ✚.

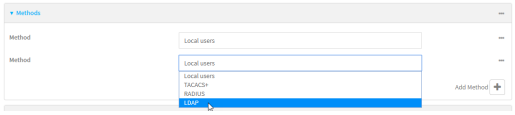


   b. For **Hostname**, type the hostname or IP address of the LDAP server.
   c. (Optional) Change the default **Port** setting to the appropriate port. Normally this should be left at the default setting of port 389 for non-TLS and 636 for TLS.
   d. (Optional) Click ✚ again to add additional LDAP servers.
5. (Optional) Enable **Authoritative** to prevent other authentication methods from being used if LDAP authentication fails. Other authentication methods will only be used if the LDAP server is unavailable.

6. For **TLS connection**, select the type of TLS connection used by the server:
   - **Disable TLS**: Uses a non-secure TCP connection on the LDAP standard port, 389.
   - **Enable TLS**: Uses an SSL/TLS encrypted connection on port 636.
   - **Start TLS**: Makes a non-secure TCP connection to the LDAP server on port 389, then sends a request to upgrade the connection to a secure TLS connection. This is the preferred method for LDAP.
7. If **Enable TLS** or **Start TLS** are selected for **TLS connection**:
   - Leave **Verify server certificate** at the default setting of enabled to verify the server certificate with a known Certificate Authority.
   - Disable **Verify server certificate** if the server is using a self-signed certificate.
8. (Optional) For **Server login**, type a distinguished name (DN) that is used to bind to the LDAP server and search for users, for example **cn=user,dc=example,dc=com**. Leave this field blank if the server allows anonymous connections.
9. (Optional) For **Server password**, type the password used to log into the LDAP server. Leave this field blank if the server allows anonymous connections.
10. For **User search base**, type the distinguished name (DN) on the server to search for users. This can be the root of the directory tree (for example, **dc=example,dc=com**) or a sub-tree (for example. **ou=People,dc=example,dc=com**).
11. For **Login attribute**, enter the user attribute containing the login of the authenticated user. For example, in the LDAP user configuration, the login attribute is **uid**. If this attribute is not set, the user will be denied access.
12. (Optional) For **Group attribute**, type the name of the user attribute that contains the list of IX15 authentication groups that the authenticated user has access to. See LDAP user configuration for further information about the group attribute.
13. For **Timeout**, type or select the amount of time in seconds to wait for the LDAP server to respond. Allowed value is between **3** and **60** seconds.
14. Add LDAP to the authentication methods:
    a. Click **Authentication** > **Methods**.
    b. For **Add method**, click ✚.



    c. Select **LDAP** for the new method from the **Method** drop-down.



    Authentication methods are attempted in the order they are listed until the first successful authentication result is returned. See Rearrange the position of authentication methods for information about rearranging the position of the methods in the list.
15. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. (Optional) Prevent other authentication methods from being used if LDAP authentication fails. Other authentication methods will only be used if the LDAP server is unavailable.

```
(config)> auth ldap authoritative true
(config)>
```

4. Set the type of TLS connection used by the LDAP server:

```
(config)> auth ldap tls value
(config)>
```

   where *value* is one of:

   - **off**: Uses a non-secure TCP connection on the LDAP standard port, 389.
   - **on**: Uses an SSL/TLS encrypted connection on port 636.
   - **start_tls**: Makes a non-secure TCP connection to the LDAP server on port 389, then sends a request to upgrade the connection to a secure TLS connection. This is the preferred method for LDAP.

   The default is **off**.
5. If **tls** is set to **on** or **start_tls**, configure whether to verify the server certificate:

```
(config)> auth ldap verify_server_cert value
(config)>
```

   where *value* is either:

   - **true**: Verifies the server certificate with a known Certificate Authority.
   - **false**: Does not verify the certificate. Use this option if the server is using a self-signed certificate.

   The default is **true**.
6. Set the distinguished name (DN) that is used to bind to the LDAP server and search for users. Leave this option unset if the server allows anonymous connections.

```
(config)> auth ldap bind_dn dn_value
(config)>
```

For example:

```
(config)> auth ldap bind_dn cn=user,dc=example,dc=com
(config)>
```

7. Set the password used to log into the LDAP server. Leave this option unset if the server allows anonymous connections.

```
(config)> auth ldap bind_password password
(config)>
```

8. Set the distinguished name (DN) on the server to search for users. This can be the root of the directory tree (for example, **dc=example,dc=com**) or a sub-tree (for example. **ou=People,dc=example,dc=com**).

```
(config)> auth ldap base_dn value
(config)>
```

9. Set the login attribute:

```
(config)> auth ldap login_attribute value
(config)>
```

where value is the user attribute containing the login of the authenticated user. For example, in the LDAP user configuration, the login attribute is **uid**. . If this attribute is not set, the user will be denied access.

10. (Optional) Set the name of the user attribute that contains the list of IX15 authentication groups that the authenticated user has access to. See LDAP user configuration for further information about the group attribute.

```
(config)> auth ldap group_attribute value
(config)>
```

For example:

```
(config)> auth ldap group_attribute ou
(config)>
```

11. Configure the amount of time in seconds to wait for the LDAP server to respond.

```
(config)> auth ldap timeout value
(config)>
```

where *value* is any integer from **3** to **60**. The default value is **3**.

12. Add an LDAP server:
    a. Add the server:

```
(config)> add auth ldap server end
(config auth ldap server 0)>
```

b. Enter the LDAP server's IP address or hostname:

```
(config auth ldap server 0)> hostname hostname|ip-address
(config auth ldap server 0)>
```

c. (Optional) Change the default port setting to the appropriate port:

```
(config auth ldap server 0)> port port
(config auth ldap server 0)>
```

d. (Optional) Repeat the above steps to add additional LDAP servers.

13. Add LDAP to the authentication methods. Authentication methods are attempted in the order they are listed until the first successful authentication result is returned. This example will add LDAP to the end of the list. See User authentication methods for information about adding methods to the beginning or middle of the list.

```
(config)> add auth method end ldap
(config)>
```

14. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

15. Type **exit** to exit the Admin CLI.

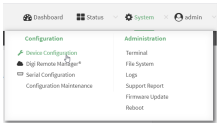
    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure serial authentication

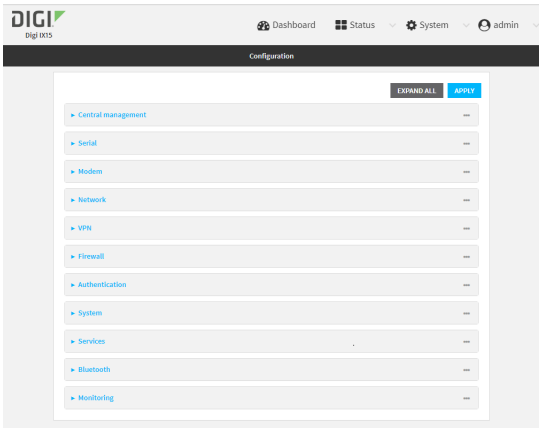
This section describes how to configure authentication for serial access.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Authentication** > **Serial**.
4. (Optional) For **TLS identity certificate**, paste a TLS certificate and private key in PEM format. If empty, the certificate for the web administration service is used. See Configure the web administration service for more information.
5. For **Peer authentication**, select the method used to verify the certificate of a remote peer.
6. **Include standard CAs** is enabled by default. This allows peers with certificates that have been signed by standard Certificate Authorities (CAs) to authenticate.
7. Click to expand **Custom certificate authorities** to add the public certificates of custom CAs.
   a. For **Add CA certificate**, type the name of a custom CA and click ✚.
   b. Paste the public certificate for the custom CA in PEM format.
   c. Repeat for additional custom CA certificates.
8. Click to expand **Peer certificates** to add the public certificates of trusted peers.
   a. For **Add Peer certificate**, type the name of a trusted peer and click ✚.
   b. Paste the public certificate for the trusted peer in PEM format.
   c. Repeat for additional trusted peer certificates.
9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. (Optional) Paste a TLS certificate and private key in PEM format:

```
(config)> auth serial identiy "cert-and-private-key"
(config)>
```

4. Set the method used to verify the certificate of a remote peer:

```
(config)> auth serial verify value
(config)>
```

where *value* is either:

- **ca**: Uses certificate authorities (CAs) to verify.
- **peer**: Uses the remote peer's public certificate to verify.

5. By default, peers with certificates that have been signed by standard Certificate Authorities (CAs) are allowed to authenticate. To disable:

```
(config)> auth serial ca_standard false
(config)>
```

6. Add the public certificate for a custom certificate authority:

```
(config)> add auth serial ca_certs CA-cert-name "cert-and-private-key"
(config)>
```

where:

- *CA-cert-name* is the name of the certificate for the custom certificate authority.
- *cert-and-private-key* is the certificate and private key for the custom certificate authority.

Repeat for additional custom certificate authorities.

7. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

8. Type **exit** to exit the Admin CLI.

Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

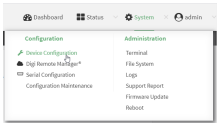
Disable shell access

To prohibit access to the shell prompt for all authentication groups, disable the **Allow shell** parameter.. This does not prevent access to the Admin CLI.

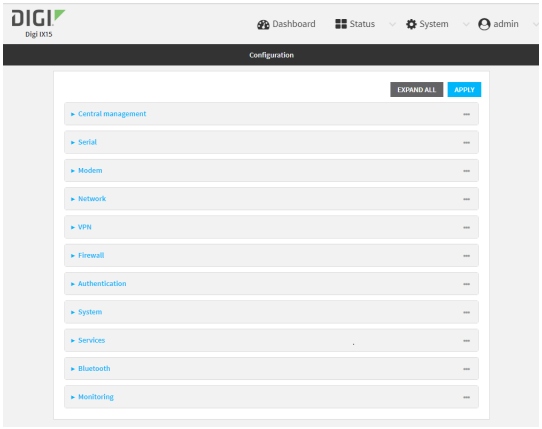
Note If shell access is disabled, re-enabling it will erase the device's configuration and perform a factory reset.

WebUI

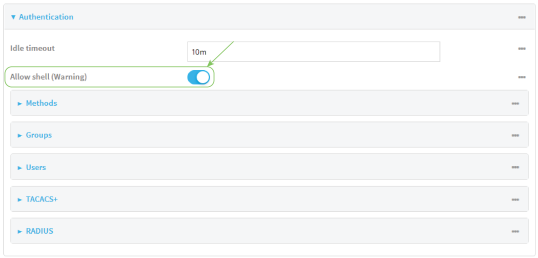
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Authentication**.
4. Click to disable **Allow shell**.



Note If shell access is disabled, re-enabling it will erase the device's configuration and perform a factory reset.

5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Set the **allow_shell** parameter to **false**:

```
(config)> auth allow_shell false
```

Note If shell access is disabled, re-enabling it will erase the device's configuration and perform a factory reset.

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

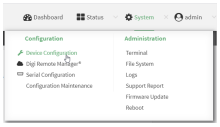
Set the idle timeout for IX15 users

To configure the amount of time that the user's active session can be inactive before it is automatically disconnected, set the **Idle timeout** parameter.

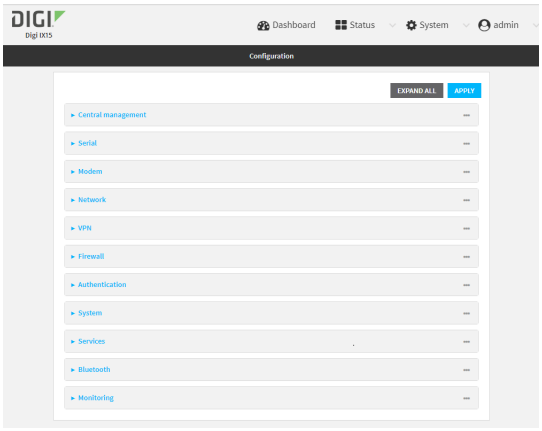
By default, the Idle timeout is set to 10 minutes.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



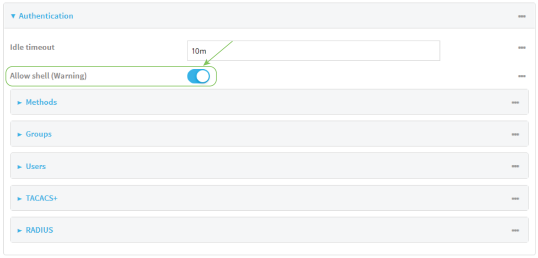
The **Configuration** window is displayed.



3. Click **Authentication**.
4. For **Idle timeout**, enter the amount of time that the active session can be idle before the user is automatically logged out.

   Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

   For example, to set **Idle timeout** to ten minutes, enter **10m** or **600s**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type:

```
(config)# auth idle_timeout value
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **idle_timeout** to ten minutes, enter either **10m** or **600s**:

```
(config)> auth idle_timeout  600s
(config)>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

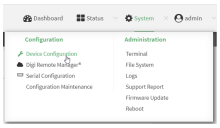
Example user configuration

Example 1: Administrator user with local authentication

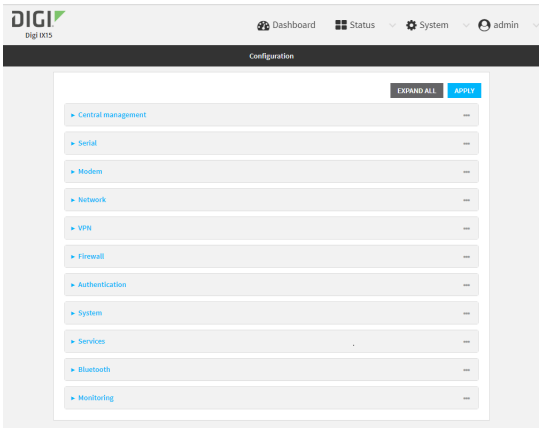
Goal: To create a user with administrator rights who is authenticated locally on the device.

WebUI

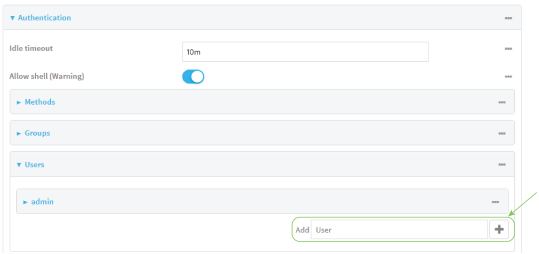
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



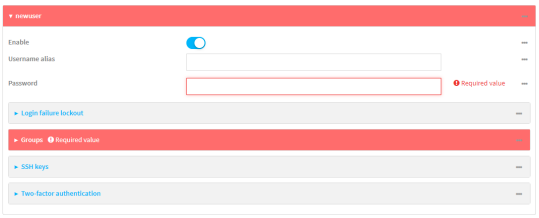
The **Configuration** window is displayed.



3. Click **Authentication** > **Users**.
4. In **Add User:** enter a name for the user and click ✚.



The user configuration window is displayed.



5. Enter a **Password** for the user.
6. Assign the user to the **admin** group:
   a. Click **Groups**.
   b. For **Add Group**, click ✚.
   c. For **Group**, select the **admin** group.
   d. Verify that the **admin** group has full administrator rights:
      i. Click **Authentication** > **Groups**.
      ii. Click **admin**.
      iii. Verify that the admin group has **Admin access** enabled. If not, click **Admin access** to enable.
      iv. Verify that **Access level** is set to **Full access**. If not, select **Full access**.
   e. Verify that **Local users** is one of the configured authentication methods:
      i. Click **Authentication** > **Methods**.
      ii. Verify that **Local users** is one of the methods listed in the list. If not:
         i. For **Add Method**, click ✚.
         ii. For **Method**, select **Local users**.
7. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Verify that the **admin** group has full administrator rights:

```
(config)> show auth group admin acl
admin
        enable true
```

```
       level full
...
(config)>
```

If **admin** > **enable** is set to false:

```
(config)> auth group admin acl admin enable true
(config)>
```

If **admin** > **level** is set to read-only:

```
(config)> auth group admin acl admin level full
(config)>
```

4. Verify that **local** is one of the configured authentication methods:

```
(config)> show auth method
0 local
(config)>
```

If **local** is not listed:

```
(config)> add auth method end local
(config)>
```

5. Create the user. In this example, the user is being created with the username **adminuser**:

```
(config)> add auth user adminuser
(config auth user adminuser)>
```

6. Assign a password to the user:

```
(config auth user adminuser)> password pwd
(config auth user adminuser)>
```

7. Assign the user to the **admin** group:

```
(config auth user adminuser)> add group end admin
(config auth user adminuser)>
```

8. Save the configuration and apply the change:

```
(config auth user adminuser)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Example 2: RADIUS, TACACS+, and local authentication for one user

Goal: To create a user with administrator rights who is authenticated by using all three authentication methods.

In this example, when the user attempts to log in to the IX15 device, user authentication will occur in the following order:

1. The user is authenticated by the RADIUS server. If the RADIUS server is unavailable,
2. The user is authenticated by the TACACS+ server. If both the RADIUS and TACACS+ servers are unavailable,
3. The user is authenticated by the IX15 device using local authentication.

This example uses a FreeRadius 3.0 server running on ubuntu, and a TACACS+ server running on ubuntu. Server configuration may vary depending on the platforms or type of servers used in your environment.

WebUI

1. Configure a user on the RADIUS server:
   a. On the ubuntu machine hosting the FreeRadius server, open the **/etc/freeradius/3.0/users** file:

   ```
   $ sudo gedit /etc/freeradius/3.0/users
   ```

   b. Add a RADIUS user to the **users** file:

   ```
   admin1 Cleartext-Password := "password1"
          Unix-FTP-Group-Names := "admin"
   ```

   In this example:
   - The user's username is **admin1**.
   - The user's password is **password1**.
   - The authentication group on the IX15 device, **admin**, is identified in the **Unix-FTP-Group-Names** parameter.

   c. Save and close the **users** file.
2. Configure a user on the TACACS+ server:
   a. On the ubuntu machine hosting the TACACS+ server, open the **/etc/tacacs+/tac_plus.conf** file:

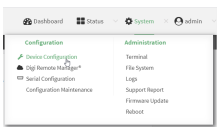
   ```
   $ sudo gedit /etc/tacacs+/tac_plus.conf
   ```

   b. Add a TACACS+ user to the **tac_plus.conf** file:

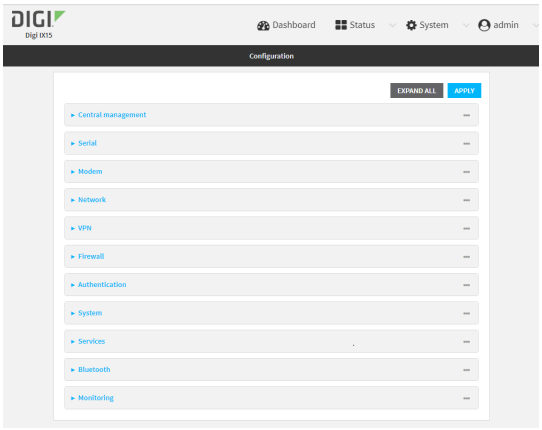
   ```
   user = admin1 {
          name ="Admin1 for TX64"
          pap = cleartext password1
          service = system {
                  groupname = admin
                  }
          }
   }
   ```

   In this example:
   - The user's username is **admin1**.
   - The user's password is **password1**.
   - The authentication group on the IX15 device, **admin**, is identified in the **groupname** parameter.

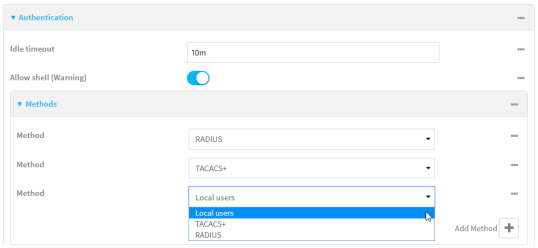
   c. Save and close the **tac_plus.conf** file.
3. Log into the IX15 WebUI as a user with full Admin access rights.
4. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



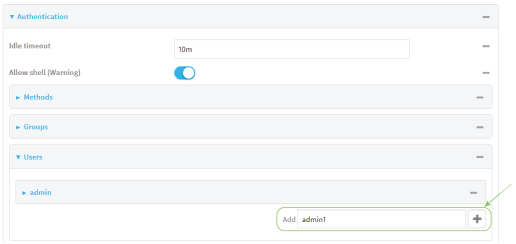
The **Configuration** window is displayed.



5. Configure the authentication methods:
   a. Click **Authentication** > **Methods**.
   b. For **Method**, select **RADIUS**.
   c. For **Add Method**, click ✚ to add a new method.
   d. For the new method, select **TACACS+**.
   e. Click ✚ to add another new method.
   f. For the new method, select **Local users**.

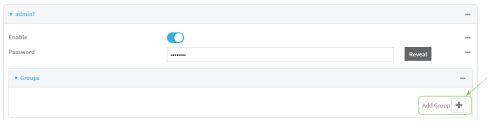


6. Create the local user:
   a. Click **Authentication** > **Users**.
   b. In **Add User:**, type **admin1** and click ✚.

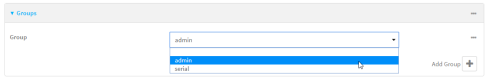


   c. For **password**, type **password1**.

d. Assign the user to the **admin** group:
   i. Click **Groups**.
   ii. For **Add Group**, click ✚.



   iii. For **Group**, select the **admin** group.



a. Verify that the **admin** group has full administrator rights:
   i. Click **Authentication** > **Groups**.
   ii. Click **admin**.
   iii. Verify that the admin group has **Admin access** enabled. If not, click **Admin access** to enable.
   iv. Verify that **Access level** is set to **Full access**. If not, select **Full access**.

7. Click **Apply** to save the configuration and apply the change.



Command line

1. Configure a user on the RADIUS server:
   a. On the ubuntu machine hosting the FreeRadius server, open the **/etc/freeradius/3.0/users** file:

```
$ sudo gedit /etc/freeradius/3.0/users
```

   b. Add a RADIUS user to the **users** file:

```
admin1 Cleartext-Password := "password1"
       Unix-FTP-Group-Names := "admin"
```

   In this example:
   - The user's username is **admin1**.
   - The user's password is **password1**.
   - The authentication group on the IX15 device, **admin**, is identified in the **Unix-FTP-Group-Names** parameter.

   c. Save and close the **users** file.

2. Configure a user on the TACACS+ server:
   a. On the ubuntu machine hosting the TACACS+ server, open the **/etc/tacacs+/tac_plus.conf** file:

   ```
   $ sudo gedit /etc/tacacs+/tac_plus.conf
   ```

   b. Add a TACACS+ user to the **tac_plus.conf** file:

   ```
   user = admin1 {
          name ="Admin1 for TX64"
          pap = cleartext password1
          service = system {
                  groupname = admin
                  }
          }
   }
   ```

   In this example:
   - The user's username is **admin1**.
   - The user's password is **password1**.
   - The authentication group on the IX15 device, **admin**, is identified in the **groupname** parameter.

   c. Save and close the **tac_plus.conf** file.
3. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
4. At the command line, type **config** to enter configuration mode:

   ```
   > config
   (config)>
   ```

5. Configure the authentication methods:
   a. Determine the current authentication method configuration:

   ```
   (config)> show auth method
   0 local
   (config)>
   ```

   This output indicates that on this example system, only local authentication is configured.

   b. Add RADIUS authentication to the beginning of the list:

   ```
   (config)> add auth method 0 radius
   (config)>
   ```

   c. Add TACACS+ authentication second place in the list:

   ```
   (config)> add auth method 1 tacacs+(config)>
   ```

d. Verify that authentication will occur in the correct order:

```
(config)> show auth method
0 radius
1 tacacs+
2 local
(config)>
```

6. Verify that the **admin** group has full administrator rights:

```
(config)> show auth group admin acl
admin
       enable true
       level full
...
(config)>
```

If **admin** > **enable** is set to false:

```
(config)> auth group admin acl admin enable true
(config)>
```

If **admin** > **level** is set to read-only:

```
(config)> auth group admin acl admin level full
(config)>
```

7. Configure the local user:
   a. Create a local user with the username **admin1**:

```
(config)> add auth user admin1
(config auth user admin1)>
```

   b. Assign a password to the user:

```
(config auth user adminuser)> password password1
(config auth user adminuser)>
```

   c. Assign the user to the **admin** group:

```
(config auth user adminuser)> add group end admin
(config auth user adminuser)>
```

8. Save the configuration and apply the change:

```
(config auth user adminuser)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Firewall

This chapter contains the following topics:

Firewall configuration 688
Port forwarding rules 693
Packet filtering 701
Configure custom firewall rules 709
Configure Quality of Service options 711

Firewall configuration

Firewall configuration includes the following configuration options:

- **Zones**: A zone is a firewall access group to which network interfaces can be added. You then use zones to configure packet filtering and access control lists for interfaces that are included in the zone. Preconfigured zones include:
  - **Any**: Matches any network interface, even if they are not assigned to this zone.
  - **Loopback**: Zone for interfaces that are used for communication between processes running on the device.
  - **Internal**: Used for interfaces connected to trusted networks. By default, the firewall will allow most access from this zone.
  - **External**: Used for interfaces to connect to untrusted zones, such as the internet. This zone has Network Address Translation (NAT) enabled by default. By default, the firewall will block most access from this zone.
  - **Edge**: Used for interfaces connected to trusted networks, where the device is a client on the edge of the network rather than a router or gateway.
  - **Setup**: Used for interfaces involved in the initial setup of the device. By default, the firewall will only allow this zone to access administration services.
  - **IPsec**: The default zone for IPsec tunnels.
  - **Dynamic routes**: Used for routes learned using routing services.
- **Port forwarding**: A list of rules that allow network connections to the IX15 to be forwarded to other servers by translating the destination address.
- **Packet filtering**: A list of packet filtering rules that determine whether to accept or reject network connections that are forwarded through the IX15.
- **Custom rules**: A script that is run to install advanced firewall rules beyond the scope/capabilities of the standard device configuration.
- **Quality Of Service**: Quality of Service (QOS) options for bandwidth allocation and policy-based traffic shaping and prioritizing.

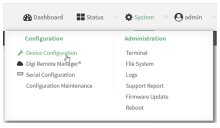
Create a custom firewall zone

In addition to the preconfigured zones, you can create your custom zones that can be used to configure packet filtering and access control lists for network interfaces.

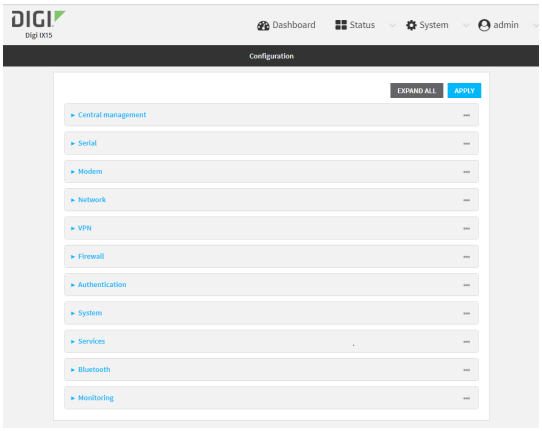
To create a zone:

WebUI

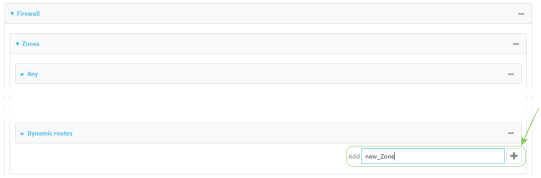
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



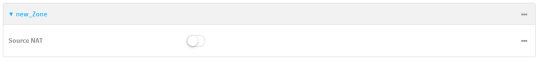
The **Configuration** window is displayed.



3. Click **Firewall** > **Zones**.
4. In **Add Zone**, enter a name for the zone and click ✚.



The firewall configuration window is displayed.



5. (Optional) If traffic on this zone will be forwarded from a private network to the internet, enable Network Address Translation (NAT).
6. Click **Apply** to save the configuration and apply the change.



See Configure the firewall zone for a network interface for information about how to configure network interfaces to use a zone.

Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add the new zone. For example, to add a zone named **my_zone**:

```
(config)> add firewall zone my_zone
(config firewall zone my_zone)>
```

4. (Optional) Enable Network Address Translation (NAT):

```
(config firewall zone my_zone)> src_nat true
(config firewall zone my_zone)>
```

5. Save the configuration and apply the change:

```
(config firewall zone my_zone)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

See Configure the firewall zone for a network interface for information about how to configure network interfaces to use a zone.

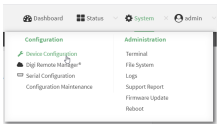
Configure the firewall zone for a network interface

Firewall zones allow you to group network interfaces for the purpose of packet filtering and access control. There are several preconfigured firewall zones, and you can create custom zones as well. The firewall zone that a network interfaces uses is selected during interface configuration.

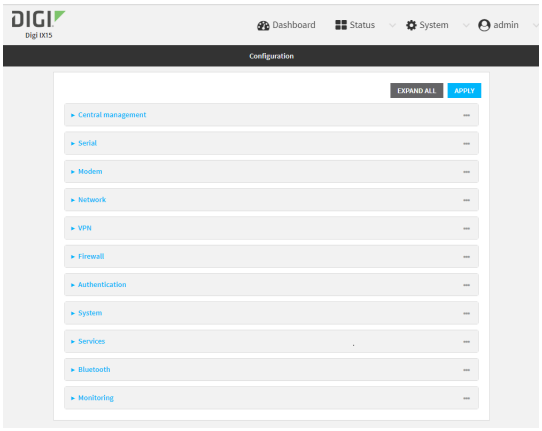
This example procedure uses an existing network interface named **ETH** and changes the firewall zone from the default zone, **Internal**, to **External**.

WebUI

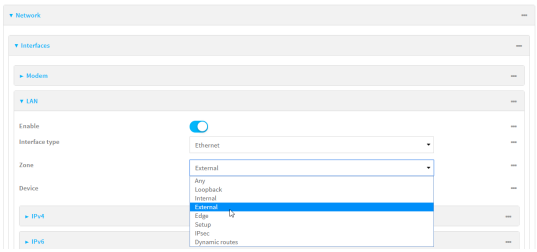
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **Network** > **Interfaces** > **ETH**.
4. For **Zone**, select **External**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type:

```
(config)> network interface eth zone my_zone
(config)>
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

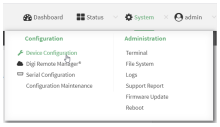
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Delete a custom firewall zone

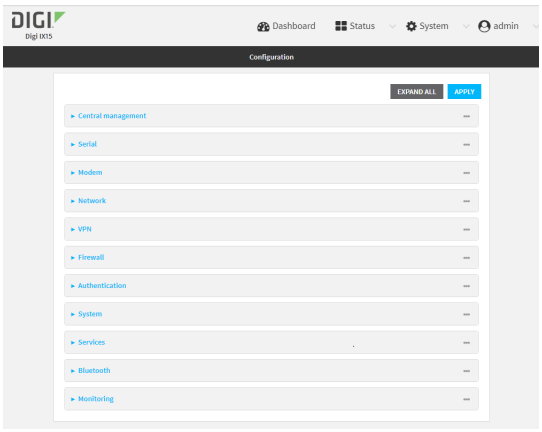
You cannot delete preconfigured firewall zones. To delete a custom firewall zone:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **Firewall** > **Zones**.
4. Click the menu icon (**...**) next to the appropriate custom firewall zone and select **Delete**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Use the **del** command to delete a custom firewall rule. For example:

```
(config)> del firewall zone my_zone
```

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Port forwarding rules

Most computers are protected by a firewall that prevents users on a public network from accessing servers on the private network. To allow a computer on the Internet to connect to a specific server on a private network, set up one or more port forwarding rules. Port forwarding rules provide mapping instructions that direct incoming traffic to the proper device on a LAN.

Configure port forwarding

Required configuration items

- The network interface for the rule.

  Network connections will only be forwarded if their destination address matches the IP address of the selected network interface.
- The public-facing port number that network connections must use for their traffic to be forwarded.
- The IP address of the server to which traffic should be forwarded.
- The port or range of ports to which traffic should be forwarded.

Additional configuration items

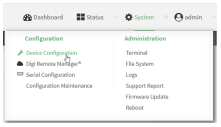
- A label for the port forwarding rule.
- The IP version (either IPv4 or IPv6) that incoming network connections must match.
- The protocols that incoming network connections must match.

- A white list of devices, based on either IP address or firewall zone, that are authorized to leverage this forwarding rule.

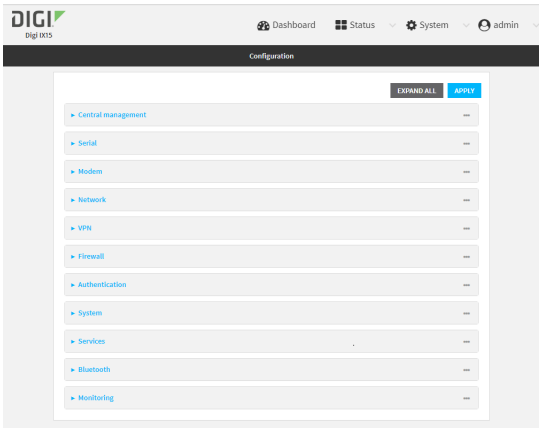
To configure a port forwarding rule:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



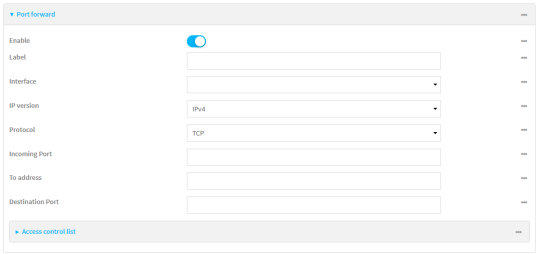
The **Configuration** window is displayed.



3. Click **Firewall** > **Port forwarding**.
4. For **Add port forward**, click ✚.



The port forwarding rule configuration window is displayed.



Port forwarding rules are enabled by default. To disable, click to toggle off **Enable**.

5. (Optional) Type a **Label** that will be used to identify the rule.

6. For **Interface**, select the network interface for the rule.

   Network connections will only be forwarded if their destination address matches the IP address of the selected network interface.
7. For **IP version**, select either **IPv4** or **IPv6**.

   Network connections will only be forwarded if they match the selected IP version.
8. For **Protocol**, select the type of internet protocol.

   Network connections will only be forwarded if they match the selected protocol.
9. For **Incoming port(s)**, type the public-facing port number that network connections must use for their traffic to be forwarded.
10. For **To Address**, type the IP address of the server to which traffic should be forwarded.
11. For **Destination Port(s)**, type the port number, comma-separated list of port numbers, or range of port numbers on the server to which traffic should be forwarded. For example, to forward traffic to ports one, three, and five through ten, enter: **1, 3, 5-10**.
12. (Optional) Click **Access control list** to create a white list of devices that are authorized to leverage this forwarding rule, based on either the IP address or firewall zone:
    - To white list IP addresses:
      a. Click **Addresses**.
      b. For **Add Address**, enter an IP address and click ✚.
      c. Repeat for each additional IP address that should be white listed.
    - To specify firewall zones for white listing:
      a. Click **Zones**.
      b. For **Add zone**, click ✚.
      c. For **Zone**, select the appropriate zone.
      d. Repeat for each additional zone.
13. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type:

```
(config)> add firewall dnat end
(config firewall dnat 0)>
```

Port forwarding rules are enabled by default. To disable the rule:

```
(config firewall dnat 0)> enable false
(config firewall dnat 0)>
```

4. Set the network interface for the rule.

```
(config firewall dnat 0)> interface
(config firewall dnat 0)>
```

Network connections will only be forwarded if their destination address matches the IP address of this network interface.

a. Use the **?** to determine available interfaces:
b. Set the interface. For example:

```
(config firewall dnat 0)> interface eth1
(config firewall dnat 0)>
```

5. Set the IP version. Allowed values are **ipv4** and **ipv6**. The default is **ipv4**.

```
(config firewall dnat 0)> ip_version ipv6
(config firewall dnat 0)>
```

6. Set the public-facing port number that network connections must use for their traffic to be forwarded.

```
(config firewall dnat 0)> port port
(config firewall dnat 0)>
```

7. Set the type of internet protocol .

```
(config firewall dnat 0)> protocol value
(config firewall dnat 0)>
```

Network connections will only be forwarded if they match the selected protocol. Allowed values are **custom**, **tcp**, **tcpudp**, or **upd**. The default is **tcp**.

8. Set the IP address of the server to which traffic should be forwarded:
   - For IPv4 addresses:

```
(config firewall dnat 0)> to_address ip-address
(config firewall dnat 0)>
```

   - For IPv6 addresses:

```
(config firewall dnat 0)> to_address6 ip-address
(config firewall dnat 0)>
```

9. Set the public-facing port number(s) that network connections must use for their traffic to be forwarded.

```
(config firewall dnat 0)> to_port value
(config firewall dnat 0)>
```

where *value* is the port number, comma-separated list of port numbers, or range of port numbers on the server to which traffic should be forwarded. For example, to forward traffic to ports one, three, and five through ten, enter **1, 3, 5-10**.

10. (Optional) To create a white list of devices that are authorized to leverage this forwarding rule, based on either the IP address or firewall zone, change to the acl node:

```
(config firewall dnat 0)> acl
(config firewall dnat 0 acl)>
```

- To white list an IP address:
  - For IPv4 addresses:

    ```
    (config firewall dnat 0 acl> add address end ip-address
    (config firewall dnat 0 acl)>
    ```

  - For IPv6 addresses:

    ```
    (config firewall dnat 0 acl> add address6 end ip-address
    (config firewall dnat 0 acl)>
    ```

  Repeat for each appropriate IP address.
- To specify the firewall zone for white listing:

  ```
  (config firewall dnat 0 acl)> add zone end zone
  ```

  Repeat for each appropriate zone.

  To view a list of available zones:

  ```
  (config firewall dnat 0 acl)> .. .. .. zone ?

  Zones: A list of groups of network interfaces that can be referred
  to by packet filtering rules
  and access control lists.

   Additional Configuration
   ------------------------------------------------------------------
  -----------
    any
    dynamic_routes
    edge
    external
    internal
    ipsec
    loopback
    setup

  (config firewall dnat 0 acl)>
  ```

11. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

12. Type **exit** to exit the Admin CLI.

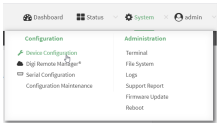
    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Delete a port forwarding rule

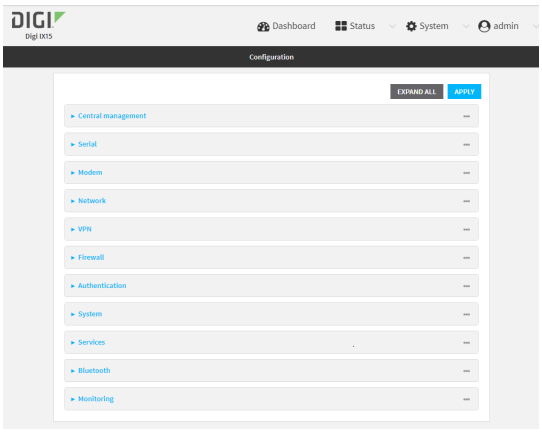
To delete a port forwarding rule:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Firewall** > **Port forwarding**.
4. Click the menu icon (**...**) next to the appropriate port forwarding rule and select **Delete**.



5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Determine the index number of the port forwarding rule you want to delete:

```
(config)> show firewall dnat
0
        acl
                no address
                no zone
        enable true
        interface
        ip_version ipv4
        label IPv4 port forwarding rule
        port 10000
        protocol tcp
        to_address6 10.10.10.10
        to_port 10001

1
        acl
                no address6
                no zone
        enable false
        interface
        ip_version ipv6
        label IPv6 port forwarding rule
        port 10002
        protocol tcp
        to_address6 c097:4533:bd63:bb12:9a6f:5569:4b53:c29a
        to_port 10003
(config)>
```

4. To delete the rule, use the index number with the **del** command. For example:

```
(config)> del firewall dnat 1
```

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Packet filtering

Configure packet filtering

Required configuration items

- The action that the packet filtering rule will perform, either **Accept**, **Reject**, or **Drop**.
- The source firewall zone: Packets originating from interfaces on this zone will be monitored by this rule.
- The destination firewall zone: Packets destined for interfaces on this zone will be accepted, rejected, or dropped by this rule.

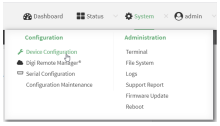
Additional configuration requirements

- A label for the rule.
- The IP version to be matched, either **IPv4**, **IPv6**, or **Any**.
- The protocol to be matched, one of:
  - **TCP**
  - **UDP**
  - **ICMP**
  - **ICMP6**
  - **Any**

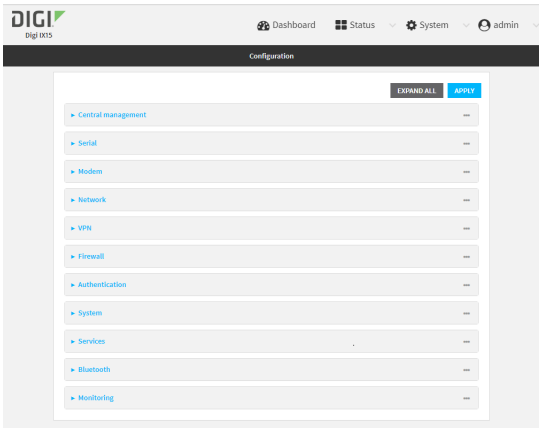
To configure a packet filtering rule:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

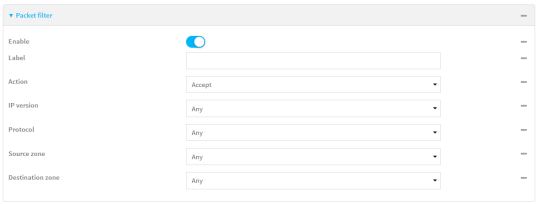


The **Configuration** window is displayed.



3. Click **Firewall** > **Packet filtering**.
   - To create a new packet filtering rule, for **Add packet filter**, click ✚.
   - To edit the default packet filtering rule or another existing packet filtering rule, click to expand the rule.

   The packet filtering rule configuration window is displayed.



   Packet filters are enabled by default. To disable, click to toggle off **Enable**.
4. (Optional) Type a **Label** that will be used to identify the rule.
5. For **Action**, select one of:
   - **Accept**: Allows matching network connections.
   - **Reject**: Blocks matching network connections, and sends an ICMP error if appropriate.
   - **Drop**: Blocks matching network connections, and does not send a reply.
6. Select the **IP version**.
7. Select the **Protocol**.
8. For **Source zone**, select the firewall zone that will be monitored by this rule for incoming connections from network interfaces that are a member of this zone.

   See Firewall configuration for more information about firewall zones.
9. For **Destination zone**, select the firewall zone. Packets destined for network interfaces that are members of this zone will either be accepted, rejected or dropped by this rule.

   See Firewall configuration for more information about firewall zones.
10. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

   To edit the default packet filtering rule or another existing packet filtering rule:

   a. Determine the index number of the appropriate packet filtering rule:

```
(config)> show firewall filter
0
    action accept
    dst_zone any
    enable true
    ip_version any
    label Allow all outgoing traffic
    protocol any
    src_zone internal
1
    action drop
    dst_zone internal
    enable true
    ip_version any
    label myfilter
    protocol any
    src_zone external
(config)>
```

   b. Select the appropriate rule by using its index number:

```
(config)> firewall filter 1
(config firewall filter 1)>
```

   To create a new packet filtering rule:

```
(config)> add firewall filter end
(config firewall filter 1)>
```

   Packet filtering rules are enabled by default. To disable the rule:

```
(config firewall filter 1)> enable false
(config firewall filter 1)>
```

3. (Optional) Set the label for the rule.

```
(config firewall filter 1)> label "My filter rule"
(config firewall filter 1)>
```

4. Set the action to be performed by the filter rule.

```
(config firewall filter 1)> action value
(config firewall filter 1)>
```

where *value* is one of:

- **accept**: Allows matching network connections.
- **reject**: Blocks matching network connections, and sends an ICMP error if appropriate.
- **drop**: Blocks matching network connections, and does not send a reply.

5. Set the firewall zone that will be monitored by this rule for incoming connections from network interfaces that are a member of this zone:

   See Firewall configuration for more information about firewall zones.

```
(config firewall filter 1)> src_zone my_zone
(config firewall filter 1)>
```

6. Set the destination firewall zone. Packets destined for network interfaces that are members of this zone will either be accepted, rejected or dropped by this rule.

   See Firewall configuration for more information about firewall zones.

```
(config firewall filter 1)> dst_zone my_zone
(config firewall filter 1)>
```

7. Set the IP version.

```
(config firewall filter 1)> ip_version value
(config firewall filter 1)>
```

where *value* is one of:

- **any**
- **ipv4**
- **ipv6**
- The default is **any**.

8. Set the protocol.

```
(config firewall filter 1)> protocol value
(config firewall filter 1)>
```

where value is one of:

- **any**
- **icmp**
- **icmpv6**

- **tcp**
- **upd**

The default is **any**.

9. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

10. Type **exit** to exit the Admin CLI.

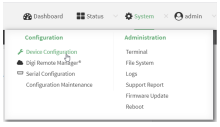
    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Enable or disable a packet filtering rule

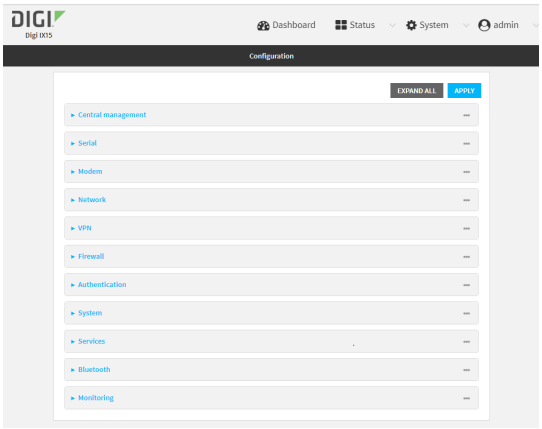
To enable or disable a packet filtering rule:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

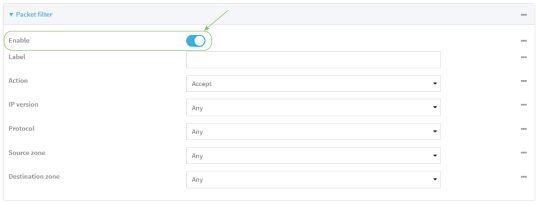


The **Configuration** window is displayed.



3. Click **Firewall** > **Packet filtering**.
4. Click the appropriate packet filtering rule.

5. Click **Enable** to toggle the rule between enabled and disabled.



6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Determine the index number of the appropriate port forwarding rule:

```
(config)> show firewall filter
0
    action accept
    dst_zone any
    enable true
    ip_version any
    label Allow all outgoing traffic
    protocol any
    src_zone internal
1
    action drop
    dst_zone internal
    enable true
    ip_version any
    label My packet filter
    protocol any
    src_zone external
(config)>
```

4. To enable a packet filtering rule, use the index number with the **enable true** command. For example:

```
(config)> firewall filter 1 enable true
```

5. To disable a packet filtering rule, use the index number with the **enable false** command. For example:

```
(config)> firewall filter 1 enable false
```

6. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

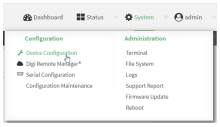
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Delete a packet filtering rule

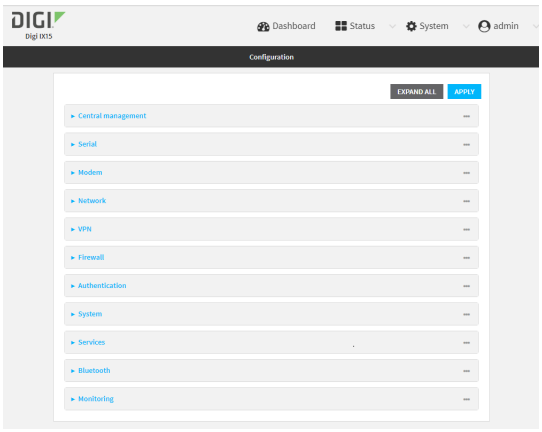
To delete a packet filtering rule:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **Firewall** > **Packet filtering**.
4. Click the menu icon (**...**) next to the appropriate packet filtering rule and select **Delete**.

5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Determine the index number of the packet filtering rule you want to delete:

```
(config)> show firewall filter
0
    action accept
    dst_zone any
    enable true
    ip_version any
    label Allow all outgoing traffic
    protocol any
    src_zone internal
1
    action drop
    dst_zone internal
    enable true
    ip_version any
    label My packet filter
    protocol any
    src_zone external
(config)>
```

4. To delete the rule, use the index number with the **del** command. For example:

```
(config)> del firewall filter 1
```

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

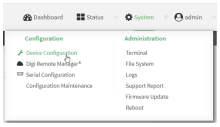
Configure custom firewall rules

Custom firewall rules consist of a script of shell commands that can be used to install firewall rules, ipsets, and other system configuration. These commands are run whenever system configuration changes occur that might cause changes to the firewall.

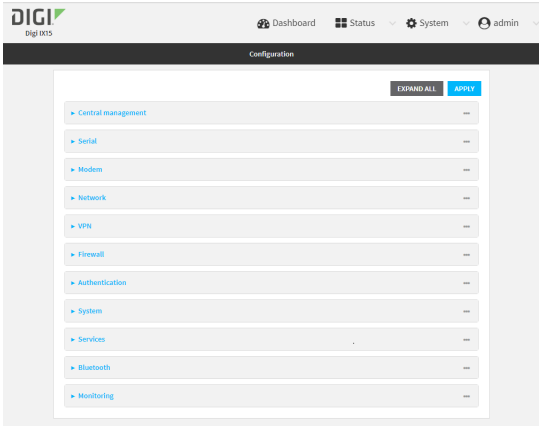
To configure custom firewall rules:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Firewall** > **Custom rules**.



4. **Enable** the custom rules.
5. (Optional) Enable **Override** to override all preconfigured firewall behavior and rely solely on the custom firewall rules.
6. For **Rules**, type the shell command that will execute the custom firewall rules script.

7. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable custom firewall rules:

```
(config)> firewall custom enable true
(config)>
```

4. (Optional) Instruct the device to override all preconfigured firewall behavior and rely solely on the custom firewall rules:

```
(config)> firewall custom override true
(config)>
```

5. Set the shell command that will execute the custom firewall rules script:

```
(config)> firewall custom rules "shell-command"
(config)>
```

6. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure Quality of Service options

Quality of Service (QoS) options allow you to manage the traffic performance of various services, such as Voice over IP (VoIP), cloud computing, traffic shaping, traffic prioritizing, and bandwidth allocation. When configuring QOS, you can only control the queue for outgoing packets on each interface (egress packets), not what is received on the interface (packet ingress).

A QoS *binding* contains the policies and rules that apply to packets exiting the IX15 device on the binding's interface. By default, the IX15 device has two preconfigured QoS bindings, **Outbound** and **Inbound**. These bindings are an example configuration designed for a typical VoIP site:

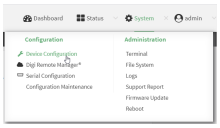
- **Outbound** provides an example of matching packets as they are routed from the device onto the WAN interface.
- **Inbound** provides an example of matching packets as they are routed from the device onto a LAN interface.

These example bindings are disabled by default.

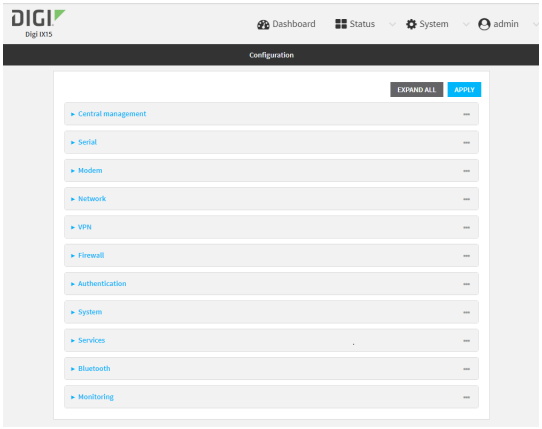
Enable the preconfigured bindings

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Firewall** > **Quality of Service**.
4. Click to expand either **Outbound** or **Inbound**.
5. **Enable** the binding.
6. Select an **Interface**.
7. Examine the remaining default settings and modify as appropriate for your network.
8. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable one of the preconfiged bindings:
   - To enable the Outbound binding:

     ```
     (config)> firewall qos 0 enable true
     (config)>
     ```

   - To enable the Inbound binding:

     ```
     (config)> firewall qos 1 enable true
     (config)>
     ```

4. Set the interface for the binding. Use the index number of the binding; for example, to set the interface for the Outbound binding:
   a. Use the **?** to determine available interfaces:
   b. Set the interface. For example:

      ```
      (config)> firewall qos 0 interface /network/interface/eth1
      (config)>
      ```

5. Examine the remaining default settings and modify as appropriate for your network.
6. Save the configuration and apply the change:

   ```
   (config)> save
   Configuration saved.
   >
   ```

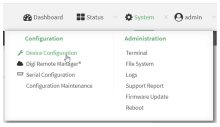
7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

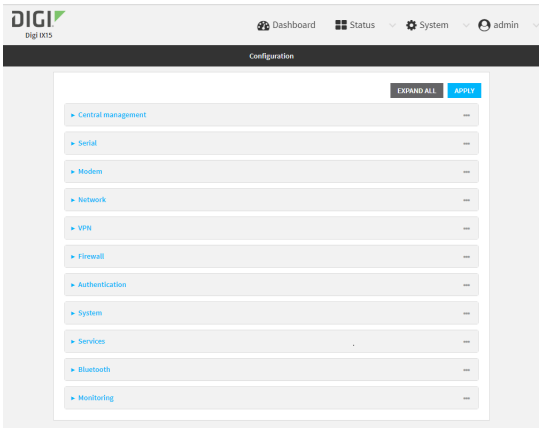
Create a new binding

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



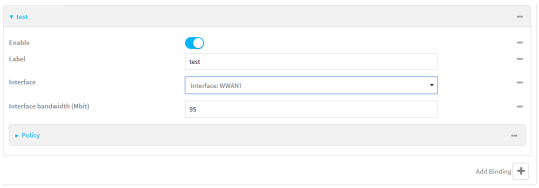
The **Configuration** window is displayed.



3. Click **Firewall** > **Quality of Service**.
4. For **Add Binding**, click ✚.



The quality of service binding configuration window is displayed.

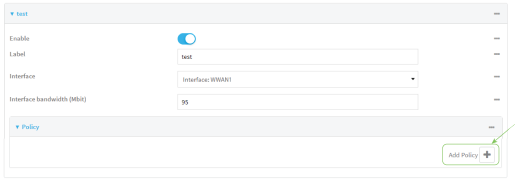


5. **Enable** the binding.
6. (Optional) Type a **Label** for the binding.
7. Select an **Interface** to queue egress packets on. The binding will only match traffic that is being sent out on this interface.
8. (Optional) For **Interface bandwidth (Mbit)**, set the maximum egress bandwidth of the interface, in megabits, allocated to this binding. Typically, this should be 95% of the available bandwidth. Allowed value is any integer between **1** and **1000**.
9. Create a policy for the binding:

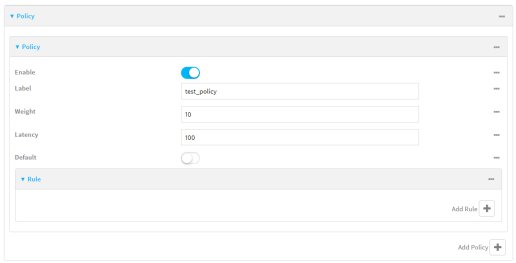
   At least one policy is required for each binding. Each policy can contain up to 30 rules.

a. Click to expand **Policy**.

b. For **Add Policy**, click ✚.



The QoS binding policy configuration window is displayed.



New QoS binding policies are enabled by default. To disable, click **Enable**.

c. (Optional) Type a **Label** for the binding policy.

d. For **Weight**, type a value for the amount of available bandwidth allocated to the policy, relative to other policies for this binding.

The larger the weight, with respect to the other policy weights, the larger portion of the maximum bandwidth is available for this policy. For example, if a binding contains three policies, and each policy contains a weight of 10, each policy will be allocated one third of the total interface bandwidth.

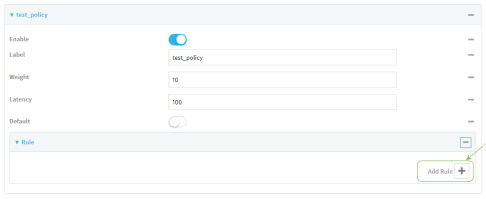
e. For **Latency**, type the maximum delay before the transmission of packets. A lower latency means that the packets will be scheduled more quickly for transmission.

f. Select **Default** to identify this policy as a fall-back policy. The fall-back policy will be used for traffic that is not matched by any other policy. If there is no default policy associated with this binding, packets that do not match any policy rules will be dropped.

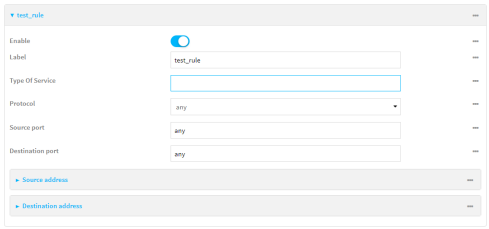
g. If **Default** is disabled, you must configure at least one rule:

i. Click to expand **Rule**.

ii. For **Add Rule**, click ✚.



The QoS binding policy rule configuration window is displayed.



New QoS binding policy rules are enabled by default. To disable, click **Enable**.

iii. (Optional) Type a **Label** for the binding policy rule.

iv. For **Type Of Service**, type the value of the Type of Service (ToS) packet header that defines packet priority. If unspecified, this field is ignored.

See https://www.tucny.com/Home/dscp-tos for a list of common TOS values.

v. For **Protocol**, select the IP protocol matching criteria for this rule.

vi. For **Source port**, type the port, or **any**, as a source traffic matching criteria.

vii. For **Destination port**, type the port, or **any**, as a destination traffic matching criteria.

viii. Click to expand **Source address** and select the **Type**:

- **Any**: Source traffic from any address will be matched.
- **Interface**: Only traffic from the selected **Interface** will be matched.
- **IPv4 address**: Only traffic from the IP address typed in **IPv4 address** will be matched. Use the format ***IPv4_address***[**/*netmask***], or use **any** to match any IPv4 address.
- **IPv6 address**: Only traffic from the IP address typed in **IPv6 address** will be matched. Use the format ***IPv6_address***[**/*prefix_length***], or use **any** to match any IPv6 address.
- **MAC address**: Only traffic from the MAC address typed in **MAC address** will be matched.

ix. Click to expand **Destination address** and select the **Type**:

- **Any**: Traffic destined for anywhere will be matched.
- **Interface**: Only traffic destined for the selected **Interface** will be matched.
- **IPv4 address**: Only traffic destined for the IP address typed in **IPv4 address** will be matched. Use the format ***IPv4_address***[**/*netmask***], or use **any** to match any IPv4 address.
- **IPv6 address**: Only traffic destined for the IP address typed in **IPv6 address** will be matched. Use the format ***IPv6_address***[**/*prefix_length***], or use **any** to match any IPv6 address.

Repeat to add a new rule. Up to 30 rules can be configured.

10. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a binding:

```
(config)> add firewall qos end
(config firewall qos 2)>
```

   New binding are enabled by default. To disable:

```
(config firewall qos 2)> enable false
(config firewall qos 2)>
```

4. (Optional) Set a label for the new binding:

```
(config firewall qos 2)> label my_binding
(config firewall qos 2)>
```

5. Set the interface to queue egress packets on. The binding will only match traffic that is being sent out on this interface:
   a. Use the **?** to determine available interfaces:
   b. Set the interface. For example:

```
(config firewall qos 2)> interface /network/interface/eth1
(config firewall qos 2)>
```

6. (Optional) Set the maximum egress bandwidth of the interface, in megabits, allocated to this binding.

```
(config firewall qos 2)> bandwidth int
(config firewall qos 2)>
```

   where *int* is an integer between **1** and **1000**. Typically, this should be 95% of the available bandwidth. The default is **95**.
7. Create a policy for the binding:

   At least one policy is required for each binding. Each policy can contain up to 30 rules.
   a. Change to the policy node of the configuration:

```
(config firewall qos 2)> policy
(config firewall qos 2 policy)>
```

b. Add a policy:

```
(config firewall qos 2 policy)> add end
(config firewall qos 2 policy 0)>
```

New QoS binding policies are enabled by default. To disable:

```
(config firewall qos 2 policy 0)> enable false
(config firewall qos 2 policy 0)>
```

c. (Optional) Set a label for the new binding policy:

```
(config firewall qos 2 policy 0)> label my_binding_policy
(config firewall qos 2 policy 0)>
```

d. Set a value for the amount of available bandwidth allocated to the policy, relative to other policies for this binding.

The larger the weight, with respect to the other policy weights, the larger portion of the maximum bandwidth is available for this policy. For example, if a binding contains three policies, and each policy contains a weight of 10, each policy will be allocated one third of the total interface bandwidth.

```
(config firewall qos 2 policy 0)> weight int
(config firewall qos 2 policy 0)>
```

where *int* is any integer between **1** and **65535**. The default is **10**.

e. Set the maximum delay before the transmission of packets. A lower number means that the packets will be scheduled more quickly for transmission.

```
(config firewall qos 2 policy 0)> latency int
(config firewall qos 2 policy 0)>
```

where *int* is any integer, **1** or greater. The default is **100**.

f. To identify this policy as a fall-back policy:

```
(config firewall qos 2 policy 0)> default true
(config firewall qos 2 policy 0)>
```

The fall-back policy will be used for traffic that is not matched by any other policy. If there is no default policy associated with this binding, packets that do not match any policy rules will be dropped. If the policy is not a fall-back policy, you must configure at least one rule:

i. Change to the rule node of the configuration:

```
(config firewall qos 2 policy 0)> rule
(config firewall qos 2 policy 0 rule)>
```

ii. Add a rule:

```
(config firewall qos 2 policy 0 rule)> add end
(config firewall qos 2 policy 0 rule 0)>
```

New QoS binding policy rules are enabled by default. To disable:

```
(config firewall qos 2 policy 0 rule 0)> enable false
(config firewall qos 2 policy 0 rule 0)>
```

iii. (Optional) Set a label for the new binding policy rule:

```
(config firewall qos 2 policy 0 rule 0)> label my_binding_policy_
rule
(config firewall qos 2 policy 0 rule 0)>
```

iv. Set the value of the Type of Service (ToS) packet header that defines packet priority. If unspecified, this field is ignored.

```
(config firewall qos 2 policy 0 rule 0)> tos value
(config firewall qos 2 policy 0 rule 0)>
```

where value is a hexadecimal number. See https://www.tucny.com/Home/dscp-tos for a list of common TOS values.

v. Set the IP protocol matching criteria for this rule:

```
(config firewall qos 2 policy 0 rule 0)> protocol value
(config firewall qos 2 policy 0 rule 0)>
```

where *value* is one of **tcp**, **udp**, or **any**.

vi. Set the source port to define a source traffic matching criteria:

```
(config firewall qos 2 policy 0 rule 0)> srcport value
(config firewall qos 2 policy 0 rule 0)>
```

where *value* is the IP port number, a range of port numbers using the format *IP_port-IP_port*, or **any**.

vii. Set the destination port to define a destination matching criteria:

```
(config firewall qos 2 policy 0 rule 0)> dstport value
(config firewall qos 2 policy 0 rule 0)>
```

where *value* is the IP port number, a range of port numbers using the format *IP_port-IP_port*, or **any**.

viii. Set the source address type:

```
(config network qos 2 policy 0 rule 0)> src type value
(config network qos 2 policy 0 rule 0)>
```

where *value* is one of:

- **any**: Source traffic from any address will be matched.

  See Firewall configuration for more information about firewall zones.
- **interface**: Only traffic from the selected interface will be matched. Set the interface:

i. Use the **?** to determine available interfaces:

ii. Set the interface. For example:

```
(config network qos 2 policy 0 rule 0)> src interface
/network/interface/eth1
(config network qos 2 policy 0 rule 0)>
```

- **address**: Only traffic from the IP address typed in **IPv4 address** will be matched. Set the address that will be matched:

```
(config network qos 2 policy 0 rule 0)> src address value
(config network qos 2 policy 0 rule 0)>
```

where value uses the format ***IPv4_address***[**/*netmask***], or **any** to match any IPv4 address.

- **address6**: Only traffic from the IP address typed in **IPv6 address** will be matched. Set the address that will be matched:

```
(config network qos 2 policy 0 rule 0)> src address6 value
(config network qos 2 policy 0 rule 0)>
```

where value uses the format ***IPv6_address***[**/*prefix_length***], or **any** to match any IPv6 address.

- **mac**: Only traffic from the MAC address typed in **MAC address** will be matched. Set the MAC address to be matched:

```
(config network qos 2 policy 0 rule 0)> src mac MAC_address
(config network qos 2 policy 0 rule 0)>
```

ix. Set the destination address type:

```
(config network qos 2 policy 0 rule 0)> dst type value
(config network qos 2 policy 0 rule 0)>
```

where *value* is one of:

- **any**: Traffic destined for anywhere will be matched.

  See Firewall configuration for more information about firewall zones.

- **interface**: Only traffic destined for the selected **Interface** will be matched. Set the interface:

  i. Use the **?** to determine available interfaces:

  ii. Set the interface. For example:

```
(config network qos 2 policy 0 rule 0)> dst interface
/network/interface/eth1
(config network qos 2 policy 0 rule 0)>
```

- **address**: Only traffic destined for the IP address typed in **IPv4 address** will be matched. Set the address that will be matched:

```
(config network qos 2 policy 0 rule 0)> src address value
(config network qos 2 policy 0 rule 0)>
```

where value uses the format ***IPv4_address***[**/*netmask***], or **any** to match any IPv4 address.

- **address6**: Only traffic destined for the IP address typed in **IPv6 address** will be matched. Set the address that will be matched:

```
(config network qos 2 policy 0 rule 0)> src address6 value
(config network qos 2 policy 0 rule 0)>
```

where value uses the format ***IPv6_address***[**/*prefix_length***], or **any** to match any IPv6 address.

Repeat to add a new rule. Up to 30 rules can be configured.

8. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Containers

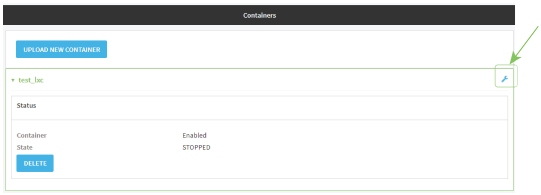
The IX15 device includes support for LXC Linux containers. LXC containers are a lightweight, operating system level method of virtualization that allows you to run one or more isolated Linux instances on a the same host using the host's Linux kernal.

Upload a new LXC container

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. From the main menu, click **Status**. Under **Services**, click **Containers**.
3. Click **Upload New Container**.
4. From your local file system, select the container file in *.tgz format.

   You can download a simple example container file, test_lxc.tgz, from the Digi website.
5. **Create Configuration** is selected by default. This will create a configuration on the device for the container when it is installed. If deselected, you will need to create the configuration manually.
6. Click **Apply**.
7. If **Create Configuration** was selected when the container was created, click to go to the container configuration.



See Configure a container for further information about configuring the container.

Configure a container

Required configuration items

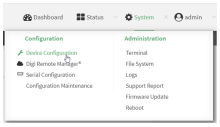
- The following configuration options are completed automatically if **Create Configuration** was selected when the container was created. See Upload a new LXC container for details:
  - Name of the container.
  - Enable the container.
  - Whether or not the container should use the device's system libraries.
- Determine whether or not the device should including virtual networking capabilities.

Additional configuration items

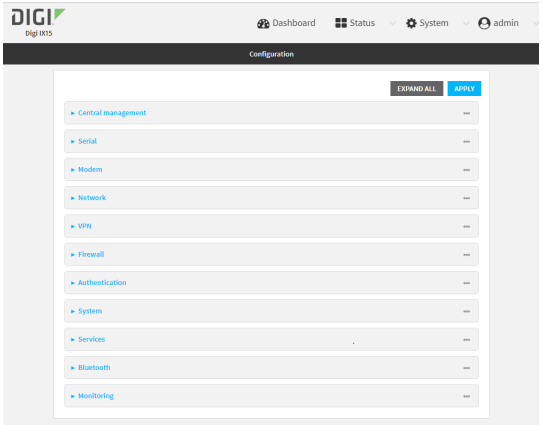
- If virtual networking is enabled:
  - The bridge to be used to provide network connectivity.
  - A static IP address for the container.
  - The network gateway.
- Serial ports on the device that the container will have access to.

WebUI

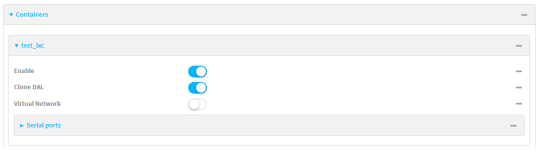
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **System** > **Containers**.
4. For **Add Container**, type the name of the container and click ✚.
   The Container configuration window is displayed.



New containers are enabled by default. To disable, or to enable a container if it has been disabled, click **Enable**.

5. **Clone DAL** is enabled by default. This allows the container to use the device's system libraries.
6. Enable **Virtual Network** if the container should have network access:
   a. Select a **Network Bridge Device** that will provide access to the container.
   b. (Optional) Enter a static **IP Address** and netmask for the container. This must be a valid IP address for the bridge, or, if left blank, a DHCP server can assign the container an IP address.
   c. (Optional) For **Gateway**, type the IP address of the network gateway.
7. Click to expand **Serial ports** to sssign serial ports that the container will have access to.
   a. For **Add Port**, click ✚.
   b. For **Port**, select the serial port.
8. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Create a new container:

```
(config)> add system container name
(config system container name)>
```

   where *name* is the

   New access points are enabled by default.
4. New containers are enabled by default. To disable:

```
(config system container name)> enable false
(config system container name)>
```

5. By default, the container will use the device's system libraries. To disable:

```
(config system container name)> dal false
(config system container name)>
```

6. If the device will use virtual networking:
   a. Enable virtual networking:

```
(config system container name)> network true
(config system container name)>
```

   b. Set the network bridge device that will be used to provide network access:
      i. Use the **?** to determine the available bridges:

```
(config system container name)> bridge ?

Network Bridge Device: Containers require a bridge to access the
network. Choose
which bridge to connect the container to.
Format:
  lan1
Current value:

(config system container name)>
```

      ii. Set the bridge:

```
(config system container name)> bridge lan1
(config system container name)>
```

   c. (Optional) Set the IP address and netmask for the container:

```
(config system container name)> address IP_address/netmask
(config system container name)>
```

   d. (Optional) Set the IP address of the network gateway:

```
(config system container name)> gateway IP_address
(config system container name)>
```

7. (Optional) Assign serial ports that the container will have access to:
   a. Determine available serial ports:

```
(config system container name)> ... serial

Serial

 Additional Configuration
 ------------------------------------------------------------------------
----------
 port1                      Port 1
 ...
```

```
(config system container name)>
```

b. Add the port:

```
(config system container name)> add ports end port1
(config system container name)>
```

8. Save the configuration and apply the change:

```
(config network wireless client new_client)> save
Configuration saved.
>
```

9. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Starting and stopping the container

Container commands are not available from the Admin CLI. You must access the device shell in order to run Python applications from the command line. See Authentication groups for information about configuring authentication groups that include shell access.

Starting the container

There are two methods to start containers:

- Non-persistent: Changes made to the container file system will be lost when the container is stopped.
- Persistent: Changes made to the container file system when not be lost when the container is stopped.

Starting a container in non-persistent mode

To start the container in non-persistent mode:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

2. At the shell prompt, type:

```
# lxc container_name
lxc #
```

   where *container_name* is the name of the container as configured on the device. For example:

```
# lxc test_lxc
lxc #
```

This will start the container by using **/bin/sh -l,** which runs the shell and loads the shell profile. The default shell profile includes an **lxc #** prompt.

Starting a container in persistent mode

To start the container in persistent mode, include the **-p** option at the command line. For example:

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the shell prompt, type:

```
# lxc test_lxc -p
lxc #
```

This will start the container by using **/bin/sh -l,** which runs the shell and loads the shell profile. The default shell profile includes an **lxc #** prompt.

Starting a container by including an executable

You can supply an executable to run when you start the container, along with any parameters. If you don't supply a parameter, the default behavior is to run the executable by using **/bin/sh -l,** which runs the shell and loads the shell profile. This is useful when you use the **Clone DAL** option when uploading the container, which includes the devices's system libraries. In this case, the command without any additional parameters will use the device's shell. See Upload a new LXC container for more information.

For example, to start a container and run a python script called my_python_script.py in the default shell, type:

```
# lxc test_lxc /usr/bin/python3 /usr/bin/my_python_sctipt.py
```

This will run the script from /usr/bin inside the container. If you have /usr/bin/my_python_script.py on your device's native system, it will be ignored.

Stopping the container

1. Log into the IX15 command line as a user with shell access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.
2. At the lxc shell prompt, type:

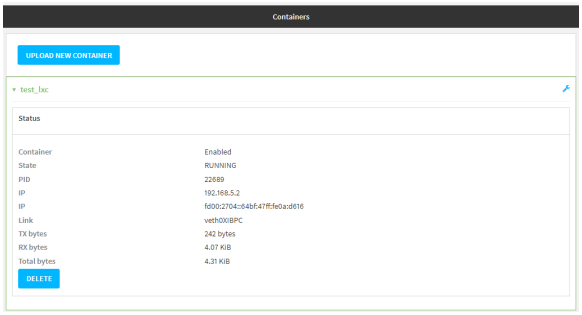
```
lxc # exit
#
```

View the status of containers

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. From the main menu, click **Status**. Under **Services**, click **Containers**.

The Containers status page is displayed.



Command line

Show status of all containers

Use the show containers command with no additional arguments to show the status of all containers on the system:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the prompt, type:

```
> show containers

 Container  Configured  Enabled  State
 ---------  ----------  -------  -----------------
 mytest1    True        enabled  STOPPED
 test_lxc   True        enabled  RUNNING PID 19327
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Show status of a specific container

Use the show containers container *name* command to show the status of the specified container:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the prompt, type:

```
> show containers container test_lxc

 Container  Configured  Enabled  State
 ---------  ----------  -------  -----------------
 test_lxc   True        enabled  RUNNING PID 19327
```

>

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

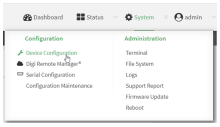
Schedule a script to run in the container

This simple example will:

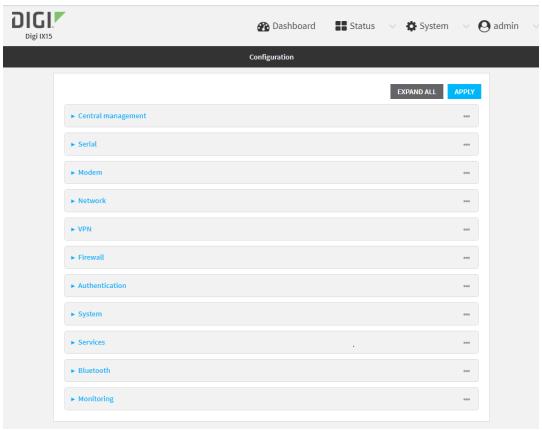
1. Start the container in non-persistent mode.
2. Execute a ping command every ten seconds from inside the container.

WebUI

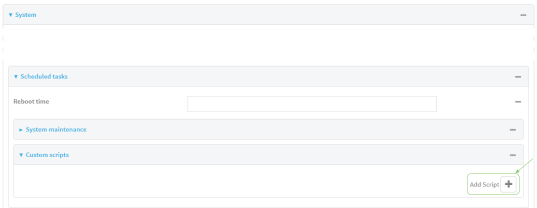
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



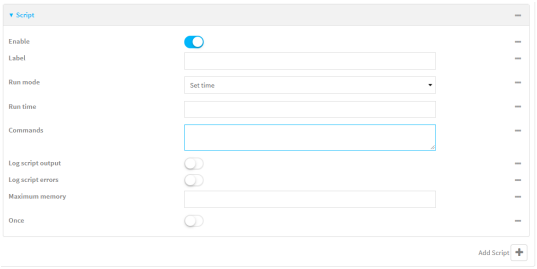
The **Configuration** window is displayed.



3. Click **System** > **Scheduled tasks** > **Custom scripts**.
4. For **Add Script**, click ✚.



The script configuration window is displayed.



5. (Optional) For **Label**, type **container_script**.
6. For **Run mode**, select **Interval**.
7. For **Interval**, type **10s**.
8. For **Commands**, type the following:

```
lxc container_name /bin/ping -c 1 IP_address
```

   For example:

```
lxc test_lxc /bin/ping -c 1 192.168.1.146
```

9. Click to disable **Sandbox**. Sandbox restrictions are not necessary when a container is used.
10. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Add a script:

```
(config)> add system schedule script end
(config system schedule script 0)>
```

4. Provide a label for the script, for example:

```
(config system schedule script 0)> label test_lxc
(config system schedule script 0)>
```

5. Set the mode to interval:

```
(config system schedule script 0)> when interval
(config system schedule script 0)>
```

6. Set the interval to ten seconds:

```
(config system schedule script 0)> on_interval 10s
(config system schedule script 0)>
```

7. Set the commands that will execute the script:

```
(config system schedule script 0)> commands "lxc script_name /bin/ping -c
1 IP_address"
(config system schedule script 0)>
```

   For example:

```
(config system schedule script 0)> commands "lxc test_lxc /bin/ping -c 1
192.168.1.146"
(config system schedule script 0)>
```

8. Disable the **sandbox**. Sandbox restrictions are not necessary when a container is used.

```
(config system schedule script 0)> sandbox false
(config system schedule script 0)>
```

9. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

10. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Create a custom container

This example creates a simple custom container that contains a python script in the /etc directory.

In this example, we will use a simple container file named test_lxc.tgz. You can download test_lxc.tgz from the Digi website.

At the command line of a Linux host, we will unpack the file, add a simple python script, and create a new container file that includes the python script.

Create the custom container file

1. At the command line of a Linux host, unpack the test_lxc.tgz file:

```
$ tar -xfv test_lxc.tgz
rootfs/
```

```
rootfs/usr/
rootfs/etc/
rootfs/etc/group
rootfs/etc/profile
rootfs/etc/passwd
rootfs/tmp/
$
```

2. Change to the rootfs/etc directory:

```
$ cd rootfs/etc
$
```

3. Create a file named test.py with the following contents:

```
print("Hello world.\n")
```

4. Change directories to leave the container file structure:

```
$ cd ../..
```

5. Change user and group permissions on all files in the container file structure:

```
$ sudo chown -R 165536 rootfs
$ sudo chgrp -R 165536 rootfs
```

6. Tar and zip the directory structure to create a new container file:

```
$ tar -cvf sudo tar -cvf python_lxc.tgz rootfs
```

Test the custom container file

1. Add the new container to your IX15 device:
    i. Log into the IX15 WebUI as a user with Admin access.
    ii. From the main menu, click **Status**. Under **Services**, click **Containers**.
    iii. Click **Upload New Container**.
    iv. From your local file system, select the container file.
    You can download a simple example container file, test_lxc.tgz, from the Digi website.
    v. **Create Configuration** is selected by default. This will create a configuration on the device for the container when it is installed. If deselected, you will need to create the configuration manually.
    vi. Click **Apply**.
2. Log into the IX15 command line as a user with shell access.
    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **shell** to access the device shell.

3. At the shell prompt, type:

```
# lxc python_lxc
lxc #
```

4. Execute the python command:

```
lxc # python /etc/test.py
Hello world.
lxc #
```

System administration

This chapter contains the following topics:

Review device status 735
Configure system information 736
Update system firmware 738
Update cellular module firmware 744
Reboot your IX15 device 747
Erase device configuration and reset to factory defaults 750
Locate the device by using the Find Me feature 755
Configuration files 757
Schedule system maintenance tasks 762
Disable device encryption 767
Configure the speed of your Ethernet port 769

Review device status

You can review the system of your device from either the **Status** page of the Web interface, or from the command line:

WebUI

To display system information:

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **Status**.

   A secondary menu appears, along with a status panel.
3. On the secondary menu, click to display the details panel for the status you want to view.

Command line

To display system information, use the show system command.

- Show basic system information:
  1. Log into the IX15 command line as a user with Admin access.

     Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
  2. Enter **show system** at the prompt:

```
> show system

  Model                    : Digi IX15
  Serial Number            : IX15-000065
  SKU                      : IX15
  Hostname                 : IX15
  MAC Address              : DF:DD:E2:AE:21:18

  Hardware Version         : 50001947-01 1P
  Firmware Version         : 22.2.9.85
  Alt. Firmware Version    : 22.2.9.85
  Alt. Firmware Build Date : Thurs, 03 March 2022 10:16:23
  Bootloader Version       : 19.7.23.0-15f936e0ed

  Current Time             : Thurs, 03 March 2022 10:16:23 +0000
  CPU                      : 1.4%
  Uptime                   : 6 days, 6 hours, 21 minutes, 57 seconds
(541317s)
  Temperature              : 40C

>
```

- Show more detailed system information:
  1. Log into the IX15 command line as a user with Admin access.

     Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
  2. Enter **show system verbose** at the prompt:

```
> show system verbose

  Model                    : Digi IX15
  Serial Number            : IX15-000065
  SKU                      : IX15
  Hostname                 : IX15
  MAC Address              : DF:DD:E2:AE:21:18

  Hardware Version         : 50001947-01 1P
  Firmware Version         : 22.2.9.85
  Alt. Firmware Version    : 22.2.9.85
  Alt. Firmware Build Date : Thurs, 03 March 2022 10:16:23
  Bootloader Version       : 19.7.23.0-15f936e0ed
  Schema Version           : 715

  Timezone                 : UTC
  Current Time             : Thurs, 03 March 2022 10:16:23 +0000
  CPU                      : 1.4%
  Uptime                   : 6 days, 6 hours, 21 minutes, 57 seconds
(541317s)
  Load Average             : 0.01, 0.03, 0.02
  RAM Usage                : 119.554MB/1878.984MB(6%)
  Temperature              : 40C
  Disk
  ----
  Load Average             : 0.09, 0.10, 0.08
  RAM Usage                : 127.843MB/1880.421MB(6%)
  Disk /etc/config Usage   : 18.421MB/4546.371MB(0%)
  Disk /opt Usage          : -4523.-46MB/549.304MB(-822%)
  Disk /overlay Usage      : MB/MB(%)
  Disk /tmp Usage          : 0.007MB/256.0MB(0%)
  Disk /var Usage          : 1.765MB/256.0MB(1%)

>
```

Configure system information

You can configure information related to your IX15 device, such as providing a name and location for the device.

Configuration items

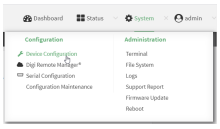
- A name for the device.
- The name of a contact for the device.

- The location of the device.
- A description of the device.
- A banner that will be displayed when users access terminal services on the device.

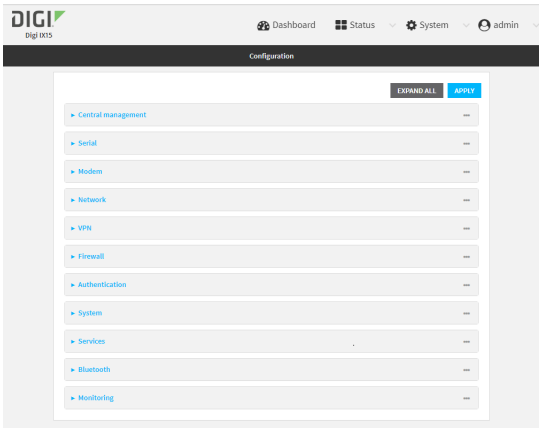
To enter system information:

WebUI

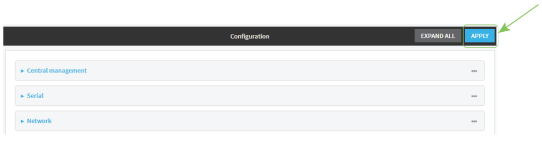
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **System**.
4. For **Name**, type a name for the device. This name will appear in log messages and at the command prompt.
5. For **Contact**, type the name of a contact for the device.
6. For **Location**, type the location of the device.
7. For **Banner**, type a banner message that will be displayed when users log into terminal services on the device.
8. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Set a name for the device. This name will appear in log messages and at the command prompt.

```
(config)> system name 192.168.3.1
192.168.3.1(config)>
```

4. Set the contact for the device:

```
192.168.3.1(config)> system contact "Jane User"
192.168.3.1(config)>
```

5. Set the location for the device:

```
192.168.3.1(config)> system location "9350 Excelsior Blvd., Suite 700,
Hopkins, MN"
192.168.3.1(config)>
```

6. Set the banner for the device. This is displayed when users access terminal services on the device.

```
192.168.3.1(config)> system banner "Welcome to the Digi IX15."
192.168.3.1(config)>
```

7. Save the configuration and apply the change:

```
192.168.3.1(config)> save
Configuration saved.
192.168.3.1>
```

8. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Update system firmware

The IX15 operating system firmware images consist of a single file with the following naming convention:

***platform-version*.bin**

For example, **IX15-22.2.9.85.bin**.

Manage firmware updates using Digi Remote Manager

If you have a network of many devices, you can use Digi Remote Manager **Profiles** to manage firmware updates. Profiles ensure all your devices are running the correct firmware version and that

all newly installed devices are updated to that same version. For more information, see the **Profiles** section of the *Digi Remote Manager User Guide*.

Certificate management for firmware images

The system firmware files are signed to ensure that only Digi-approved firmware load onto the device. The IX15 device validates the system firmware image as part of the update process and only successfully updates if the system firmware image can be authenticated.

Downgrading

Downgrading to an earlier release of the firmware may result in the device configuration being erased.

Downgrading from firmware version 22.2.9.x

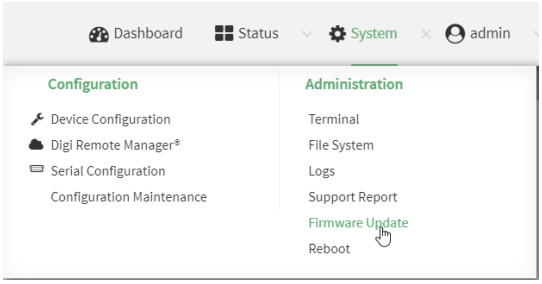
Beginning with firmware version 22.2.9.x, the IX15 device uses certificate-based communication for enhanced security when connecting to Digi Remote Manager. If you downgrade your firmware from version 22.2.9.x to version 21.11.x or previous, your device will no longer be able to communicate with Remote Manager.

To remedy this issue, select the device in Remote Manager and select **Actions** > **Reset Device Certificate**.

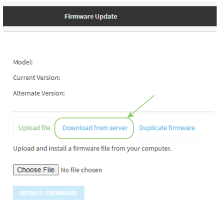
Update firmware over the air (OTA) from the Digi firmware server

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **System**. Under **Administration**, click **Firmware Update**.



3. Click **Download from server**.



4. For **Version:**, select the appropriate version of the device firmware.
5. Click **Update Firmware**.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. >Use the **system firmware ota check** command to determine if new modem firmware is available on the Digi firmware repository.

```
> system firmware ota check
Current firmware version is 21.11.60.63
Checking for latest IX15 firmware...
Newest firmware version available to download is '22.2.9.85'
Device firmware update from '21.11.60.63' to '22.2.9.85' is needed
>
```

3. Use the **modem firmware ota list** command to list available firmware on the Digi firmware repository.

```
> system firmware ota list
21.11.60.63
22.2.9.85
>
```

4. Perform an OTA firmware update:
   - To perform an OTA firmware update by using the most recent available firmware from the Digi firmware repository:
     a. Update the firmware:

```
> system firmware ota update
Downloading firmware version '22.2.9.85'...
Downloaded firmware /tmp/cli_firmware.bin remaining
Applying firmware version '22.2.9.85'...
41388K
netflash: got "/tmp/cli_firmware.bin", length=42381373
netflash: authentication successful
netflash: vendor and product names are verified.
netflash: programming FLASH device /dev/flash/image1
41408K 100%
Firmware update completed, reboot device
>
```

     b. Reboot the device:

```
> reboot
>
```

   - To perform an OTA firmware update by using a specific version from the Digi firmware repository, use the **version** parameter to identify the appropriate firmware version as determined by using **system firmware ota list** command. For example:
     a. Update the firmware:

```
> system firmware ota update version 22.2.9.85
Downloading firmware version '22.2.9.85'...
```

```
Downloaded firmware /tmp/cli_firmware.bin remaining
Applying firmware version '22.2.9.85'...
41388K
netflash: got "/tmp/cli_firmware.bin", length=42381373
netflash: authentication successful
netflash: vendor and product names are verified.
netflash: programming FLASH device /dev/flash/image1
41408K 100%
Firmware update completed, reboot device
>
```

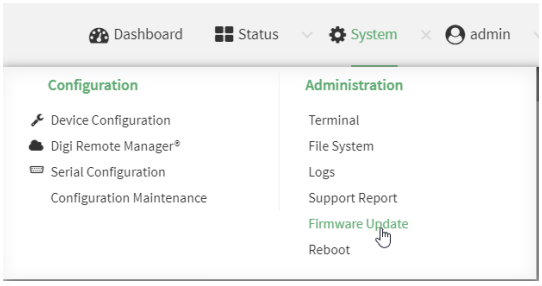
b. Reboot the device:

```
> reboot
>
```

Update firmware from a local file

WebUI

1. Download the IX15 operating system firmware from the Digi Support FTP site to your local machine.
2. Log into the IX15 WebUI as a user with Admin access.
3. On the main menu, click **System**. Under **Administration**, click **Firmware Update**.



4. Click **Choose file**.
5. Browse to the location of the firmware on your local file system and select the file.
6. Click **Update Firmware**.

Command line

1. Download the IX15 operating system firmware from the Digi Support FTP site to your local machine.
2. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
3. Load the firmware image onto the device:

```
> scp host hostname-or-ip user username remote remote-path local local-
path to local
```

where:

- *hostname-or-ip* is the hostname or ip address of the remote host.
- *username* is the name of the user on the remote host.
- *remote-path* is the path and filename of the file on the remote host that will be copied to the IX15 device.
- *local-path* is the location on the IX15 device where the copied file will be placed.

For example:

```
> scp host 192.168.4.1 user admin remote /home/admin/bin/IX15-
22.2.9.85.bin local /etc/config/ to local
admin@192.168.4.1's password: adminpwd
IX15-22.2.9.85.bin                      100%    36MB    11.1MB/s        00:03
>
```

4. Verify that the firmware file has been successfully uploaded to the device:

```
> ls /etc/config/scripts
-rw-r--r--    1 root     root      37511229 May 16 20:10 IX15-
22.2.9.85.bin
-rw-r--r--    1 root     root          2580 May 16 16:44 accns.json
...
>
```

5. Update the firmware by entering the system firmware update command, specifying the firmware file name:

```
> system firmware update file IX15-22.2.9.85.bin
36632K
netflash: got "/etc/config/IX15-22.2.9.85.bin", length=37511229
netflash: authentication successful
netflash: programming FLASH device /dev/flash/image
36633K 100%
Firmware update completed, reboot device
>
```

6. Reboot the device to run the new firmware image using the reboot command.

```
> reboot
Rebooting system
>
```

7. Once the device has rebooted, log into the IX15's command line as a user with Admin access and verify the running firmware version by entering the show system command.

```
> show system

 Hostname                : IX15
 FW Version              : 22.2.9.85
 MAC                     : 0040FF800120
```

```
Model                    : Digi IX15
Current Time             : Thurs, 03 March 2022 10:16:23 +0000
Uptime                   : 42 seconds (42s)

>
```

Dual boot behavior

By default, the IX15 device stores two copies of firmware in two flash memory banks:

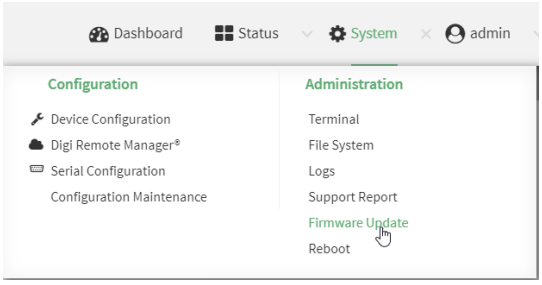
- The current firmware version that is used to boot the device.
- A copy of the firmware that was in use prior to your most recent firmware update.

When the device reboots, it will attempt to use the current firmware version. If the current firmware version fails to load after three consecutive attempts, it is marked as invalid and the device will use the previous firmware version stored in the alternate memory bank.

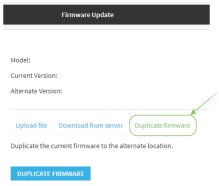
If the device consistently looses power during the boot process, this may result in the current firmware being marked as invalid and the device downgrading to a previous version of the firmware. As a result of this behavior, you can use the following procedure to guarantee that the same firmware is stored in both memory banks:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **System**. Under **Administration**, click **Firmware Update**.



3. Click **Duplicate firmware**.



4. Click **Duplicate Firmware**.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Duplicate the firmware:

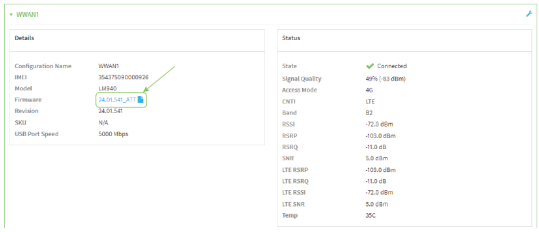
```
> system duplicate-firmware
>
```

Update cellular module firmware

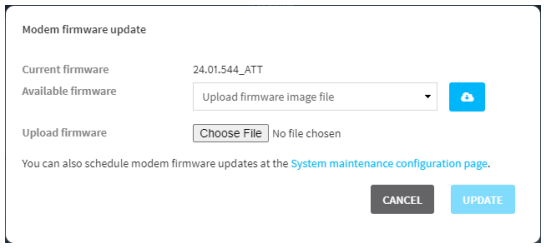
You can update modem firmware by downloading firmware from the Digi firmware repository, or by uploading firmware from your local storage onto the device. You can also schedule modem firmware updates. See Schedule system maintenance tasks for details.

WebUI

1. (Optional) Download the appropriate modem firmware from the Digi repository to your local machine.
2. Log into the IX15 WebUI as a user with Admin access.
3. From the main menu, click **Status** > **Modems**.
4. Click the modem firmware version.



The **Modem firmware update** window opens.



5. To update using firmware from the Digi firmware repository:
   a. Click to view available versions.
   b. For Available firmware, select the firmware.
6. To update using firmware from your local file system:
   a. Click **Choose File**.
   b. Select the firmware.
7. To schedule firmware updates, click **System maintenance configuration page**. See Schedule system maintenance tasks for details.

8. Click **Update**.

Command line

Update modem firmware over the air (OTA)

You can update your modem firmware by querying the Digi firmware repository to determine if there is new firmware available for your modem and performing an OTA modem firmware update:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Use the **modem firmware ota check** command to determine if new modem firmware is available on the Digi firmware repository.

```
> modem firmware ota check

Checking for latest ATT firmware ...
Retrieving modem firmware list ...
Newest firmware version available to download is '24.01.5x4_ATT'
Modem firmware update from '24.01.544_ATT' to '24.01.5x4_ATT' is needed
24.01.5x4_ATT
24.01.544_ATT

>
```

3. Use the **modem firmware ota list** command to list available firmware on the Digi firmware repository.

```
>  modem firmware ota list

Retrieving modem firmware list ...
25.20.664_CUST_044_3
25.20.666_CUST_067_1
25.20.663_CUST_040

>
```

4. Perform an OTA firmware update:
   - To perform an OTA firmware update by using the most recent available modem firmware from the Digi firmware repository, type:

```
>  modem firmware ota update

Checking for latest Generic firmware ...
Retrieving modem firmware list ...
Newest firmware version available to download is '25.20.666_CUST_
067_1'
Retrieving download location for modem firmware '25.20.666_CUST_067_
1' ...
```

```
>
```

- To perform an OTA firmware update by using a specific version from the Digi firmware repository, use the **version** parameter to identify the appropriate firmware version as determined by using **modem firmware ota list** command. For example::

```
>  modem firmware ota update version 24.01.5x4_ATT

Retrieving download location for modem firmware '24.01.5x4_ATT' ...
Downloading modem firmware '24.01.5x4_ATT' to '/opt/LE910C4_
NF/Custom_Firmware' ...
Modem firmware '24.01.5x4_ATT' downloaded
Updating modem firmware ...
Programming modem firmware ...

Found modem ...
Validate modem firmware ...
Getting ready for update ...
Stopping services ...
Running update pass 1 of 3 ...
Restarting services ...
-----------------------------
Successfully updated firmware
Modem firmware update complete

>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Update modem firmware by using a local firmware file

You can update your modem firmware by uploading a modem firmware file to your IX15 device. Firmware should be uploaded to /opt/*MODEM_MODEL*/Custom_Firmware, for example, /opt/LM940/Custom_Firmware.

Modem firmware can be downloaded from Digi here. Follow instructions on this page to determine the cellular module used by your device. After downloading, use tar or a similar unzipping tool to extract the firmware prior to uploading to the device. Note that the firmware file may not have a tar.gz extension, but it is a tar file and can be unzipped with tar or a similar tool. See Use the scp command for information about uploading files to the IX15 device.

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Use the **modem firmware check** command to determine if new modem firmware is available on local device.

```
> modem firmware check

Checking for latest ATT firmware in flash ...
Newest firmware version available in flash is '05.05.58.00_ATT_005.026_
000'
Modem firmware up to date
05.05.58.00_ATT_005.026_000

> modem firmware check
```

3. Use the **modem firmware list** command to list available firmware on the IX15 device.

```
>  modem firmware list

ATT, 24.01.544_ATT, current
Generic, 24.01.514_Generic, image
Verizon, 24.01.524_Verizon, image
ATT, 24.01.544_ATT, image
Sprint, 24.01.531-B003_Sprint, image

>
```

4. To perform an firmware update by using a local file, use the **version** parameter to identify the appropriate firmware version as determined using the **modem firmware check** or **modem firmware list** command. For example::

```
>  modem firmware update version 24.01.5x4_ATT

Updating modem firmware ...

----------------------------
Successfully updated firmware
Modem firmware update complete

>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Reboot your IX15 device

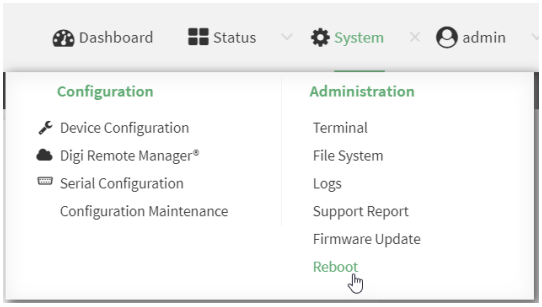
You can reboot the IX15 device immediately or schedule a reboot for a specific time every day.

Note You may want to save your configuration settings to a file before rebooting. See Save configuration to a file.

Reboot your device immediately

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. From the main menu, click **System**.
3. Click **Reboot**.



4. Click **Reboot** to confirm that you want to reboot the device.

Command line

1. Log into the IX15 command line as a user with Admin access.

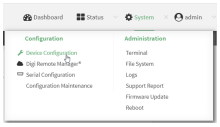
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the prompt, type:

```
> reboot
```

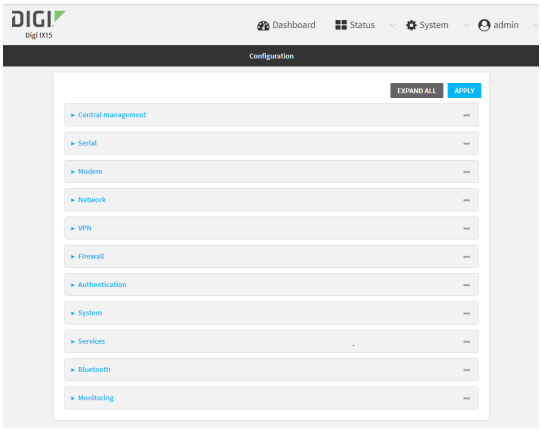
Schedule reboots of your device

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Select **System** > **Scheduled tasks**.
4. For **Reboot time**, enter the time of the day that the device should reboot, using the format *HH*:*MM*. The device will reboot at this time every day.

   If **Reboot time** is set, but the device is unable to synchronize its time with an NTP server, the device will reboot after it has been up for 24 hours. See System time for information about configuring NTP servers. If **Reboot window** is set, the reboot will occur during a random time within the reboot window.
5. For **Reboot window**, enter the maximum random delay that will be added to **Reboot Time**.

   Allowed values are any number of hours, minutes, or seconds, and take the format ***number*** {**h**|**m**|**s**}.

   For example, to set **parameter name** to ten minutes, enter **10m** or **600s**.
   The default is **10m**, and the maximum allowed time is **24h**.
6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Set the reboot time:

```
(config>> system schedule reboot_time time
(config)>
```

   where *time* is the time of the day that the device should reboot, using the format *HH*:*MM*. For example, the set the device to reboot at two in the morning every day:

```
(config>> system schedule reboot_time 02:00
(config)>
```

If **reboot_time** is set, but the device is unable to synchronize its time with an NTP server, the device will reboot after it has been up for 24 hours. See System time for information about configuring NTP servers. If **reboot_window** is set, the reboot will occur during a random time within the reboot window.

4. Set the maximum random delay that will be added to **reboot_time**:

```
(config>> system schedule reboot_window value
(config)>
```

where *value* is any number of hours, minutes, or seconds, and takes the format ***number*** {**h**|**m**|**s**}.

For example, to set **reboot_window** to ten minutes, enter either **10m** or **600s**:

```
(config)> system schedule reboot_window 600s
(config)>
```

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Erase device configuration and reset to factory defaults

You can erase the device configuration in the WebUI, at the command line, or by using the **ERASE** button on the device. Erasing the device configuration performs the following actions:

- Clears all configuration settings. When the device restarts, it uses the factory default configuration.
- Deletes all user files including Python scripts.
- Clears event and system log files.

Additionally, if the **ERASE** button is used to erase the configuration, pressing the **ERASE** button a second time immediately after the device has rebooted:

- Erases all automatically generated certificates and keys.
- With firmware release 22.2.9.x and newer, erases the client-side certificate used for communication with Digi Remote Manager.

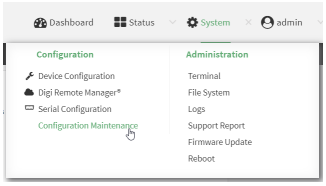
  If you are using Digi Remote Manager with firmware release 22.2.9.x and newer, by default the device uses a client-side certificate for communication with Remote Manager. If the client-side certificate is erased, you must use the Remote Manager interface to reset the certificate.

You can also reset the device to the default configuration without removing scripts, keys, and logfiles by using the **revert** command.

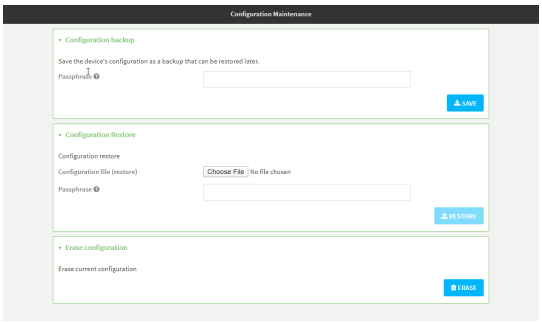
When resetting an IX15 to factory defaults, the XBee defaults are also restored and applied. Default values are those established by the user or factory values if no custom defaults are defined for the XBee device.

WebUI

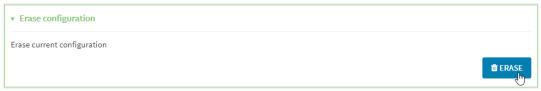
1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **System**. Under **Configuration**, click **Configuration Maintenance**.



The **Configuration Maintenance** windows is displayed.



3. In the **Erase configuration** section, click **ERASE**.



4. Click **CONFIRM**.
5. After resetting the device:
   a. Connect to the IX15 by using the serial port or by using an Ethernet cable to connect the IX15 **ETH** port to your PC.
   b. Log into the IX15:

      User name: Use the default user name: **admin**.

      Password: Use the unique password printed on the bottom label of the device (or the printed label included in the package).
   c. (Optional) Reset the default password for the admin account. See Change the default password for the admin user for further information.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Enter the following:

```
> system factory-erase
```

3. After resetting the device:
   a. Connect to the IX15 by using the serial port or by using an Ethernet cable to connect the IX15 **ETH** port to your PC.
   b. Log into the IX15:

      User name: Use the default user name: **admin**.

      Password: Use the unique password printed on the bottom label of the device (or the printed label included in the package).
   c. (Optional) Reset the default password for the admin account. See Change the default password for the admin user for further information.

Reset the device by using the ERASE button.

1. Locate the **ERASE** button on your device.



2. Press the **ERASE** button perform a device reset. The **ERASE** button has the following modes:
   - **Configuration reset**:
     - Press and release the **ERASE** button .
     - The device reboots automatically and resets to factory defaults. This does not remove any automatically generated certificates and keys.
   - **Full device reset**:
     - After the device reboots from the first button press, immediately press and releasethe **ERASE** button again.
     - The device reboots again and resets to factory defaults, as well as also removing generated certificates and keys.
3. After resetting the device:
   a. Connect to the IX15 by using the serial port or by using an Ethernet cable to connect the IX15 **ETH** port to your PC.
   b. Log into the IX15:

      User name: Use the default user name: **admin**.

      Password: Use the unique password printed on the bottom label of the device (or the printed label included in the package).

c. (Optional) Reset the default password for the admin account. See Change the default password for the admin user for further information.

Reset the device with the revert command

You can reset the device to the default configuration without removing scripts, keys, and logfiles by using the **revert** command:

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, enter **revert**:

```
(config)> revert
(config)>
```

4. Set the password for the admin user prior to saving the changes:

```
(config)> auth user admin password pwd
(config)>
```

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure the IX15 device to use custom factory default settings

You can configure your IX15 device to use custom factory default settings. This way, when you erase the device's configuration, the device will reset to your custom configuration rather than to the original factory defaults.

Note To clear the custom default configuration, press the ERASE button, wait for the device to reboot, then press the ERASE button again.

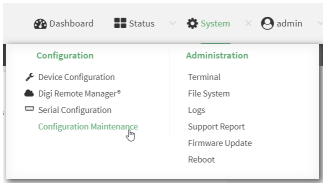
Required configuration items

- Custom factory default file

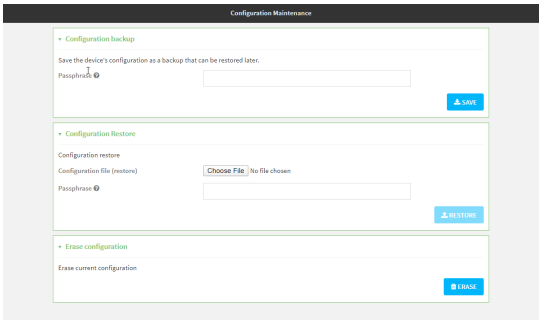
WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. Configure your IX15 device to match the desired custom factory default configuration.

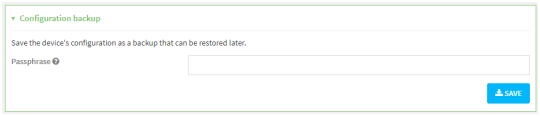
   For example, you may want to configure the device to use a custom APN or a particular network configuration, so that when you reset the device to factory defaults, it will automatically have your required network configuration.
3. On the main menu, click **System**. Under **Configuration**, click **Configuration Maintenance**.



   The **Configuration Maintenance** windows is displayed.

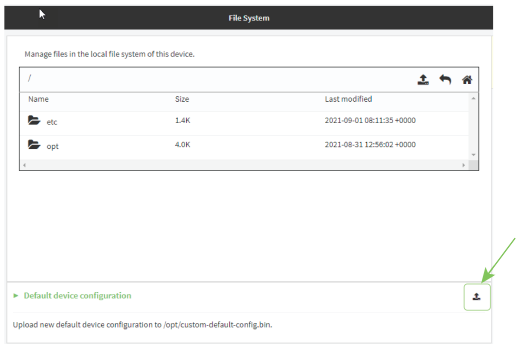


4. In the **Configuration backup** section, click **SAVE**.



   Do not set a **Passphrase** for the configuration backup. The file will be downloaded using your browser's standard download process.
5. After the configuration backup file has been downloaded, rename the file to:

   custom-default-config.bin
6. Upload the file to the device:
   a. From the main menu, select **System** > **Filesystem**.
   b. Under **Default device configuration**, click .



c. Select the file from your local file system.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Enter the following:

```
> system backup / type custom-defaults
Backup saved as /opt/custom-default-config.bin
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Locate the device by using the Find Me feature

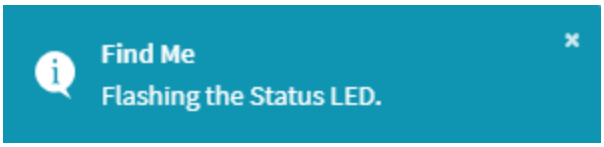
Use the **Find Me** feature to cause LEDs on the device to blink, which can help you to identify the specific device.

To use this feature:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Administration**, click **Find Me**.

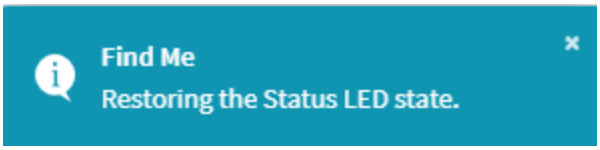
   A notification message appears, noting that the LED is flashing on the device. Click the **x** in the message to close it.



3. On the menu, click **System** again. Ablue circle next to **Find Me** is blinking, indicating that the **Find Me** feature is active.

4. To deactivate the **Find Me** feature, click **System** and click **Find Me** again.

   A notification message appears, noting that the LED is no longer flashing on the device. Click the **x** in the message to close it.



Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. To activate the **Find Me** feature, at the prompt, type the following at the command prompt:

```
> system find-me on
>
```

3. To deactivate the **Find Me** feature, type the following at the command prompt:

```
> system find-me off
>
```

4. To determine the status of the **Find Me** feature, type the following at the command prompt:

```
> system find-me status
off
>
```

Configuration files

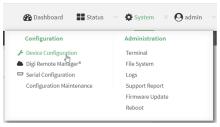
The IX15 configuration file, /etc/config/accns.json, contains all configuration changes that have been made to the device. It does not contain the complete device configuration; it only contains changes to the default configuration. Both the default configuration and the changes contained in the accns.json file are applied when the device reboots.

Save configuration changes

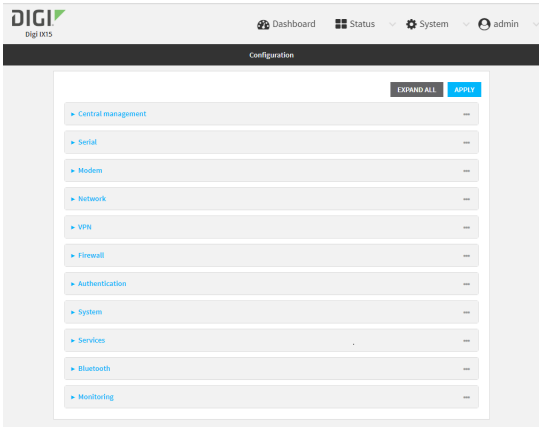
When you make changes to the IX15 configuration, the changes are not automatically saved. You must explicitly save configuration changes, which also applies the changes. If you do not save configuration changes, the system discards the changes.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Make any necessary configuration changes.
4. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Make any necessary configuration changes.
4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

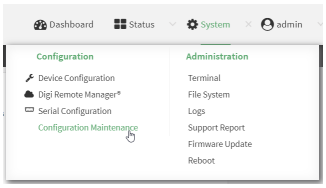
Save configuration to a file

You can save your IX15 device's configuration to a file and use this file to restore the configuration, either to the same device or to similar devices.

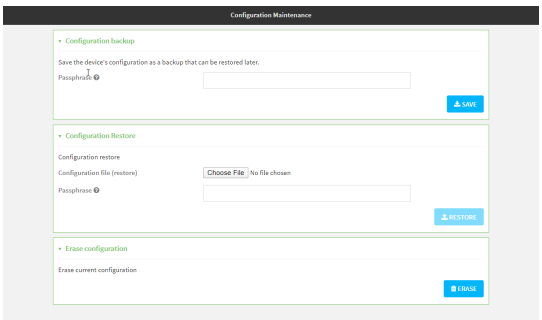
WebUI

This procedure creates a binary archive file containing the device's configuration, certificates and keys, and other information. It does not include the profile already applied to the local XBee, unless this profile is inside the **/etc/config** directory.

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **System**. Under **Configuration**, click **Configuration Maintenance**.



   The **Configuration Maintenance** windows is displayed.



3. In the **Configuration backup** section:
   a. (Optional) To encrypt the configuration using a passphrase, for **Passphrase (save/restore)**, enter the passphrase.
   b. Click **SAVE**.

   The file will be downloaded using your browser's standard download process.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Enter the following:

```
> system backup path [passphrase passphrase] type type
```

   where
   - *path* is the location on the IX15's filesystem where the configuration backup file should be saved.
   - *passphrase* (optional) is a passphrase used to encrypt the configuration backup.
   - *type* is the type of backup, either:
     - **archive**: Creates a binary archive file containing the device's configuration, certificates and keys, and other information.
     - **cli-config**: Creates a text file containing only the configuration changes.

   For example:

```
> system backup /etc/config/scripts/ type archive
```

3. (Optional) Use **scp** to copy the file from your device to another host:

```
> scp host hostname-or-ip user username remote remote-path local local-
path to remote
```

   where:
   - *hostname-or-ip* is the hostname or ip address of the remote host.
   - *username* is the name of the user on the remote host.
   - *remote-path* is the location on the remote host where the file will be copied.
   - *local-path* is the path and filename on the IX15 device.

   For example:

```
> scp host 192.168.4.1 user admin remote /home/admin/bin/ local
/etc/config/backup-archive-0040FF800120-19.05.17-19.01.17.bin to remote
```

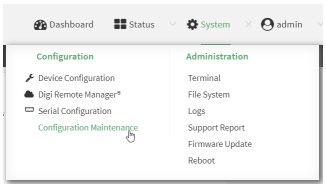
Restore the device configuration

You can restore a configuration file to your IX15 device by using a backup from the device, or a backup from a similar device.

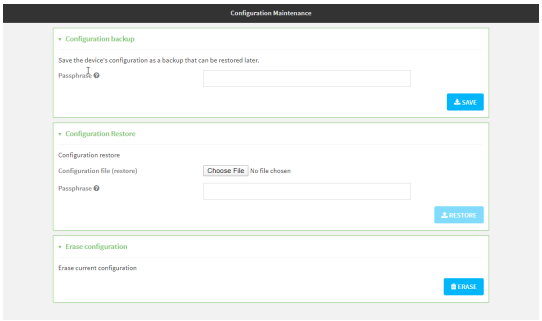
This process does not apply any profile to the IX15's XBee, a custom profile must be applied manually or programmatically after the backup restoration from the WebUI, with the CLI or using the XBee API.

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **System**. Under **Configuration**, click **Configuration Maintenance**.



The **Configuration Maintenance** windows is displayed.



3. In the **Configuration Restore** section:
   a. If a passphrase was used to create the configuration backup, for **Passphrase (save/restore)**, enter the passphrase.
   b. Under **Configuration Restore**, click **Choose File**.
   c. Browse to the system firmware file location on your local computer and select the file.
   d. Click **RESTORE**.
4. Click **CONFIRM**.

   The configuration will be restored and the device will be rebooted.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. If the configuration backup is on a remote host, use **scp** to copy the file from the host to your device:

```
> scp host hostname-or-ip user username remote remote-path local local-path to local
```

   where:

   - *hostname-or-ip* is the hostname or ip address of the remote host.
   - *username* is the name of the user on the remote host.

- *remote-path* is the path and filename of the file on the remote host that will be copied to the IX15 device.
- *local-path* is the location on the IX15 device where the copied file will be placed.

For example:

```
> scp host 192.168.4.1 user admin remote /home/admin/bin/backup-archive-
0040FF800120-22.2.9.85-19.23.42.bin local /opt to local
```

3. Enter the following:

```
> system restore filepath [passphrase passphrase]
```

where

- *filepath* is the the path and filename of the configuration backup file on the IX15's filesystem (*local-path* in the previous step).
- *passphrase* (optional) is the passphrase to restore the configuration backup, if a passphrase was used when the backup was created.

For example:

```
> system restore /opt/backup-archive-0040FF800120-22.2.9.85-
19.23.42.bin
```

Schedule system maintenance tasks

You can configure tasks to be run during a specified maintenance window. When the device is within its maintenance window, firmware updates and Digi Remote Manager configuration checks will be performed.

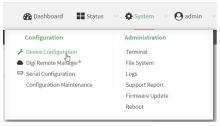
You can also schedule custom scripts to run during the maintenance window. See Configure scripts to run automatically for more information.

Required configuration items

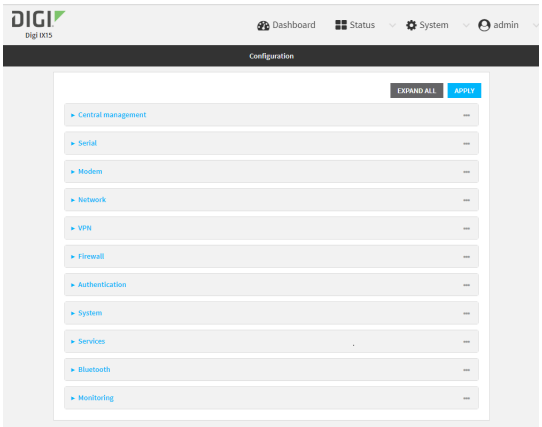
- Events that trigger the maintenance window to begin.
- Whether all configured triggers, or only one of the triggers, must be met.
- The tasks to be performed. Options are:
  - Firmware updates.
  - Digi Remote Manager configuration check.
- Whether the device will check for updates to the device firmware.
- Whether the device will check for updates to the modem firmware.
- The frequency (daily, weekly, or monthly) that checks for firmware updates will run.

WebUI

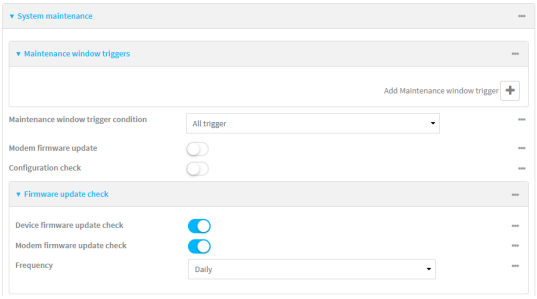
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **System** > **Scheduled tasks** > **System maintenance**.



4. Click to expand **Maintenance window triggers**.
5. Click ✚ to add a maintenance window trigger.



6. For **Maintenance window trigger type**, select one of the following:
   - **Check if interface is up**, for **Test Interface**, select the interface.
   - **Time period for maintenance window**:
     a. Click to expand **Maintenance window**.
     b. For **Start time**, type the time of day that the maintenance window should start, using the syntax *HH*:*MM*. If **Start time** is not set, maintenance tasks are not scheduled and will not be run.

        The behavior of **Start time** varies depending on the setting of **Duration window**, which is configured in the next step.

        - If **Duration window** is set to **Immediately**, all scheduled tasks will begin at the exact time specified in **Start time**.
        - If **Duration window** is set to **24 hours**, **Start time** is effectively obsolete and the maintenance tasks will be scheduled to run at any time. Setting **Duration window** to **24 hours** can potentially overstress the device and should be used with caution.
        - If **Duration window** is set to any value other than to **Immediately** or **24 hours**, the maintenance tasks will run at a random time during the time allotted for the duration window.
        - If **Duration window** is set to one or more hours, the minutes field in **Start time** is ignored and the duration window will begin at the beginning of the specified hour.

     c. For **Duration window**, select the amount of time that the maintenance tasks will be run. If **Immediately** is selected, all scheduled tasks will begin at the exact time specified in **Start time**.
     d. For **Frequency**, select whether the maintenance window will be started every day, or once per week.

- If **Check if Python Out-of-Service** is set, the maintenance window will only start if the Python Out-of-Service is set. See Use Python to set the maintenance window for further information.

7. (Optional) Click to enable **Device firmware update** to instruct the system to look for any updated device firmware during the maintenance window. If updated firmware is found, it will then be installed.

   Note If your device is managed by a Digi Remote Manager configuration, the configuration manages the device's firmware version. You should not enable this option.

8. (Optional) Click to enable **Modem firmware update** to instruct the system to look for any updated modem firmware during the maintenance window. If updated firmware is found, it will then be installed. Modem firmware update looks for updated firmware both on the local device and over the network, using either a WAN or cellular connection.
9. (Optional) Click to enable **Configuration check** to allow for the configuration to be updated, including by custom scripts, during the maintenance window.
10. (Optional) Configure automated checking for device firmware updates:
    a. Click to expand **Firmware update check**.
    b. **Device firmware update check** is enabled by default. This enables to automated checking for device firmware updates.
    c. **Modem firmware update check** is enabled by default. This enables to automated checking for modem firmware updates.
    d. For Frequency, select how often automated checking for device and modem firmware should take place. Allowed values are **Daily**, **Weekly**, and **Monthly**. The default is **Daily**.
11. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Configure a system maintenance trigger:
   a. Add a trigger:

```
(config)> add system schedule maintenance trigger end
(config)>
```

b. Set the type of trigger:

```
(config add system schedule maintenance trigger)> type value
(config)>
```

where *value* is one of:

- **interface_up**: If **interface_up** is set:
  i. Set the interface:

  ```
  (config add system schedule maintenance trigger)> interface
  value
  (config)>
  ```

  ii. i. Use the **?** to determine available interfaces:
  ii. Set the interface. For example:

  ```
  (config system schedule maintenance trigger 0)> interface
  /network/interface/eth1
  (config system schedule maintenance trigger 0)>
  ```

- **out_of_service**: The maintenance window will only start if the Python Out-of-Service is set. See Use Python to set the maintenance window for further information.
- **time**: Configure a time period for the maintenance window:
  i. Configure the time of day that the maintenance window should start, using the syntax *HH:MM*. If the start time is not set, maintenance tasks are not scheduled and will not be run.

  ```
  (config system schedule maintenance trigger 0)> time from
  HH:MM
  (config system schedule maintenance trigger 0)>
  ```

  The behavior of the start time varies depending on the setting of the duration length, which is configured in the next step.
  - If the duration length is set to **0**, all scheduled tasks will begin at the exact time specified in the start time.
  - If the duration length is set to **24 hours**, the start time is effectively obsolete and the maintenance tasks will be scheduled to run at any time. Setting the duration length to **24 hours** can potentially overstress the device and should be used with caution.
  - If the duration length is set to any value other than to **0** or **24 hours**, the maintenance tasks will run at a random time during the time allotted for the duration window.
  - If the duration length is set to one or more hours, the minutes field in the start time is ignored and the duration window will begin at the beginning of the specified hour.

  ii. Configure the duration length (the amount of time that the maintenance tasks will be run). If **0** is used, all scheduled tasks will begin at the start time, defined

in the previous step.

```
(config system schedule maintenance trigger 0)> length num
(config system schedule maintenance trigger 0)>
```

where *num* is any whole number between **0** and **24**.

iii. Configure the frequency that the maintenance tasks should be run:

```
(config system schedule maintenance trigger 0)> frequency
value
(config system schedule maintenance trigger 0)>
```

where *value* is either **daily** or **weekly**. **Daily** is the default.

4. (Optional)) Configure the device to look for any updated device firmware during the maintenance window. If updated firmware is found, it will then be installed. The device will look for updated firmware both on the local device and over the network, using either a WAN or cellular connection.

```
(config)> system schedule maintenance device_fw_update value
(config)>
```

where *value* is either **true** or **false**. **yes** or **no**, and **1** or **0** are also allowed.

Note If your device is managed by a Digi Remote Manager configuration, the configuration manages the device's firmware version. You should not enable this option.

5. (Optional) Configure the device to look for any updated modem firmware during the maintenance window. If updated firmware is found, it will then be installed. The device will look for updated firmware both on the local device and over the network, using either a WAN or cellular connection.

```
(config)> system schedule maintenance modem_fw_update value
(config)>
```

where *value* is either **true** or **false**. **yes** or **no**, and **1** or **0** are also allowed.

6. (Optional) Configure automated checking for device firmware updates:
   a. **Device firmware update check** is enabled by default. This enables to automated checking for device firmware updates. To disable:

```
(config)> system schedule maintenance firmware_update_check device
false
(config)>
```

   b. Set how often automated checking for device firmware should take place:

```
(config)> system schedule maintenance frequency value
(config)>
```

   where *value* is either **daily**, **weekly**, or **monthly**. **daily** is the default.

7. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

8. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Disable device encryption

You can disable the cryptography on your IX15 device. This can be used to ship unused devices from overseas without needing export licenses from the country from which the device is being shipped.

When device encryption is disabled, the following occurs:

- The device is reset to the default configuration and rebooted.
- After the reboot:
  - Access to the device via the WebUI and SSH are disabled.
  - All internet connectivity is disabled, including WAN and WWAN. Connectivity to central management software is also disabled.
  - All IP networks and addresses are disabled except for the default 192.168.210.1/24 network on the local LAN Ethernet port. DHCP server is also disabled.

    The device can only be accessed by using telnet from a local machine connecting to the 192.168.210.1/24 network.

Disabling device encryption is not available in the WebUI. It can only be performed from the Admin CLI.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Disable encryption with the following command:

```
> system disable-cryptography
>
```

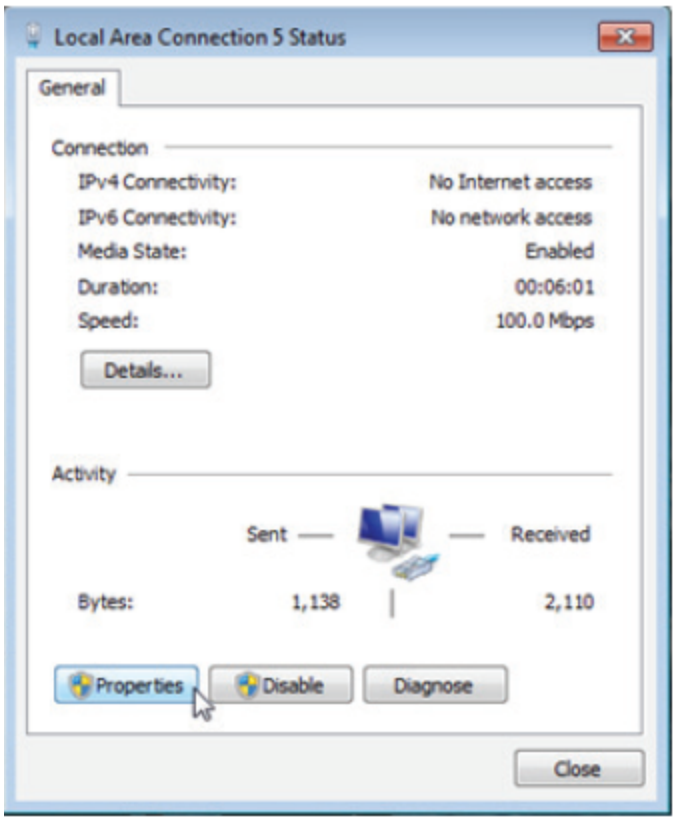
3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Re-enable cryptography after it has been disabled.

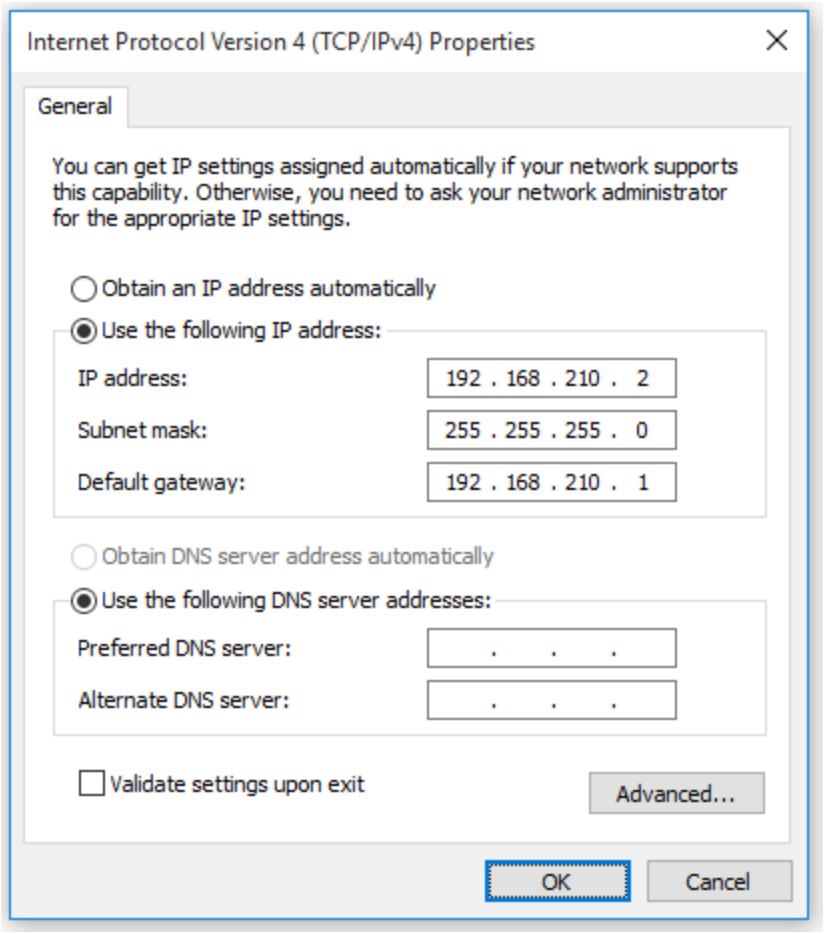
To re-enable cryptography:

1. Configure your PC network to connect to the 192.168.210 subnet. For example, on a Windows PC:
   a. Select the **Properties** of the relevant network connection on the Windows PC.



   b. Click the **Internet Protocol Version 4 (TCP/IPv4)** parameter.
   c. Click **Properties**. The **Internet Protocol Version 4 (TCP/IPv4) Properties** dialog appears.
   d. Configure with the following details:
      - **IP address** for PC: 192.168.210.2
      - **Subnet**: 255.255.255.0

- **Gateway**: 192.168.210.1



2. Connect the PC's Ethernet port to the Ethernet port on your IX15 device.
3. Open a telnet session and connect to the IX15 device at the IP address of 192.168.210.1.
4. Log into the device:
   - Username: **admin**
   - Password: The default unique password for your device is printed on the device label.
5. At the shell prompt, type:

```
# rm /etc/config/.nocrypt
# flatfsd -i
```

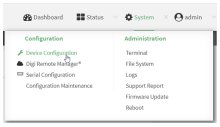
This will re-enable encryption and leave the device at its factory default setting.

Configure the speed of your Ethernet port

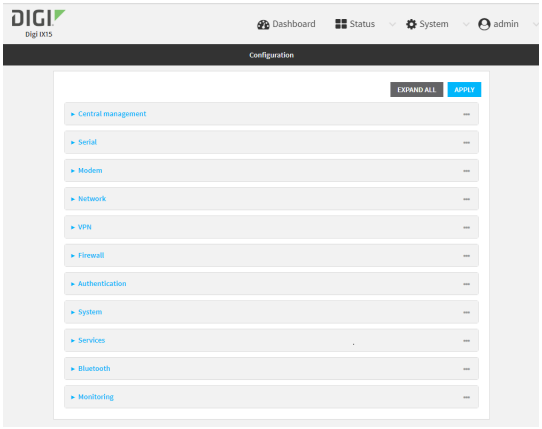
You can configure the speed of your IX15 device's Ethernet port.

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Network** > **Device** > ETH.
4. For **Speed**, select the appropriate speed for the Ethernet port, or select **Auto** to automatically detect the speed. The default is **Auto**.
5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, type:

```
(config)> network device eth_port value
```

where:

- *eth_port* is the name of the Ethernet port (for example, **eth**)
- *value* is one of:

- **10**—Sets the speed to 10 Mbps.
- **100**—Sets the speed to 100 Mbps.
- **1000**—Sets the speed to 1 Gbps. Available only for devices with Gigabit Ethernet ports.

  auto—Configures the device to automatically determine the best speed for the Ethernet port.

The default is **auto**.

4. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Monitoring

This chapter contains the following topics:

intelliFlow 773
Configure NetFlow Probe 780

intelliFlow

intelliFlow monitors system information, network data usage, and traffic information, and displays the information in a series of charts available in the local WebUI. To use intelliFlow, the IX15 must be powered on and you must have access to the local WebUI. Once you enable intelliFlow, the **Status** > **intelliFlow** option is available in the main menu. By default, intelliFlow is disabled.

intelliFlow provides charts on the following information:

- System utilisation
- Top data usage by host
- Top data usage by server
- Top data usage by service
- Host data usage over time

intelliFlow charts are dymanic; at any point, you can click inside the chart to drill down to view more granular information, and menu options allow you to change various aspects of the information being displayed.

Note When intelliFlow is enabled and the device is connected to Digi aView, it adds an estimated 50MB of data usage for the device by reporting the metrics to aView. intelliflow does not currently work with Digi Remote Manager.

Enable intelliFlow

Required configuration items

- Enable intelliFlow.

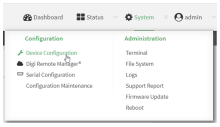
Additional configuration items

- The firewall zone for internal clients being monitored by intelliFlow.

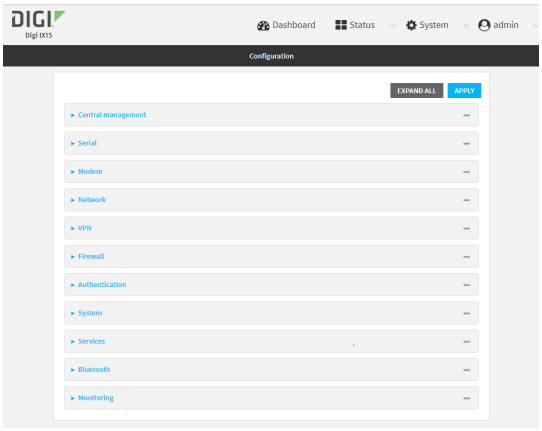
To enable intelliFlow:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

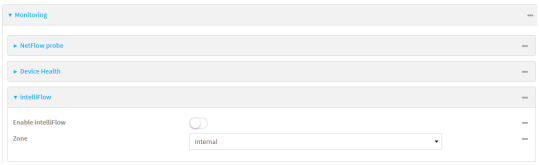


The **Configuration** window is displayed.



3. Click **Monitoring** > **intelliFlow**.

   The intelliFlow configuration window is displayed.



4. Click **Enable intelliFlow**.
5. For **Zone**, select the firewall zone. Internal clients that are being monitored by IntelliFlow should be present on the specified zone.
6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable IntelliFlow:

```
(config)> monitoring intelliflow enable true
```

4. Set the firewall zone. Internal clients that are being monitored by IntelliFlow should be present on the specified zone:

a. Determine available zones:

```
(config)> monitoring intelliflow zone ?

Zone: The firewall zone which is assigned to the network interface(s)
that
intelliFlow will see as internal clients.  intelliFlow relies on an
internal to
external relationship, where the internal clients are present on the
zone specified.
Format:
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup
Default value: internal
Current value: internal

(config)>
```

b. Set the zone to be used by IntelliFlow:

```
(config)> monitoring intelliflow zone my_zone
```

5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Use intelliFlow to display average CPU and RAM usage

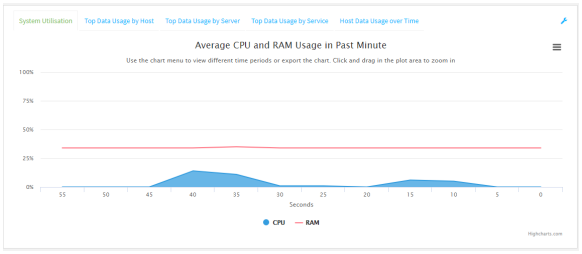
This procedure is only available from the WebUI.

To display display average CPU and RAM usage:

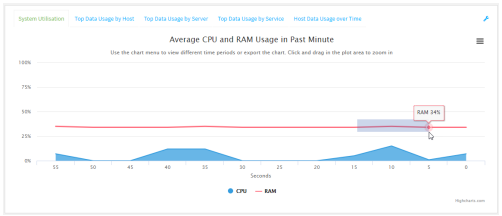
WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. If you have not already done so, enable intelliFlow. See Enable intelliFlow.
3. From the menu, click **Status** > **intelliFlow**.

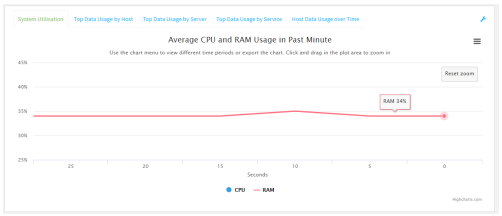
The System Utilisation chart is displayed:



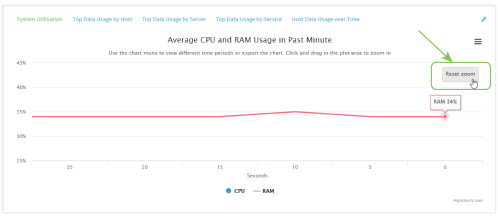
- Display more granular information:
  1. Click and drag over an area in the chart to zoom into that area and provide more granular information.



  2. Release to display the selected portion of the chart:



3. Click **Reset zoom** to return to the original display:

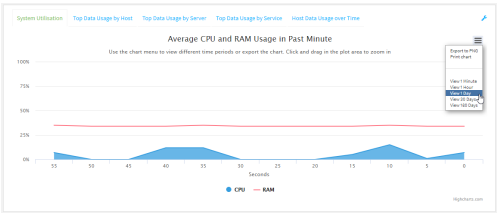


- Change the time period displayed by the chart.

  By default, the **System utilisation** chart displays the average CPU and RAM usage over the last minute. You can change this to display the average CPU and RAM usage:

  - Over the last hour.
  - Over the last day.
  - Over the last 30 days.
  - Over the last 180 days.

  1. Click the menu icon (☰).
  2. Select the time period to be displayed.



- Save or print the chart.

  1. Click the menu icon (☰).
  2. To save the chart to your local filesystem, select **Export to PNG**.
  3. To print the chart, select **Print chart**.

Use intelliFlow to display top data usage information

With intelliFlow, you can display top data usage information based on the following:

- Top data usage by host
- Top data usage by server
- Top data usage by service

To generate a top data usage chart:

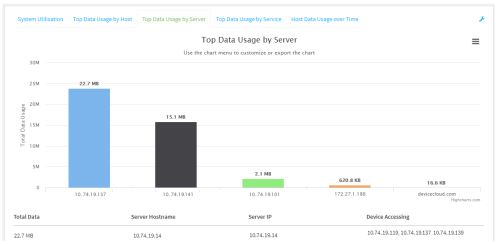
WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. If you have not already done so, enable intelliFlow. See Enable intelliFlow.
3. From the menu, click **Status** > **intelliFlow**.

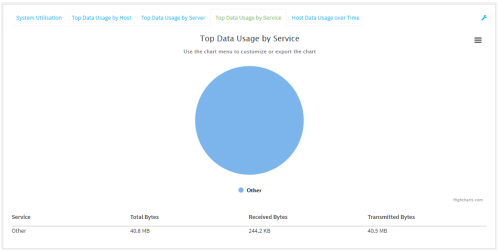
4. Display a data usage chart:
   - To display the **Top Data Usage by Host** chart, click **Top Data Usage by Host**.



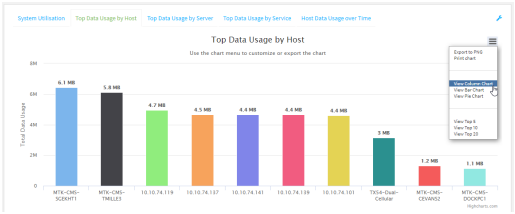
   - To display the **Top Data Usage by Server** chart, click **Top Data Usage by Server**.



   - To display the **Top Data Usage by Service** chart, click **Top Data Usage by Service**.

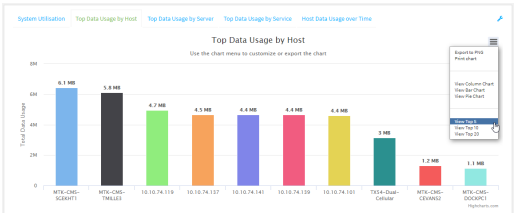


5. Change the type of chart that is used to display the data:
   a. Click the menu icon (☰).
   b. Select the type of chart.



6. Change the number of top users displayed.
   You can display the top five, top ten, or top twenty data users.

a. Click the menu icon (☰).
b. Select the number of top users to displayed.



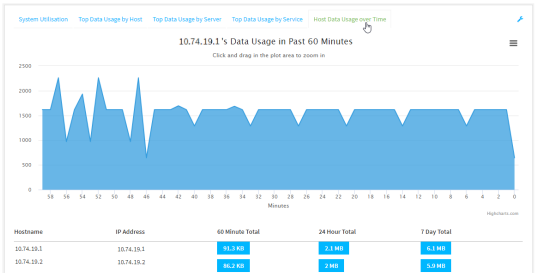
7. Save or print the chart.
   a. Click the menu icon (☰).
   b. To save the chart to your local filesystem, select **Export to PNG**.
   c. To print the chart, select **Print chart**.

Use intelliFlow to display data usage by host over time

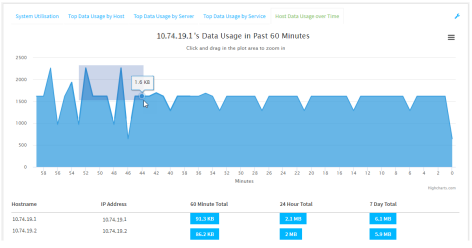
To generate a chart displaying a host's data usage over time:

WebUI

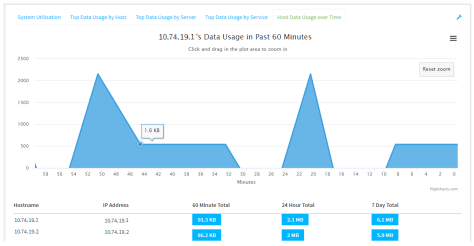
1. Log into the IX15 WebUI as a user with Admin access.
2. If you have not already done so, enable intelliFlow. See Enable intelliFlow.
3. From the menu, click **Status** > **intelliFlow**.
4. Click **Host Data Usage Over Time**.



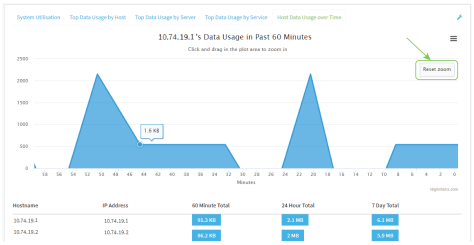
- Display more granular information:
  a. Click and drag over an area in the chart to zoom into that area and provide more granular information.



b. Release to display the selected portion of the chart:



c. Click **Reset zoom** to return to the original display:



- Save or print the chart.
  a. Click the menu icon (☰).
  b. To save the chart to your local filesystem, select **Export to PNG**.
  c. To print the chart, select **Print chart**.

Configure NetFlow Probe

NetFlow probe is used to probe network traffic on the IX15 device and export statistics to NetFlow collectors.

Required configuration items

- Enable NetFlow.
- The IP address of a NetFlow collector.

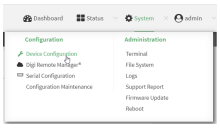
Additional configuration items

- The NetFlow version.
- Enable flow sampling and select the flow sampling technique.
- The number of flows from which the flow sampler can sample.
- The number of seconds that a flow is inactive before it is exported to the NetFlow collectors.
- The number of seconds that a flow is active before it is exported to the NetFlow collectors.
- The maximum number of simultaneous flows.
- A label for the NetFlow collector.
- The port of the NetFlow collector.
- Additional NetFlow collectors.

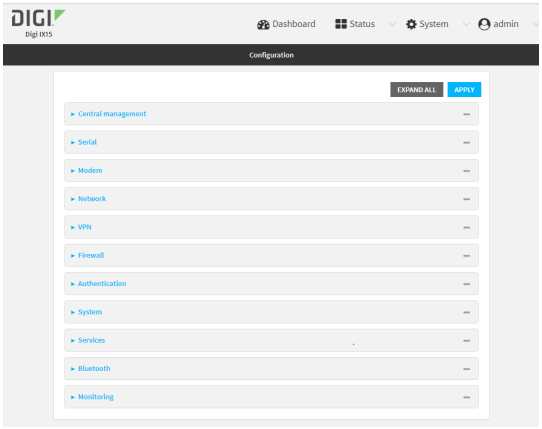
To probe network traffic and export statistics to NetFlow collectors:

WebUI

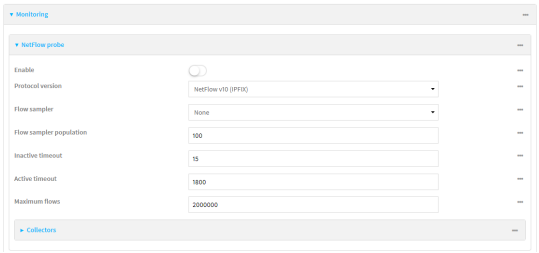
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Monitoring** > **NetFlow probe**.



4. **Enable** NetFlow probe.
5. **Protocol version:** Select the **Protocol version**. Available options are:
   - **NetFlow v5**—Supports IPv4 only.
   - **NetFlow v9**—Supports IPv4 and IPv6.
   - **NetFlow v10 (IPFIX)**—Supports both IPv4 and IPv6 and includes IP Flow Information Export (IPFIX).

   The default is **NetFlow v10 (IPFIX)**.
6. Enable **Flow sampler** by selecting a sampling technique. Flow sampling can reduce flow processing and transmission overhead by providing a representative subset of all flows. Available options are:
   - **None**—No flow sampling method is used. Each flow is accounted.
   - **Deterministic**—Selects every *n*th flow, where *n* is the value of **Flow sampler population**.

- **Random**—Randomly selects one out of every *n* flows, where *n* is the value of **Flow sampler population**.
- **Hash**—Randomly selects one out of every *n* flows using the hash of the flow key, where *n* is the value of **Flow sampler population**.

7. For **Flow sampler population**, if you selected a flow sampler, enter the number of flows for the sampler. Allowed value is any number between **2** and **16383**. The default is **100**.
8. For **Inactive timeout**, type the the number of seconds that a flow can be inactive before sent to a collector. Allowed value is any number between **1** and **15**. The default is **15**.
9. For **Active timeout**, type the number of seconds that a flow can be active before sent to a collector. Allowed value is any number between **1** and **1800**. The default is **1800**.
10. For **Maximum flows**, type the maximum number of flows to probe simultaneously. Allowed value is any number between **0** and **2000000**. The default is **2000000**.
11. Add collectors:
    a. Click to expand **Collectors**.
    b. For **Add Collector**, click ✚.
    c. (Optional) Type a **Label** for the collector.
    d. For **Address**, type the IP address of the collector.
    e. (Optional) For **Port**, enter the port number used by the collector. The default is 2055.

    Repeat to add additional collectors.
12. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Enable NetFlow:

```
(config)> monitoring netflow enable true
(config)>
```

4. Set the protocol version:

```
(config)> monitoring netflow protocol version
                                                (config)>
```

where *version* is one of:

- **v5**—NetFlow v5 supports IPv4 only.
- **v9**—NetFlow v9 supports IPv4 and IPv6.
- **v10**—NetFlow v10 (IPFIX) supports both IPv4 and IPv6 and includes IP Flow Information Export (IPFIX).

The default is **v10**.

1. Enable flow sampling by selecting a sampling technique. Flow sampling can reduce flow processing and transmission overhead by providing a representative subset of all flows.

```
(config)> monitoring netflow sampler type
(config)>
```

where *type* is one of:

- **none**—No flow sampling method is used. Each flow is accounted.
- **deterministic**—Selects every *n*th flow, where *n* is the value of the flow sample population.
- **random**—Randomly selects one out of every *n* flows, where *n* is the value of the flow sample population.
- **hash**—Randomly selects one out of every *n* flows using the hash of the flow key, where *n* is the value of the flow sample population.

5. If you are using a flow sampler, set the number of flows for the sampler:

```
(config)> monitoring netflow sampler_population value
(config)>
```

where *value* is any number between **2** and **16383**. The default is **100**.

6. Set the number of seconds that a flow can be inactive before sent to a collector:

```
(config)> monitoring netflow inactive_timeout value
(config)>
```

where *value* is any is any number between **1** and **15**. The default is **15**.

7. Set the number of seconds that a flow can be active before sent to a collector:

```
(config)> monitoring netflow active_timeout value
(config)>
```

where *value* is any is any number between **1** and **1800**. The default is **1800**.

8. Set the maximum number of flows to probe simultaneously:

```
(config)> monitoring netflow max_flows value
(config)>
```

where *value* is any is any number between **0** and **2000000**. The default is **2000000**.

9. Add collectors:
   a. Add a collector:

```
(config)> add monitoring netflow collector end
(config monitoring netflow collector 0)>
```

   b. Set the IP address of the collector:

```
(config monitoring netflow collector 0)> address ip_address
(config monitoring netflow collector 0)>
```

   c. (Optional) Set the port used by the collector:

```
(config monitoring netflow collector 0)> port port
(config monitoring netflow collector 0)>
```

   d. (Optional) Set a label for the collector:

```
(config monitoring netflow collector 0)> label "This is a collector."
(config monitoring netflow collector 0)>
```

   Repeat to add additional collectors.
10. Save the configuration and apply the change:

```
(config monitoring netflow collector 0)> save
Configuration saved.
>
```

11. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Central management

This chapter contains the following topics:

Digi Remote Manager support 786
Certificate-based enhanced security 786
Configure Digi Remote Manager 786
Collect device health data and set the sample interval 793
Enable event log upload to Digi Remote Manager 796
Log into Digi Remote Manager 798
Use Digi Remote Manager to view and manage your device 799
Add a device to Digi Remote Manager 800
View Digi Remote Manager connection status 800
Configure multiple devices using profiles 801
Amazon AWS IoT 802
Microsoft Azure 802
Learn more 802

Digi Remote Manager support

Digi Remote Manager is a hosted remote configuration and management system that allows you to remotely manage a large number of devices. Remote Manager includes a web-based interface that you can use to perform device operations, such as viewing and changing device configurations and performing firmware updates. Remote Manager servers also provide a data storage facility. The Digi Remote Manager is the default cloud-based management system, and is enabled by default. You can also select to use Digi aView as the cloud-based management system. See *Digi aView User Guide* for information about aView.

To use Remote Manager, you must set up a Remote Manager account. To set up a Remote Manager account and learn more about Digi Remote Manager, go to www.digi.com/products/cloud/digi-remote-manager.

To learn more about Remote Manager features and functions, see the *Digi Remote Manager User Guide*.

Certificate-based enhanced security

Beginning with firmware version 22.2.9.x, the default URL for the device's Remote Manager connection is edp12.devicecloud.com. This URL is required to utilize the client-side certificate support. Prior to release 22.2.9.x, the default URL was my.devicecloud.com.

- If your Digi device is configured to use a non-default URL to connect to Remote Manager, updating the firmware will not change your configuration. However, if you erase the device's configuration, the Remote Manager URL will change to the default of edp12.devicecloud.com.
- If you perform a factory reset by pressing the **ERASE** twice, the client-side certificate will be erased and you must use the Remote Manager interface to reset the certificate. Select the device in Remote Manager and select **Actions** > **Reset Device Certificate**.

The new URL of edp12.devicecloud.com is for device communication only. Use my.devicecloud.com for user interaction with remote manager.

Firewall issues

To utilize the certificate-based security, you may need to open a port through your firewall for egress connectivity to edp12.devicecloud.com. TCP port 3199 is used for communication with Remote Manager.

Configure Digi Remote Manager

By default, your IX15 device is configured to use central management using Digi Remote Manager.

Additional configuration options

These additional configuration settings are not typically configured, but you can set them as needed:

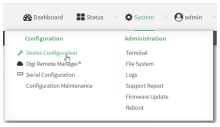
- Disable the Digi Remote Manager connection if it is not required. You can also configure an alternate cloud-based central management application.
- Change the reconnection timer.
- The non-cellular keepalive timeout.
- The cellular keepalive timeout.
- The keepalive count before the Remote Manager connection is dropped.

- SMS support.
- HTTP proxy server support.

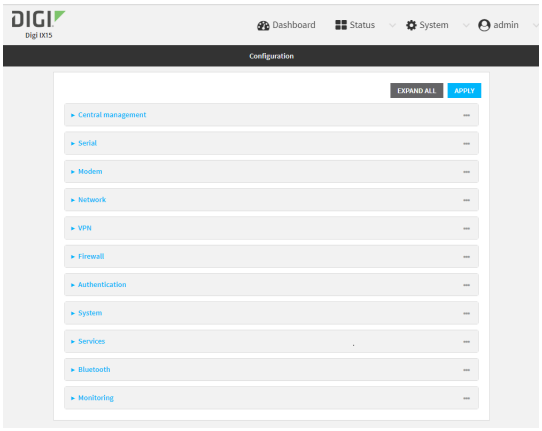
To configure Digi Remote Manager:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

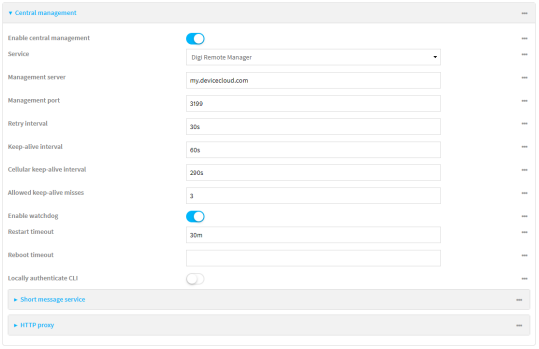


The **Configuration** window is displayed.



3. Click **Central management**.

   The Central management configuration window is displayed.



   Digi Remote Manager support is enabled by default. To disable, click **Enable central management**.

4. (Optional) For **Service**, select either **Digi Remote Manager** or **Digi aView**. The default is **Digi Remote Manager**.
5. (Optional) For **Management server**, type the URL for the central management server.

   The default varies depending on firmware versions:

   - Firmware version 22.2.9.x and newer, the default is the edp12.devicecloud.com. This server is for device-connectivity only, and uses enhanced security through certificate-based communication. See Digi Remote Manager support for further infomation.
   - Firmware prior to version 22.2.9.x, the default is the Digi Remote Manager server, my.devicecloud.com.

6. (Optional) For **Management port**, type the destination port for the remote cloud services connection. The default is **3199**.
7. (Optional) For **Retry interval**, type the amount of time that the IX15 device should wait before reattempting to connect to remote cloud services after being disconnected. The default is 30 seconds.

   Allowed values are any number of hours, minutes, or seconds, and take the format ***number*** {**h**|**m**|**s**}.

   For example, to set **Retry interval** to ten minutes, enter **10m** or **600s**.
8. (Optional) For **Keep-alive interval**, type the amount of time that the IX15 device should wait between sending keep-alive messages to remote cloud services when using a non-cellular interface. The default is 60 seconds.

   Allowed values are any number of hours, minutes, or seconds, and take the format ***number*** {**h**|**m**|**s**}.

   For example, to set **Keep-alive interval** to ten minutes, enter **10m** or **600s**.
9. (Optional) For **Cellular keep-alive interval**, type the amount of time that the IX15 device should wait between sending keep-alive messages to remote cloud services when using a cellular interface. The default is 290 seconds.

   Allowed values are any number of hours, minutes, or seconds, and take the format ***number*** {**h**|**m**|**s**}.

   For example, to set **Cellular keep-alive interval** to ten minutes, enter **10m** or **600s**.

10. (Optional) For **Allowed keep-alive misses**, type the number of allowed keep-alive misses. The default is **3**.
11. **Enable watchdog** is used to monitor the connection to remote cloud services. If the connection is down, you can configure the device to restart the connection, or to reboot. The watchdog is enabled by default.
12. If **Enable watchdog** is enabled:
    a. (Optional) For **Restart Timeout**, type the amount of time to wait before restarting the connection to the remote cloud services, once the connection is down.

    Allowed values are any number of hours, minutes, or seconds, and take the format ***number***{**h|m|s**}.

    For example, to set **Restart Timeout** to ten minutes, enter **10m** or **600s**.

    The minimum value is 30 minutes and the maximum is 48 hours. If not set, this option is disabled. The default is 30 minutes.

    b. (Optional) For **Reboot Timeout**, type the amount of time to wait before rebooting the device, once the connection to the remote cloud servicesis down. By default, this option is not set, which means that the option is disabled.

    Allowed values are any number of hours, minutes, or seconds, and take the format ***number***{**h|m|s**}.

    For example, to set **Reboot Timeout** to ten minutes, enter **10m** or **600s**.

    The minimum value is 30 minutes and the maximum is 48 hours. If not set, this option is disabled. The default is disabled.
13. (Optional) Enable **Locally authenticate CLI** to require a login and password to authenticate the user from the remote cloud services CLI. If disabled, no login prompt will be presented and the user will be logged in as **admin**. The default is disabled.
14. (Optional) Configure the IX15 device to communicate with remote cloud services by using SMS:

    a. Click to expand **Short message service**.
    b. **Enable** SMS messaging.
    c. For **Destination phone number**, type the phone number for the remote cloud services.
    d. (Optional) Type the **Service identifier**.
15. (Optional) Configure the IX15 device to communicate with remote cloud services by using an HTTP proxy server:
    a. Click to expand **HTTP Proxy**.
    b. **Enable** the use of an HTTP proxy server.
    c. For **Server**, type the hostname of the HTTP proxy server.
    d. For **Port**, type or select the port number on the HTTP proxy server that the device should connect to. The default is **2138**.
16. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Digi Remote Manager support is enabled by default. To disable Digi Remote Manager support:

```
(config)> cloud enable false
(config)>
```

4. (Optional) Set the service:

```
(config)> cloud service value
(config)>
```

   where *value* is either:

   - **drm**: Digi Remote Manager
   - **aview**: Digi aView

   The default is Digi Remote Manager.
5. (Optional) Set the URL for the central management server. The default is the Digi Remote Manager server, my.devicecloud.com.

```
(config)> cloud drm drm_url url
(config)>
```

6. (Optional) Set the amount of time that the IX15 device should wait before reattempting to connect to the remote cloud services after being disconnected. The minimum value is ten seconds. The default is 30 seconds.

```
(config)> cloud drm retry_interval value
```

   where *value* is any number of hours, minutes, or seconds, and takes the format ***number*** {**h**|**m**|**s**}.

   For example, to set **the retry interval** to ten minutes, enter either **10m** or **600s**:

```
(config)> cloud drm retry_interval 600s
(config)>
```

7. (Optional) Set the amount of time that the IX15 device should wait between sending keep-alive messages to the Digi Remote Manager when using a non-cellular interface. Allowed values are from 30 seconds to two hours. The default is 60 seconds.

```
(config)> cloud drm keep_alive value
(config)>
```

   where *value* is any number of hours, minutes, or seconds, and takes the format ***number*** {**h**|**m**|**s**}.

   For example, to set **the keep-alive interval** to ten minutes, enter either **10m** or **600s**:

```
(config)> cloud drm keep_alive 600s
(config)>
```

8. (Optional) Set the amount of time that the IX15 device should wait between sending keep-alive messages to the Digi Remote Manager when using a cellular interface. Allowed values are from 30 seconds to two hours. The default is 290 seconds.

   ```
   (config)> cloud drm cellular_keep_alive value
   (config)>
   ```

   where *value* is any number of hours, minutes, or seconds, and takes the format ***number*** {**h**|**m**|**s**}.

   For example, to set **the cellular keep-alive interval** to ten minutes, enter either **10m** or **600s**:

   ```
   (config)> cloud drm cellular_keep_alive 600s
   (config)>
   ```

9. Set the number of allowed keep-alive misses. Allowed values are any integer between **2** and **64**. The default is **3**.

   ```
   (config)> cloud drm keep_alive_misses integer
   (config)>
   ```

10. The **watchdog** is used to monitor the connection to remote cloud services. If the connection is down, you can configure the device to restart the connection, or to reboot. The watchdog is enabled by default. To disable:

    ```
    (config)> cloud drm watchdog false
    (config)>
    ```

11. If **watchdog** is enabled:
    a. (Optional) Set the amount of time to wait before restarting the connection to the remote cloud services, once the connection is down.

       where *value* is any number of hours, minutes, or seconds, and takes the format ***number*** {**h**|**m**|**s**}.

       For example, to set **restart_timeout** to ten minutes, enter either **10m** or **600s**:

       ```
       (config)> cloud drm restart_timeout 600s
       (config)>
       ```

       The minimum value is 30 minutes and the maximum is 48 hours. If not set, this option is disabled. The default is 30 minutes.

    b. (Optional) Set the amount of time to wait before rebooting the device, once the connection to the remote cloud servicesis down. By default, this option is not set, which means that the option is disabled.

       where *value* is any number of hours, minutes, or seconds, and takes the format ***number*** {**h**|**m**|**s**}.

       For example, to set **reboot_timeout** to ten minutes, enter either **10m** or **600s**:

```
(config)> cloud drm reboot_timeout 600s
(config)>
```

The minimum value is 30 minutes and the maximum is 48 hours. If not set, this option is disabled. The default is disabled.

12. (Optional) Determine whether to require a login and password to authenticate the user from the remote cloud services CLI:

```
(config)> cloud drm cli_local_auth true
(config)>
```

If set to **false**, no login prompt will be presented and the user will be logged in as **admin**. The default is **false**.

13. (Optional) Configure the IX15 device to communicate with remote cloud services by using SMS:
    a. **Enable** SMS messaging:

```
(config)> cloud drm sms enable true
(config)>
```

    b. Set the phone number for Digi Remote Manager:

```
(config)> cloud drm sms destination drm_phone_number
(config)>
```

    c. (Optional) Set the service identifier:

```
(config)> cloud drm sms sercice_id id
(config)>
```

1. (Optional) Configure the IX15 device to communicate with remote cloud services by using an HTTP proxy server:
    a. **Enable** the use of an HTTP proxy server:

```
(config)> cloud drm proxy enable true
(config)>
```

    b. Set the hostname of the proxy server:

```
(config)> cloud drm proxy host hostname
(config)>
```

    c. (Optional) Set the port number on the proxy server that the device should connect to. The default is 2138.

```
(config)> cloud drm proxy port integer
(config)>
```

14. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

15. Type **exit** to exit the Admin CLI.

    Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Collect device health data and set the sample interval

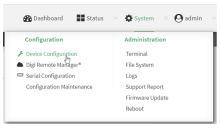
You can enable or disable the collection of device health data to upload to Digi Remote Manager, and configure the interval between health sample uploads. By default, device health data upload is enabled, and the health sample interval is set to 60 minutes.

To avoid a situation where several devices are uploading health metrics information to Remote Manager at the same time, the IX15 device includes a preconfigured randomization of two minutes for uploading metrics. For example, if **Health sample interval** is set to five minutes, the metrics will be uploaded to Remote Manager at a random time between five and seven minutes.

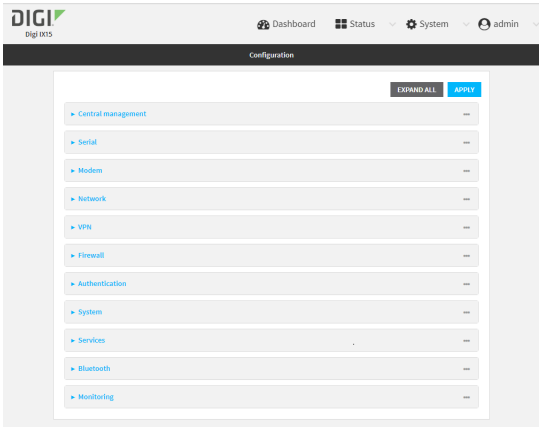
To disable the collection of device health data or enable it if it has been disabled, or to change the health sample interval:

WebUI

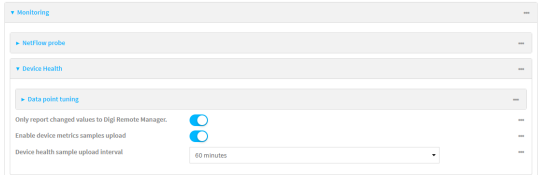
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Monitoring** > **Device Health**.



4. (Optional) Click to expand **Data point tuning**.

   Data point tuning options allow to you configure what data are uploaded to the Digi Remote Manager. All options are enabled by default.

5. **Only report changed values to Digi Remote Manager** is enabled by default.

   When enabled:

   - The device only reports device health metrics that have changed health metrics were last uploaded. This is useful to reduce the bandwidth used to report health metrics.
   - All metrics are uploaded once every hour.

   When disabled, all metrics are uploaded every **Health sample interval**.

6. Device health data upload is enabled by default. To disable, click to toggle off **Enable Device Health samples upload**.
7. For **Health sample interval**, select the interval between health sample uploads.
8. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Device health data upload is enabled by default. To enable or disable:

   - To enable:

```
(config)> monitoring devicehealth enable true
(config)>
```

   - To disable:

```
(config)> monitoring devicehealth enable false
(config)>
```

4. The interval between health sample uploads is set to 60 minutes by default. To change:

```
(config)> monitoring devicehealth interval value
(config)>
```

   where *value* is one of **1**, **5**, **15**, **30**, or **60**, and represents the number of minutes between uploads of health sample data.

5. By default, the device will only report health metrics values to Digi Remote Manager that have changed health metrics were last uploaded. This is useful to reduce the bandwidth used to

report health metrics. This is useful to reduce the bandwidth used to report health metrics. Even if enabled, all metrics are uploaded once every hour.

To disable:

```
(config)> monitoring devicehealth only_send_deltas false
(config)>
```

When disabled, all metrics are uploaded every **Health sample interval**.

6. (Optional) Tuning parameters allow to you configure what data are uploaded to the Digi Remote Manager. By default, all tuning parameters are enabled.

   To view a list of all available tuning parameters, use the **show** command:

```
(config)> show monitoring devicehealth tuning
all
       cellular
                rx
                        bytes
                                enable true
                tx
                        bytes
                                enable true
       eth
                rx
                        bytes
                                enable true
                tx
                        bytes
                                enable true
       serial
                rx
                        bytes
                                enable true
                tx
                        bytes
                                enable true
cellular
       1
                rx
                        bytes
                                enable true
                        packets
                                enable true
...
                                                    (config)>
```

   To disable a tuning parameter, set its value to false. For example, to turn off all reporting for the serial port:

```
(config)> monitoring devicehealth tuning all serial rx bytes enabled
false
```

```
(config)> monitoring devicehealth tuning all serial tx bytes enabled
false
(config)>
```

7. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

8. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

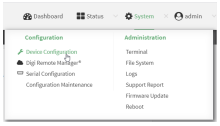
Enable event log upload to Digi Remote Manager

You can configure your device to upload the event log to Digi Remote Manager, and configure the interval between event log uploads.

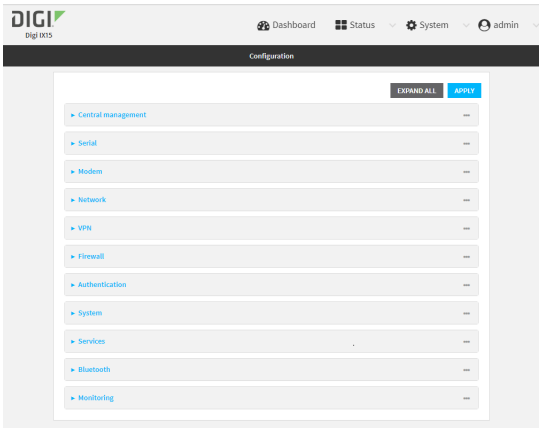
To enable the event log upload, or disable it if it has been disabled, and to change the upload interval:

WebUI

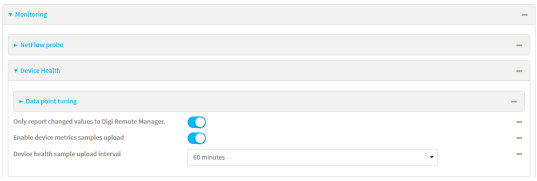
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



3. Click **Monitoring** > **Device event logs**.



4. Click **Enable event log uploads**.
5. For **Device event log upload interval**, select the interval between health sample uploads.
6. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. Device health data upload is enabled by default. To enable or disable:
   - To enable:

```
(config)> monitoring events enable true
(config)>
```

   - To disable:

```
(config)> monitoring events enable false
(config)>
```

4. The interval between event log uploads is set to 60 minutes by default. To change:

```
(config)> monitoring events interval value
(config)>
```

   where *value* is one of **1**, **5**, **15**, **30**, or **60**, and represents the number of minutes between uploads of health sample data.
5. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

6. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Log into Digi Remote Manager

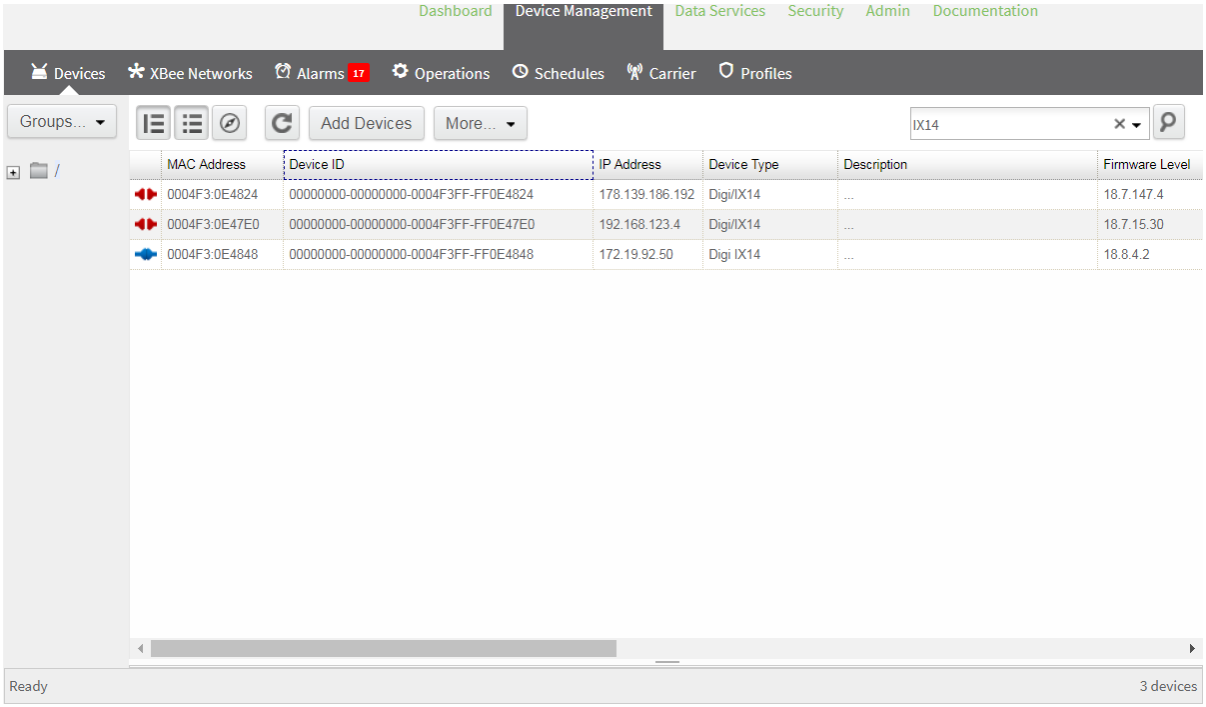
To start Digi Remote Manager

1. If you have not already done so, click here to sign up for a Digi Remote Manager account.
2. Check your email for Digi Remote Manager login instructions.
3. Go to remotemanager.digi.com.
4. Log into your Digi Remote Manager account.

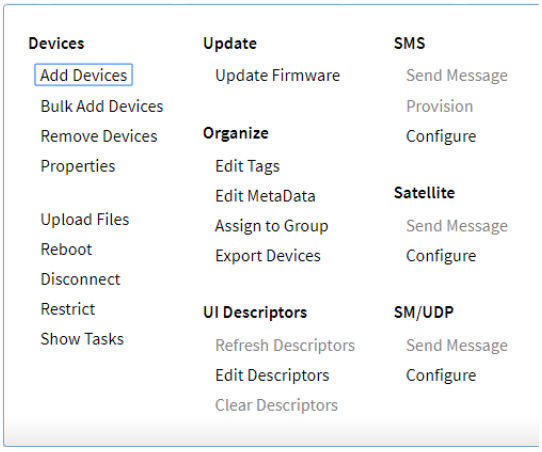
Use Digi Remote Manager to view and manage your device

To view and manage your device:

1. If you have not already done so, connect to your Digi Remote Manager account.
2. Click **Device Management** to display a list of your devices.
3. Use the Search bar to locate the device you want to manage.



4. Select the device and click **Properties** to view general information for the device.
5. Click the **More** menu to perform a task.



Add a device to Digi Remote Manager

1. If you have not already done so, connect to your Digi Remote Manager account.
2. Click **Device Management** to display a list of your devices.
3. Click **Add Devices**.
4. Select **MAC Address** and enter the Ethernet MAC address for your device.
5. For **Install Code**, enter the default password on the printed label packaged with your device. The same default password is also shown on the label affixed to the bottom of the device.
6. Click **Add**.
7. Click **OK**.

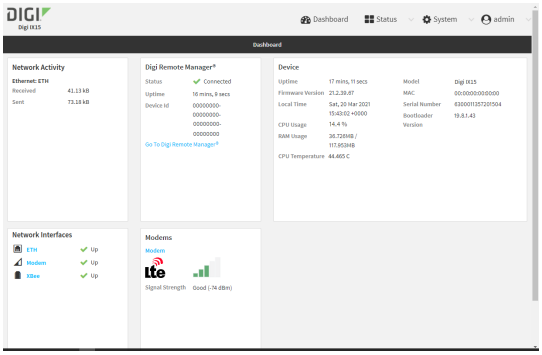
Digi Remote Manager adds your IX15 device to your account and it appears in the **Device Management** view.

View Digi Remote Manager connection status

To view the current Digi Remote Manager configuration:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. The dashboard includes a Digi Remote Manager status pane:



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. View the central management configuration:

```
(config)> show cloud
drm
       cellular_keep_alive 290s
       drm_url my.devicecloud.com
       keep_alive 60s
       keep_alive_misses 3
       retry_interval 30s
enable true
(config)>
```

1. Type **cancel** to exit configuration mode:

```
(config)> cancel
>
```

2. Type exit to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

To view the status of your device's connection to Remote Manager, use the show cloudcommand at the command line:

Command line

```
> show cloud

 Device Cloud Status
 -------------------

 Status    : Connected
 Server    : my.devicecloud.com
 Device ID : 00000000-00000000-0040FFFF-FF0F4594
>
```

The **Device ID** is the unique identifier for the device, as used by the Remote Manager.

Configure multiple devices using profiles

Digi recommends you take advantage of Digi Remote Manager profiles to manage multiple IX15 routers. Typically, if you want to provision multiple IX15 routers:

1. Using the IX15 local WebUI, configure one IX15 router to use as the model configuration for all subsequent IX15s you need to manage.
2. Register the configured IX15 device in your Digi Remote Manager account.
3. In Digi Remote Manager, create a profile based on the configured IX15.
4. Apply the profile to the IX15 devices you need to configure.

Digi Remote Manager provides multiple methods for applying profiles to registered devices. You can also include site-specific settings with a profile to override settings on a device-by-device basis.

Amazon AWS IoT

AWS IoT provides the cloud services that connect your IoT devices to other devices and AWS cloud services. AWS IoT provides device software that can help you integrate your IoT devices into AWS IoT-based solutions. If your devices can connect to AWS IoT, AWS IoT can connect them to the cloud services that AWS provides. For more information visit its web page at https://docs.aws.amazon.com/iot/latest/developerguide/what-is-aws-iot.html.

The connectivity with the AWS IoT Hub and the Digi IX15 Gateway has been validated by using the AWS IoT Device SDK: https://github.com/aws/aws-iot-device-sdk-python. Additionally, the IX15 provides several Python samples that demonstrate how to connect and interact with the AWS IoT Hub.

Note Note that the libraries are not pre-installed in the firmware by default and have to be installed, automatically by PyCharm or manually, depending on the development framework used.

To run the demos:

1. Use PyCharm and Digi's XBee plugin and run the application as usual. The environment will detect the missing libraries and will automatically install the required libraries.
2. Run the applications as any other application, see Applications. Note that in this case the libraries have to be installed manually in the device, using python pip.

Microsoft Azure

Azure IoT Hub is a managed service, hosted in the cloud, that acts as a central message hub for bi-directional communication between your IoT application and the devices it manages. For more information see docs.microsoft.com/en-us/azure/iot-hub/about-iot-hub

Connectivity with the Azure IoT Hub and the IX15 XBee Gateway has been validated by using the Azure Python SDK—github.com/Azure/azure-iot-sdks. Additionally, the IX15 provides several Python samples that demonstrate how to connect and interact with Azure.

Note The libraries are not pre-installed in the firmware by default and have to be installed automatically by PyCharm or manually, depending on the development framework used.

To run the demos:

1. Use PyCharm and Digi's XBee plugin and run the application as usual. The environment will detect the missing libraries and will automatically install the required libraries.
2. Run the applications as you would any other application, see Applications. Note that in this case the libraries have to be installed manually in the device, using python pip.

Learn more

- For information on using Digi Remote Manager to configure and manage IX15 routers, see the *Digi Remote Manager User Guide*.
- For information on using Digi Remote Manager APIs to develop custom applications, see the *Digi Remote Manager Programmer Guide*.

File system

This chapter contains the following topics:

The IX15 local file system 804
Display directory contents 804
Create a directory 805
Display file contents 806
Copy a file or directory 806
Move or rename a file or directory 807
Delete a file or directory 808
Upload and download files 809

The IX15 local file system

The IX15 local file system has approximately 150 MB of space available for storing files, such as Python programs, alternative configuration files and firmware versions, and release files, such as cellular module images. The writable directories within the filesystem are:

- /tmp
- /opt
- /etc/config

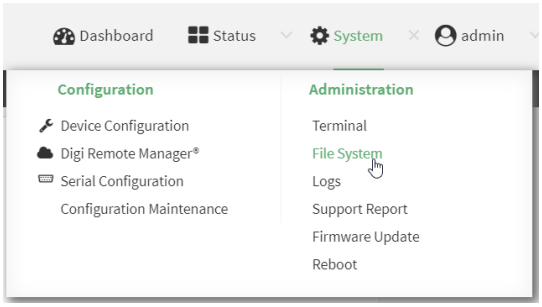
Files stored in the /tmp directory do not persist across reboots. Therefore, /tmp is a good location to upload temporary files, such as files used for firmware updates. Files stored in /opt and /etc/config do persist across reboots, but are deleted if a factory reset of the system is performed. See Erase device configuration and reset to factory defaults for more information.

Display directory contents

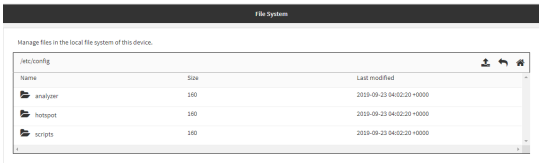
To display directory contents by using the WebUI or the Admin CLI:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Administration**, click **File System**.



The **File System** page appears.



3. Highlight a directory and click ↱ to open the directory and view the files in the directory.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the Admin CLI prompt, type **ls /*path*/*dir_name***. For example, to display the contents of the **/etc/config** directory:

```
> ls /etc/config
-rw-r--r--    1 root     root           856 Nov 20 20:12 accns.json
drw-------    2 root     root           160 Sep 23 04:02 analyzer
drwxr-xr-x    3 root     root           224 Sep 23 04:02 cc_acl
-rw-r--r--    1 root     root            47 Sep 23 04:02 dhcp.leases
...
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Create a directory

Command line

This procedure is not available through the WebUI. To make a new directory, use the mkdir command, specifying the name of the directory.

For example:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the Admin CLI prompt, type **mkdir /*path*/*dir_name***. For example, to create a directory named **temp** in **/etc/config**:

```
> mkdir /etc/config/temp
>
```

3. Verify that the directory was created:

```
> ls /etc/config
...
-rw-r--r--    1 root     root          1436 Aug 12 21:36 ssl.crt
-rw-------    1 root     root          3895 Aug 12 21:36 ssl.pem
-rw-r--r--    1 root     root            10 Aug  5 06:41 start
drwxr-xr-x    2 root     root           160 Aug 25 17:49 temp
>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Display file contents

This procedure is not available through the WebUI. To display the contents of a file by using the Admin CLI, , use the more command, specifying the name of the directory.

For example:

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the Admin CLI prompt, type **more** ***/path/filename***. For example, to view the contenct of the file **accns.json** in **/etc/config**:

```
> more /etc/config/accns.json
{
    "auth":
        "user": {
            "admin": {
                "password":
"$2a$05$W1sls1oxsadf/n4J0XT.Rgr6ewr1yerHtXQdbafsatGswKg0YUm"
            }
        }
    },
    "schema": {
        "version": "461"
    }
}
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Copy a file or directory

This procedure is not available through the WebUI. To copy a file or directory by using the Admin CLI, use the cp command, specifying the existing path and filename followed by the path and filename of the new file, or specifying the existing path and directory name followed by the path and directory name of the new directory.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the Admin CLI prompt, type **cp** ***/path/filename|dir_name /path[filename]|dir_name***. For example:

- To copy the file **/etc/config/accns.json** to a file named **backup_cfg.json** in a directory named **/etc/config/test**, enter the following:

```
> cp /etc/config/accns.json /etc/config/test/backup_cfg.json
>
```

- To copy a directory named **/etc/config/test** to **/opt**:

```
> cp /etc/config/test/ /opt/
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Move or rename a file or directory

This procedure is not available through the WebUI. To move or rename a file or directory by using the Admin CLI, use the mv command.

Command line

To rename a file named **test.py** in **/etc/config/scripts** to **final.py**:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the Admin CLI prompt, type:

```
> mv /etc/config/scripts/test.py /etc/config/scripts/final.py
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

To move **test.py** from **/etc/config/scripts** to **/opt**:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the Admin CLI prompt, type:

```
> mv /etc/config/scripts/test.py /opt/
>
```

3. Type **exit** to exit the Admin CLI.

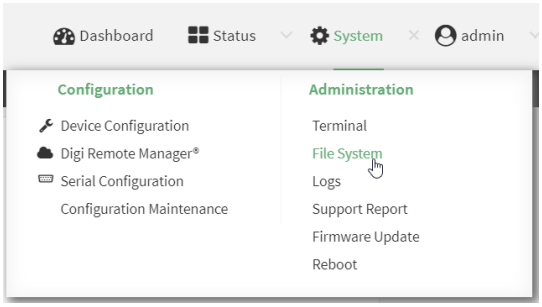
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Delete a file or directory

To delete a file or directory by using the WebUI or the Admin CLI:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Administration**, click **File System**.



The **File System** page appears.



3. Highlight the directory containing the file to be deleted and click to open the directory.
4. Highlight the file to be deleted and click .
5. Click **OK** to confirm.

Command line

To delete a file named **test.py** in **/etc/config/scripts**:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the Admin CLI prompt, type:

```
> rm /etc/config/scripts/test.py
rm: remove '/etc/config/scripts/test.py'? yes
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

To delete a directory named **temp** from **/opt**:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the Admin CLI prompt, type:

```
> rm /opt/temp/
rm: descend into directory '/opt/temp'? yes
rm: remove directory '/opt/temp'? yes
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

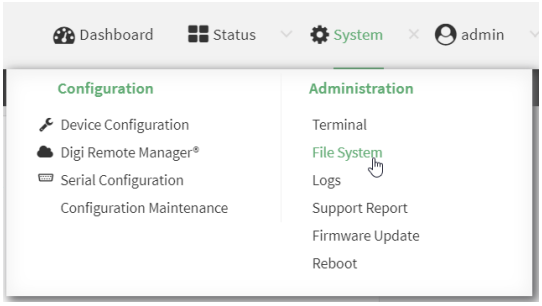
Upload and download files

You can download and upload files by using the WebUI or from the command line by using the scp Secure Copy command, or by using a utility such as SSH File Transfer Protocol (SFTP) or an SFTP application like FileZilla.

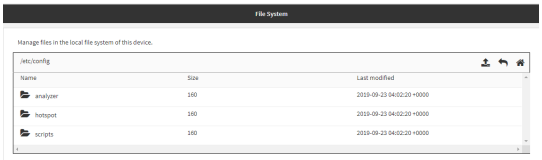
Upload and download files by using the WebUI

Upload files

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Administration**, click **File System**.



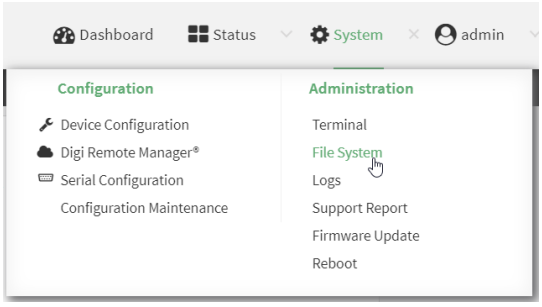
   The **File System** page appears.



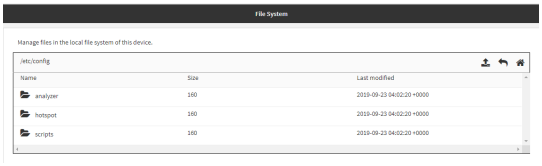
3. Highlight the directory to which the file will be uploaded and click ↩ to open the directory.
4. Click ⬆ (upload).
5. Browse to the location of the file on your local machine. Select the file and click **Open** to upload the file.

Download files

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Administration**, click **File System**.



The **File System** page appears.



3. Highlight the directory to which the file will be uploaded and click ↱ to open the directory.
4. Highlight the appropriate file and click ⭳ (download).

Upload and download files by using the Secure Copy command

Copy a file from a remote host to the IX15 device

To copy a file from a remote host to the IX15 device, use the scp command as follows:

```
> scp host hostname-or-ip user username remote remote-path local local-path to
local
```

where:

- *hostname-or-ip* is the hostname or ip address of the remote host.
- *username* is the name of the user on the remote host.
- *remote-path* is the path and filename of the file on the remote host that will be copied to the IX15 device.
- *local-path* is the location on the IX15 device where the copied file will be placed.

For example:

To copy firmware from a remote host with an IP address of 192.168.4.1 to the /etc/config directory on the IX15 device, issue the following command:

```
> scp host 192.168.4.1 user admin remote /home/admin/bin/IX15-22.2.9.85.bin
local /etc/config/scripts to local
admin@192.168.4.1's password: adminpwd
IX15-22.2.9.85.bin                        100%    36MB    11.1MB/s        00:03
>
```

Transfer a file from the IX15 device to a remote host

To copy a file from the IX15 device to a remote host, use the scp command as follows:

```
> scp host hostname-or-ip user username remote remote-path local local-path to
remote
```

where:

- *hostname-or-ip* is the hostname or ip address of the remote host.
- *username* is the name of the user on the remote host.
- *remote-path* is the location on the remote host where the file will be copied.
- *local-path* is the path and filename on the IX15 device.

For example:

To copy a support report from the IX15 device to a remote host at the IP address of 192.168.4.1:

1. Use the **system support-report** command to generate the report:

   ```
   > system support-report /var/log/
   Saving support report to /var/log/support-report-0040D0133536-22-03-03-
   10:16:23.bin
   Support report saved.
   >
   ```

2. Use the **scp** command to transfer the report to a remote host:

   ```
   > scp host 192.168.4.1 user admin remote /home/admin/temp/ local
   /var/log/support-report-00:40:D0:13:35:36-22-03-03-10:16:23.bin to remote
   admin@192.168.4.1's password: adminpwd
   support-report-0040D0133536-22-03-03-10:16:23.bin
   >
   ```

Upload and download files using SFTP

Transfer a file from a remote host to the IX15 device

This example uploads firmware from a remote host to the IX15 device with an IP address of **192.168.2.1**, using the username **ahmed**:

```
$ sftp ahmed@192.168.2.1
Password:
Connected to 192.168.2.1
sftp> put IX15-22.2.9.85
Uploading IX15-22.2.9.85 to IX15-22.2.9.85
IX15-22.2.9.85
    100%  24M  830.4KB/s   00:00
sftp> exit
$
```

Transfer a file from the IX15 device to a remote host

This example downloads a file named **test.py** from the Digi IX15 Gateway device at the IP address of **192.168.2.1** with a username of **ahmed** to the local directory on the remote host:

```
$ sftp ahmed@192.168.2.1
Password:
Connected to 192.168.2.1
sftp> get test.py
Fetching test.py to test.py
test.py
    100%  254    0.3KB/s   00:00
sftp> exit
$
```

Command line interface

This chapter contains the following topics:

Access the command line interface 814
Log in to the command line interface 814
Exit the command line interface 815
Execute a command from the web interface 815
Display help for commands and parameters 816
Auto-complete commands and parameters 818
Available commands 819
XBee-specific commands 820
Use the scp command 827
Display status and statistics using the show command 829
Device configuration using the command line interface 830
Execute configuration commands at the root Admin CLI prompt 830
Configuration mode 832
Install Python packages 844
Command line reference 846

Access the command line interface

You can access the IX15 command line interface using an SSH connection, a telnet connection, or a serial connection. You can use an open-source terminal software, such as PuTTY or TeraTerm, to access the device through one of these mechanisms.

You can also access the command line interface in the WebUI by using the **Terminal**, or the Digi Remote Manager by using the **Console**.

To access the command line, your device must be configured to allow access, and you must log in as a user who has been configured for the appropriate access. For further information about configuring access to these services, see:

- Serial: Serial port
- WebUI: Configure the web administration service
- SSH: Configure SSH access
- Telnet: Configure telnet access

Log in to the command line interface

Command line

1. Connect to the IX15 device by using a serial connection, SSH or telnet, or the **Terminal** in the WebUI or the **Console** in the Digi Remote Manager. See Access the command line interface for more information.
   - For serial connections, the default configuration is:
     - **115200** baud rate
     - **8** data bits
     - **no** parity
     - **1** stop bit
     - **no** flow control
   - For SSH and telnet connections, the default IP address of the device is**192.168.2.1** on the WAN/ETH1 .
2. At the login prompt, enter the username and password of a user with Admin access:

```
login: admin
Password: **********
```

   The default username is **admin**. The default unique password for your device is printed on the device label.
3. Depending on the device configuration, you may be presented with another menu, for example:

```
Access selection menu:

       a: Admin CLI
       s: Shell
       q: Quit
```

```
Select access or quit [admin] :
```

Type **a** or **admin** to access the IX15 command line.

You will now be connected to the Admin CLI:

```
Connecting now...
 Press Tab to autocomplete commands
 Press '?' for a list of commands and details
 Type 'help' for details on navigating the CLI
 Type 'exit' to disconnect from the Admin CLI

>
```

See Command line interface for detailed instructions on using the command line interface.

Exit the command line interface

Command line

1. At the command prompt, type **exit**.

   ```
   > exit
   ```

2. Depending on the device configuration, you may be presented with another menu, for example:

   ```
   Access selection menu:

          a: Admin CLI
          s: Shell
          q: Quit

   Select access or quit [admin] :
   ```

   Type **q** or **quit** to exit.

Execute a command from the web interface

1. Log into the IX15 WebUI as a user with Admin access.
2. At the main menu, click **Terminal**. The device console appears.

   ```
   IX15 login:
   ```

3. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

   The Admin CLI prompt appears.

   ```
   >
   ```

Display help for commands and parameters

The help command

When executed from the root command prompt, **help** displays information about autocomplete operations, how to move the cursor on the IX15 command line, and other keyboard shortcuts:

```
> help

 Commands
 ------------------------------------------------------------------------------
-
 ?              Show commands help
 <Tab>          Tab completion, displays all valid commands to complete command,
                if only one command is possible, it is used
 <Space>        Like tab except shortest prefix is used if command is valid
 <Enter>        Enter an input. If quoting then a new line is created instead. If
                the input is invalid then characters will be deleted until a
                prefix for a valid command is found.
 Ctrl + A       Move cursor to start of line
 Ctrl + E       Move cursor to end of line
 Ctrl + W       Delete word under cursor until start of line or [\',", ,\,/,.]
 Ctrl + R       If the current input is invalid then characters will be deleted
                until a prefix for a valid command is found.
 Ctrl + left    Jump cursor left until start of line or [\',", ,\,/,.]
 Ctrl + right   Jump cursor right until start of line or [\',", ,\,/,.]

>
```

The question mark (?) command

When executed from the root command prompt, **?** displays available commands:

```
> ?

 Commands
 ------------------------------------------------------------------------------
-
 config       View and modify the configuration
 exit         Exit the CLI
 analyzer     Analyzer commands.
 cp           Copy a file or directory.
 help         Show CLI editing and navigation commands.
 ls           List a directory.
 mkdir        Create a directory.
 modem        Modem commands.
 more         View a file.
 mv           Move a file or directory.
 ping         Ping a host.
 powerctrl    Power control commands.
 reboot       Reboot the system.
 rm           Remove a file or directory.
 scp          Copy a file or directory over SSH.
 show         Show instance statistics.
 system       System commands.
```

```
 traceroute  Print the route packets trace to network host.
 update      Update firmware.
 xbee        XBee commands.

>
```

Display help for individual commands

When included with a command name, both **?** and **help** provide further information about the command. For example:

1. To display further information about the **show** command, type either **show ?** or **show help**:

```
> show ?

 Commands
 ------------------------------------------------------------------------
--

 arp             Show ARP tables
 cloud           Show drm statistics
 config          Show config deltas.
 dhcp-lease      Show DHCP leases.
 dns             Show DNS servers.
 event           Show event list
 ipsec           Show IPsec statistics.
 location        Show loction information.
 log             Show syslog.
 manufacture     Show manufacturer information.
 modbus-gateway Show modbus gateway status & statistics.
 modem           Show modem statistics.
 network         Show network interface statistics.
 ntp             Show NTP information.
 openvpn         Show OpenVPN statistics.
 route           Show IP routing information.
 scripts         Show scheduled scripts.
 serial          Show serial statistics.
 surelink        Show Surelink statistics.
 system          Show system statistics.
 version         Show firmware version.
 vrrp            Show VRRP statistics.
 web-filter      Show web filter information.

 > show
```

Use the Tab key or the space bar to display abbreviated help

When executed from the root command prompt, pressing the **Tab** key or the space bar displays an abbreviated list of available commands:

Similar behavior is available with any command name:

```
> config network interface <space>
..                ...               defaultip        defaultlinklocal lan
loopback
> config network interface
```

Auto-complete commands and parameters

When entering a command and parameter, press the **Tab** key to cause the command line interface to auto-complete as much of the command and parameter as possible. Typing the space bar has similar behavior. If multiple commands are available that will match the entered text, auto-complete is not performed and the available commands are displayed instead.

Auto-complete applies to these command elements only :

- Command names. For example, typing **net<Tab>** auto-completes the command as **network**.
- Parameter names. For example:
  - **ping *hostname* int<Tab>** auto-completes the parameter as **interface**.
  - **system b<Tab>** auto-completes the parameter as **backup**.
- Parameter values, where the value is one of an enumeration or an on|off type; for example:

  ```
  (config)> serial port1 enable t<Tab>
  ```

  auto-completes to

  ```
  (config)> serial port1 enable true
  ```

Auto-complete does not function for:

- Parameter values that are string types.
- Integer values.
- File names.
- Select parameters passed to commands that perform an action.

Available commands

The following commands are available from the Admin CLI prompt:

Command	Description
config	Used to view and modify the configuration. See Device configuration using the command line interface for more information about using the **config** command.
exit	Exits the CLI.
cp	Copies a file or directory.
help	Displays: ■ CLI editing and navigation commands, when executed from the root of the Admin CLI prompt. ■ Available commands, syntax diagram, and parameter information, when executed in conjunction with another command. See Display help for commands and parameters for information about the **help** command.
ls	Lists the contents of a directory.
mkdir	Creates a directory.
modem	Executes modem commands.
more	Displays the contents of a file.
mv	Moves a file or directory.
ping	Pings a remote host using Internet Control Message Protocol (ICMP) Echo Request messages.
powerctrl	Power control commands. See Enter suspend mode for information about the **powerctrl** command.
pyinstall	Installs and manages Python packages.
reboot	Reboots the IX15 device.
rm	Removes a file.
scp	Uses the secure copy protocol (SCP) to transfer files between the IX15 device and a remote host. See Use the scp command for information about using the **scp** command.
show	Displays information about the device and the device's configuration. See Display status and statistics using the show command for more information about the show command.

Command	Description
system	Issues commands related to system functionality.
traceroute	Sends and tracks route packets to a destination host.
update	Updates the device firmware.
xbee	XBee commands. See XBee-specific commands for more information.

Note For commands that operate on the IX15's file system, such as the **cp**, **ls**, and **mkdir** commands, see File system for information about the file system, including how to copy, move and delete files and directories.

XBee-specific commands

The **xbee** command includes a set of sub-commands to work with XBee devices from the CLI.

XBee command	Description
network	XBee network commands to clear, show, and discover the network. See Manage an XBee network.
get	Reads any parameter from the local XBee—the one in the IX15—or from any remote XBee device in the network. See Read XBee parameters.
set	Writes the value of any parameter of the local or any remote XBee device. See Write XBee parameters.
execute	Executes a command in the local XBee device or any remote XBee device in the network. See Execute XBee AT commands.
update	Updates the local or any remote XBee device with a provided profile. See Apply XBee profiles.

To see all the available options, run **xbee ?**.

```
> xbee ?

 Commands
 ------------------------------------------------------------------------------
-
 execute   Execute XBee AT command.
 get       Get XBee parameter.
 network   XBee network commands.
 set       Set XBee parameter.
 update    Update XBee profile.
```

To identify an XBee device, use its 64-bit address or its node identifier (**NI**).

To refer to an XBee device, this guide uses **XBEE-ID** as its identifier—node identifier or 64-bit address.

In case any **xbee** command fails, the error is formatted as:

```
[ERROR]: ...
```

Manage an XBee network

The IX15 is designed to inspect, manage, and work with live XBee networks. The IX15 discovers the devices in the network, lists all the connected nodes, and provides information such as their 64-bit address, node identifier, role, and so on.

The IX15 network caches a list of known nodes that reflects the real XBee network. It adds new nodes to its network cache in these scenarios:

- When any kind of communication occurs between any remote node in the network and the IX15.
- When the IX15 actively searches for remote nodes in the network periodically or by customer request.

The available commands to manage an XBee network are:

Command	Description
show	Displays the list of known XBee nodes included in the network cache of the IX15. See Show the network.
discover	Explores the XBee network to find its nodes and adds them to the network cache of the IX15. See Discover the network.
clear	Eliminates all the known XBee nodes in the network cache of the IX15. See Clear the network cache.
export	Exports the XBee network to a xnet file. See Export the network.

Use **xbee network ?** to display the available options.

```
> xbee network ?

 Commands
 ------------------------------------------------------------------------------
-
 clear     XBee network clear.
 discover  XBee network discovery.
 export    Export XBee network.
 show      XBee network show.
```

Discover the network

In order to find the XBee nodes on your network, you can launch a discovery. Use the command **xbee network discover**:

- While running and when an XBee device is discovered, a line is printed showing its 64-bit address.
- When the process finishes, it shows the total amount of remote nodes discovered.

Example: xbee network discover

```
> xbee network discover
Device discovered: 0013A200DDDDDDD1
Device discovered: 0013A200DDDDDDD2
Device discovered: 0013A200DDDDDDD3

Network discover completed! 3 devices in the network.

>
```

Note See One-shot discovery for more information on how to perform a one-shot discovery from the WebUI.

Show the network

This command shows all nodes known to the IX15 and included in its network cache. To show the current nodes on your network, run **xbee network show**:

- The first node displayed is the XBee device on the IX15, the local one, followed by the remote ones.
- The information for each XBee node is displayed in columns:
  - 64-bit address—**SH** and **SL**—in hexadecimal.
  - Node identifier (**NI**).
  - Role of the node: coordinator, router or end device.
  - 16-bit address in hexadecimal.
  - Firmware version in hexadecimal.
  - Hardware name and version in hexadecimal.

Example: xbee network show

```
> xbee network show
64-bit address    Node identifier       Role          16-bit address
Firmware      Hardware
----------------  --------------------  ------------  ----------------  --------
-----  --------------------------------
0013A200417BD6ED  C1-EU8                Coordinator   00 00             1009
      XBee 3 Micro and SMT (0x41)
0013A2004195C889  COOR                  Router        29 E8             1009
      XBee 3 TH (0x42)
0013A2004195C88D  CASTOR                Router        6C 5B             1009
      XBee 3 TH (0x42)
0013A20041940C75  POLUX                 Router        C7 E9             1009
      XBee 3 Micro and SMT (0x41)

>
```

Note See Review the current XBee network state to display the list of known nodes from the WebUI.

Clear the network cache

You can clear all the network information the IX15 stores, that is, eliminate already discovered nodes and empty its network cache to start from scratch. The CLI command to clear the network cache is **xbee network clear**.

After that, the only node in the network cache is the local XBee device.

Example: xbee network clear

```
> xbee network clear

> xbee network show
64-bit address    Node identifier       Role          16-bit address
Firmware      Hardware
---------------  ---------------------  ------------  -----------------  --------
-----  -------------------------------
0013A200417BD6ED  C1-EU8                Coordinator   00 00              1009
      XBee 3 Micro and SMT (0x41)

>
```

Export the network

To export the XBee network to a file use **xbee network export** command. Use **xbee network export ?** to display its help and syntax:

```
> xbee network export ?

Syntax: export EXPORTDIR [description STRING] [name STRING]

 Parameters
 ------------------------------------------------------------------------------
-
 EXPORTDIR    Export directory path. (Required)
 description  Network description.
 name         Network name.
```

- EXPORTDIR—required—Absolute path of the directory inside the IX15 file system to leave the export file.
- description—optional: Brief description of the network.
- name—optional: Name for the network.

For example, to export the network to **/etc/config/** with the name "My IX15 network" and description "My IX15 Gateway XBee network", enter the following command:

```
> xbee network export /etc/config name "My IX15 network" description "My IX15
Gateway XBee network"
Network exported to '/etc/config/My_IX15_network_021521_130707.xnet'

>
```

Download the *.xnet file following the instructions at Upload and download files.

The exported file can be imported into the XBee Network Assistant to work offline. See Import an XBee network.

Configure individual XBee parameters

We recommend configuring way an XBee by applying a profile. You may still need to read or set some specific value, for that you can use the CLI **get**, **set**, and **execute** commands.

In the corresponding user guide you can find a complete list of AT commands, their description, and the supported value range:

- *Digi XBee 3 Zigbee RF Module User Guide*
- *Digi XBee 3 DigiMesh RF Module User Guide*
- *Digi XBee 3 802.15.4 RF Module User Guide*

Read XBee parameters

To read the value of an XBee setting use the **xbee get** command. This command allows you to get the value of an XBee parameter from the local XBee device or of any remote device in the network.

Use **xbee get ?** to display its help and syntax.

```
> xbee get ?

Gets the value of an XBee parameter.

Syntax: get XBEE-ID PARAMETER

 Parameters
 ------------------------------------------------------------------------------
-
 XBEE-ID     XBee ID (MAC or node ID). (Required)
 PARAMETER   Parameter. (Required)
```

- **XBEE-ID**—required—the XBee identifier that can be the 64-bit address or the node identifier.
- PARAMETER—required—is the XBee parameter to read.
- The returned value is:
  - A string for the node identifier (**NI**).
  - An integer value in hexadecimal format—including the **0x** prefix—for the rest of the parameters.

For example, to get the value of **D2** (AD2/DIO2 Configuration) for an XBee with the 64-bit address **0013A200DDDDDDD1**, enter the following command:

Example: xbee get

```
> xbee get 0013A200DDDDDDD1 D2
0x00

>
```

Note You can auto-complete the XBee identifier and the parameter: press the **Tab** key to cause the command line interface to auto-complete as much of the command and parameter as possible. See Auto-complete commands and parameters.

```
> xbee get 00<Tab>
> xbee get 0013A200DDDDDDD<Tab>
0013A200DDDDDDD0 0013A200DDDDDDD1 0013A200DDDDDDD2 0013A200DDDDDDD3

> xbee get 0013A200DDDDDDD2 D2
0x00

>
```

Write XBee parameters

To write the value of an XBee setting, use the command **xbee set**. This command allows you to set the value of an XBee parameter for the local XBee or of any remote XBee device in the network.

Use **xbee set ?** to display its help and syntax.

```
> xbee set ?

Sets the value of an XBee parameter. Optionally, the value can be set but not
saved, or not applied.

Syntax: set XBEE-ID PARAMETER VALUE [no-apply] [no-save]

 Parameters
 ------------------------------------------------------------------------------
-
 XBEE-ID     XBee ID (MAC or node ID). (Required)
 PARAMETER   Parameter. (Required)
 VALUE       Value. (Required)
 no-apply    no-apply
 no-save     no-save
```

- **XBEE-ID**—required—is the XBee identifier, that can be the 64-bit address or the node identifier.
- **PARAMETER**—required—is the XBee parameter to write.
- **VALUE**—required—is the new value of the provided **PARAMETER**. It is interpreted as:
  - A string if it is between single quotes.
  - An integer value without quotes. You can use prefixes to specify its format:
    - **0x** or **0X**: hexadecimal value, for example, 0x0A.
    - **0o** or **0O**: octal value, for example, 0o12.
    - **0b** or **0B**: binary value, for example, 0b1010.
    - **No prefix**: decimal value, for example, 10.
- By default, the value is applied—the new value takes effect immediately—and saved—the new value is internally programmed so it is applied the next time the XBee device boots. Optionally, you can use:
  - no-apply not to apply the value immediately and wait for a specific apply command (**AC**).
  - no-save not to store the value permanently.
- The set command does not return anything when it is successful. Otherwise, the error displays.

For example, to set to 0x01 the value of **D2** (AD2/DIO2 Configuration) of an XBee which its 64-bit address is 0013A200DDDDDDD0, enter the following command:

Example: xbee set

```
> xbee set 0013A200DDDDDDD0 D2 0x01

> xbee get 0013A200DDDDDDD0 D2
0x01

>
```

Execute XBee AT commands

To execute an AT command use **xbee execute**. This command allows you to execute an AT command in the local XBee device or in any remote device in the network.

Use **xbee execute ?** to display its help and syntax.

```
> xbee execute ?

Executes an AT command on an XBee.

Syntax: execute XBEE-ID AT-COMMAND [value STRING]

 Parameters
 ------------------------------------------------------------------------
-
 XBEE-ID     XBee ID (MAC or node ID). (Required)
 AT-COMMAND  AT command. (Required)
 value       AT command value.

>
```

- **XBEE-ID**—required—is the XBee identifier, that can be the 64-bit address or the node identifier.
- **AT-COMMAND**—required—is the AT command to execute.
- **value**—optional: Specify the AT command arguments if they are allowed.

For example, to apply all the current changes, use the **AC** AT command in an XBee with a 64-bit address of 0013A200DDDDDDD0, enter the following command:

Example: xbee execute

```
> xbee execute 0013A200DDDDDDD0 AC

>
```

Apply XBee profiles

Applying a profile is the recommended way to configure an XBee or update it. A configuration profile is an **XPRO** file that may contain several items: device firmware, setting values to configure, and a file system.

Upload XBee profiles

XPRO files are uploaded to the **/etc/config/xbee-profiles** directory of the IX15. You can upload profiles to the IX15 using:

- The WebUI. See Manage XBee profiles.
- The **scp** Secure Copy command from the CLI. See Upload and download files.
- A utility such as SSH File Transfer Protocol (SFTP) or an SFTP application like FileZilla. See Upload and download files.
- Digi Remote Manager. See Upload files to a device in the *Digi Remote Manager User Guide*.

Update an XBee

To apply a profile use the **xbee update** command. Use **xbee update ?** to display its help and syntax.

```
> xbee update ?

Updates the profile (.xpro) of an XBee.

Syntax: update XBEE-ID PROFILE

 Parameters
 ------------------------------------------------------------------------------
-
 XBEE-ID     XBee ID (MAC or node ID). (Required)
 PROFILE     Profile path and filename. (Required)

>
```

- **XBEE-ID**—required—is the XBee identifier, that can be the 64-bit address or the node identifier.
- **PROFILE**—required—is the full path of the **XPRO** file inside the IX15.

For example, to apply the **XPRO** file **/etc/config/xbee-profiles** to an XBee with 64-bit address 0013A200DDDDDDD1, enter the following command:

Example: xbee update

```
> xbee update 0013A200DDDDDDD1 /etc/config/xbee-profiles/profile_1008.xpro
Applying profile /etc/config/xbee-profiles/profile_1008.xpro (0013A200DDDDDDD1
- REMOTE1)
Reading device parameters
Updating remote XBee firmware
Updating remote XBee firmware: 1%
...
Updating remote XBee firmware: 99%
Updating remote XBee firmware: 100%
Updating XBee settings
Updating XBee settings: 10%
Updating XBee settings: 20%
...
Updating XBee settings: 90%
Updating XBee settings: 100%
Success: 100%
```

Note See Apply XBee profiles for more information on how to perform an update from the WebUI.

Use the scp command

The **scp** command uses Secure Copy Protocol (SCP) to transfer files between the IX15 device and a remote host.

Required configuration items

- The hostname or IP address of the remote host.
- The username and password of the user on the remote host.
- Whether the file is being copied to the IX15 device from a remote host, or to the remote host from the IX15 device.

- If the file is being copied to the IX15 device from a remote host:
  - The path and filename of the file on the remote host that will be copied to the IX15 device.
  - The location on the IX15 device where the file will be copied.
- If the file is being copied to a remote host from the IX15 device:
  - The path and filename of the file on the IX15 device that will be copied to the remote host.
  - The location on the remote host where the file will be copied.

Copy a file from a remote host to the IX15 device

To copy a file from a remote host to the IX15 device, use the scp command as follows:

```
> scp host hostname-or-ip user username remote remote-path local local-path to
local
```

where:

- *hostname-or-ip* is the hostname or ip address of the remote host.
- *username* is the name of the user on the remote host.
- *remote-path* is the path and filename of the file on the remote host that will be copied to the IX15 device.
- *local-path* is the location on the IX15 device where the copied file will be placed.

For example:

To copy firmware from a remote host with an IP address of 192.168.4.1 to the /etc/config directory on the IX15 device, issue the following command:

```
> scp host 192.168.4.1 user admin remote /home/admin/bin/IX15-22.2.9.85.bin
local /etc/config/scripts to local
admin@192.168.4.1's password: adminpwd
IX15-22.2.9.85.bin                      100%    36MB    11.1MB/s        00:03
>
```

Transfer a file from the IX15 device to a remote host

To copy a file from the IX15 device to a remote host, use the scp command as follows:

```
> scp host hostname-or-ip user username remote remote-path local local-path to
remote
```

where:

- *hostname-or-ip* is the hostname or ip address of the remote host.
- *username* is the name of the user on the remote host.
- *remote-path* is the location on the remote host where the file will be copied.
- *local-path* is the path and filename on the IX15 device.

For example:

To copy a support report from the IX15 device to a remote host at the IP address of 192.168.4.1:

1. Use the **system support-report** command to generate the report:

```
> system support-report /var/log/
Saving support report to /var/log/support-report-0040D0133536-22-03-03-
10:16:23.bin
Support report saved.
>
```

2. Use the **scp** command to transfer the report to a remote host:

```
> scp host 192.168.4.1 user admin remote /home/admin/temp/ local
/var/log/support-report-00:40:D0:13:35:36-22-03-03-10:16:23.bin to remote
admin@192.168.4.1's password: adminpwd
support-report-0040D0133536-22-03-03-10:16:23.bin
>
```

Display status and statistics using the show command

The IX15 **show** command display status and statistics for various features.

For example:

show config

The show config command displays all the configuration settings for the device that have been changed from the default settings. This is a particularly useful when troubleshooting the device.

```
> show config

auth tacacs+ service "login"
auth user admin password
"$2a$05$WlJQhquI7BgsytkpobKhaeLPtWraGANBcrlEaJX/wJv63JENW/HOu"
add auth user test
add auth user test group end "admin"
add auth user test group end "serial"
auth user test password
"$2a$05$RdGYz1sLKbWrqe6cZjlsd.otg03JZR6n9939XV6EYWUSP0tMAzO5W"
network interface lan ipv4 type "dhcp"
network interface lan zone "external"
network interface modem modem apn 0 apn "00000.000"
network interface modem modem apn_lock "true"
schema version "445"

>
```

show system

The show system command displays system information and statistics for the device, including CPU usage.

```
> show system

  Model                    : Digi IX15
  Serial Number            : IX15-000065
  SKU                      : IX15
```

```
  Hostname                 : IX15
  MAC Address              : DF:DD:E2:AE:21:18

  Hardware Version         : 50001947-01 1P
  Firmware Version         : 22.2.9.85
  Alt. Firmware Version    : 22.2.9.85
  Alt. Firmware Build Date : Thurs, 03 March 2022 10:16:23
  Bootloader Version       : 19.7.23.0-15f936e0ed

  Current Time             : Thurs, 03 March 2022 10:16:23 +0000
  CPU                      : 1.4%
  Uptime                   : 6 days, 6 hours, 21 minutes, 57 seconds (541317s)
  Temperature              : 40C

>
```

show network

The show network command displays status and statistics for network interfaces.

```
> show network

 Interface          Proto  Status   Address
 -----------------  -----  -------  --------------------------------
 defaultip          IPv4   up       192.168.210.1/24
 defaultlinklocal   IPv4   up       169.254.100.100/16
 lan                IPv4   up       192.168.2.1
 lan                IPv6   up       0:0:0:0:0:ffff:c0a8:301
 loopback           IPv4   up       127.0.0.1/8
 wan                IPv4   up       192.168.3.1/24
 wan                IPv6   up       fd00:2704::240:ffff:fe80:120/64

>
```

Device configuration using the command line interface

The **config** command allows for device configuration from the command line. All configuration tasks that can be performed by using the WebUI can also be performed by using the **config** command.

There are two ways to invoke the **config** command from the CLI:

- Execute the **config** command and parameters at the root prompt. See Execute configuration commands at the root Admin CLI prompt for more information.
- Enter configuration mode by executing the **config** command without any parameters. See Configuration mode for more information.

Execute configuration commands at the root Admin CLI prompt

You can execute the **config** command at the root Admin CLI prompt with any appropriate parameters. When the **config** command is used in this way, changes to the device's configuration are automatically saved when the command is executed.

For example, to disable the SSH service from the root prompt, enter the following command:

```
> config service ssh enable false
>
```

The IX15 device's ssh service is now disabled.

Note When the **config** command is executed at the root prompt, certain configuration actions that are available in configuration mode cannot be performed. This includes validating configuration changes, canceling and reverting configuration changes, and performing actions on elements in lists. See Configuration mode for information about using configuration mode.

Display help for the config command from the root Admin CLI prompt

Display additional configuration commands, as well as available parameters and values, by entering the question mark (?) character after the **config** command.

1. For example:

```
> config ?
```

Will display the following help information:

```
> config ?

Additional Configuration
------------------------------------------------------------------------
-
application               Custom scripts
auth                      Authentication
cloud                     Central management
firewall                  Firewall
monitoring                Monitoring
network                   Network
serial                    Serial
service                   Services
system                    System
vpn                       VPN

Run "config" with no arguments to enter the configuration editing mode.

> config
```

2. You can then display help for the additional configuration commands. For example, to display help for the **config service** command:

```
> config service ?
Services

Additional Configuration
------------------------------------------------------------------------
-
dns                       DNS
mdns                      Service Discovery (mDNS)
multicast                 Multicast
ntp                       NTP
```

```
remote_control            Remote control
snmp                      SNMP
ssh                       SSH
telnet                    Telnet
web_admin                 Web administration

> config service
```

3. Next, display help for the **config service ssh** command:

```
> config service ssh ?

SSH: An SSH server for managing the device.

Parameters                Current Value
--------------------------------------------------------------------------
-
enable                    true           Enable
key                       [private]      Private key
port                      22             Port

Additional Configuration
--------------------------------------------------------------------------
-
acl                       Access control list
mdns

> config service ssh
```

4. Lastly, display the allowed values and other information for the **enable** parameter:

```
> config service ssh enable ?

Enable: Enable the service.
Format: true, false, yes, no, 1, 0
Default value: true
Current value: true

> config service ssh enable
```

Configuration mode

Configuration mode allows you to perform multiple configuration tasks and validate the changes prior to saving them. You can cancel all changes without saving them at any time. Configuration changes do not take effect until the configuration is saved.

Enable configuration mode

To enable configuration mode, at the root prompt, enter the **config** command without any parameters:

```
> config
(config)>
```

When the command line is in configuration mode, the prompt will change to include **(config)**, to indicate that you are currently in configuration mode.

Enter configuration commands in configuration mode

There are two ways to enter configuration commands while in configuration mode:

- Enter the full command string from the config prompt.

  For example, to disable the ssh service by entering the full command string at the config prompt:

  ```
  (config)> service ssh enable false
  (config)>
  ```

- Execute commands by moving through the configuration schema.

  For example, to disable the ssh service by moving through the configuration and then executing the **enable false** command:

  1. At the **config** prompt, enter **service** to move to the **service** node:

     ```
     (config)> service
     (config service)>
     ```

  2. Enter **ssh** to move to the **ssh** node:

     ```
     (config service)> ssh
     (config service ssh)>
     ```

  3. Enter **enable false** to disable the **ssh** service:

     ```
     (config service ssh)> enable false
     (config service ssh)>
     ```

  See Move within the configuration schema for more information about moving within the configuration.

Save changes and exit configuration mode

To save changes that you have made to the configuration while in configuration mode, use **save**. The save command automatically validates the configuration changes; the configuration will not be saved if it is not valid. Note that you can also validate configuration changes at any time while in configuration mode by using the **validate** command.

```
(config)> save
Configuration saved.
>
```

After using **save** to save changes to the configuration, you will automatically exit configuration mode. To return to configuration mode, type **config** again.

Exit configuration mode without saving changes

You can discard any unsaved configuration changes and exit configuration mode by using the **cancel** command:

```
(config)> cancel
>
```

After using **cancel** to discard unsaved changes to the configuration, you will automatically exit configuration mode.

Configuration actions

In configuration mode, configuration actions are available to perform tasks related to saving or canceling the configuration changes, and to manage items and elements in lists. The commands can be listed by entering a question mark (**?**) at the **config** prompt.

The following actions are available:

Configuration actions	Description
cancel	Discards unsaved configuration changes and exits configuration mode.
save	Saves configuration changes and exits configuration mode.
validate	Validates configuration changes.
revert	Reverts the configuration to default settings. See The revert command for more information.
show	Displays configuration settings.
add	Adds a named element, or an element in a list. See Manage elements in lists for information about using the **add** command with lists.
del	Deletes a named element, or an element in a list. See Manage elements in lists for information about using the **del** command with lists.
move	Moves elements in a list. See Manage elements in lists for information about using the **move** command with lists.

Display command line help in configuration mode

Display additional configuration commands, as well as available parameters and values, by entering the question mark (?) character at the **config** prompt. For example:

1. Enter **?** at the **config** prompt:

```
(config)> ?
```

This will display the following help information:

```
(config)> ?

 Additional Configuration
 ------------------------------------------------------------------------
--
 application               Custom scripts
 auth                      Authentication
 cloud                     Central management
 firewall                  Firewall
 monitoring                Monitoring
 network                   Network
 serial                    Serial
 service                   Services
 system                    System
 vpn                       VPN

(config)>
```

2. You can then display help for the additional configuration commands. For example, to display help for the **config service** command, use one of the following methods:
   - At the **config** prompt, enter **service ?**:

     ```
     (config)> service ?
     ```

   - At the **config** prompt:
     a. Enter **service** to move to the **service** node:

        ```
        (config)> service
        (config service)>
        ```

     b. Enter **?** to display help for the **service** node:

        ```
        (config service)> ?
        ```

Either of these methods will display the following information:

```
config> service ?

Services

 Additional Configuration
 ------------------------------------------------------------------------
--
 dns                       DNS
 mdns                      Service Discovery (mDNS)
 multicast                 Multicast
```

```
 ntp                        NTP
 remote_control             Remote control
 snmp                       SNMP
 ssh                        SSH
 telnet                     Telnet
 web_admin                  Web administration

(config)> service
```

3. Next, to display help for the **service ssh** command, use one of the following methods:
   - At the **config** prompt, enter **service ssh ?**:

     ```
     (config)> service ssh ?
     ```

   - At the **config** prompt:
     a. Enter **service** to move to the **service** node:

        ```
        (config)> service
        (config service)>
        ```

     b. Enter **ssh** to move to the **ssh** node:

        ```
        (config service)> ssh
        (config service ssh)>
        ```

     c. Enter **?** to display help for the **ssh** node:

        ```
        (config service ssh)> ?
        ```

   Either of these methods will display the following information:

```
(config)> service ssh ?

SSH: An SSH server for managing the device.

 Parameters                 Current Value
 ------------------------------------------------------------------------
--
 enable                     true           Enable
 key                        [private]      Private key
 port                       22             Port

 Additional Configuration
 ------------------------------------------------------------------------
--
 acl                        Access control list
 mdns

(config)> service ssh
```

4. Lastly, to display allowed values and other information for the **enable** parameter, use one of the following methods:

- At the **config** prompt, enter **service ssh enable ?**:

```
(config)> service ssh enable ?
```

- At the **config** prompt:
  a. Enter **service** to move to the **service** node:

```
(config)> service
(config service)>
```

  b. Enter **ssh** to move to the **ssh** node:

```
(config service)> ssh
(config service ssh)>
```

  c. Enter **enable ?** to display help for the **enable** parameter:

```
(config service ssh)> enable ?
(config service ssh)>
```

Either of these methods will display the following information:

```
(config)> service ssh enable ?

 Enable: Enable the service.
 Format: true, false, yes, no, 1, 0
 Default value: true
 Current value: true

(config)> service ssh enable
```

Move within the configuration schema

You can perform configuration tasks at the CLI by moving within the configuration.

- Move forward one node in the configuration by entering the name of an Additional Configuration option:
  1. At the **config** prompt, type **service** to move to the **service** node:

```
(config)> service
(config service)>
```

  2. Type **ssh** to move to the **ssh** node:

```
(config service)> ssh
(config service ssh)>
```

  3. Type **acl** to move to the **acl** node:

```
(config service ssh)> acl
(config service ssh acl)>
```

4. Type **zone** to move to the **zone** node:

```
(config service ssh acl)> zone
(config service ssh acl zone)>
```

You can also enter multiple nodes at once to move multiple steps in the configuration:

```
(config)> service ssh acl zone
(config service ssh acl zone)>
```

- Move backward one node in the configuration by entering two periods (**..**):

```
(config service ssh acl zone)> ..
(config service ssh acl)>
```

You can also move back multiples nodes in the configuration by typing multiple sets of two periods:

```
(config service ssh acl zone)> .. .. ..
(config service)>
```

- Move to the root of the config prompt from anywhere within the configuration by entering three periods (**...**):

```
(config service ssh acl zone)> ...
(config)>
```

Manage elements in lists

While in configuration mode, you can use the **add**, **del**, and **move** action commands to manage elements in a list. When working with lists, these actions require an index number to identify the list item that will be acted on.

Add elements to a list

When used with parameters that contains lists of elements, the **add** command is used to add an element to the list.

For example, to add an authentication method:

1. Display current authentication method by using the **show** command:

```
(config)> show auth method
0 local
(config)>
```

2. Add an authentication method by using the **add *index_item*** command. For example:
   - To add the TACACS+ authentication method to the beginning of the list, use the index number **0**:

```
(config)> add auth method 0 tacacs+
(config)> show auth method
0 tacacs+
```

```
1 local
(config)>
```

- To add the TACACS+ authentication method to the end of the list, use the **end** keyword:

```
(config)> add auth method end tacacs+
(config)> show auth method
0 local
1 tacacs+
(config)>
```

The end keyword

As demonstrated above, the **end** keyword is used to add an element to the end of a list. Additionally, the **end** keyword is used to add an element to a list that does not have any elements.

For example, to add an authentication group to a user that has just been created:

1. Use the **show** command to verify that the user is not currently a member of any groups:

```
(config)> show auth user new-user group
(config)>
```

2. Use the **end** keyword to add the admin group to the user's configuration:

```
(config)> add auth user new-user group end admin
(config)>
```

3. Use the **show** command again to verify that the admin group has been added to the user's configuration:

```
(config)> show auth user new-user group
0 admin
(config)>
```

Delete elements from a list

When used with parameters that contains lists of elements, the **del** command is used to delete an element in the list.

For example, to delete an authentication method:

1. Use the **show** command to display current authentication method configuration:

```
(config)> show auth method
0 local
1 tacacs+
2 radius
(config)>
```

2. Delete one of the authentication methods by using the **del *index_number*** command. For example:
   a. To delete the local authentication method, use the index number **0**:

```
(config)> del auth method 0
(config)>
```

b. Use the **show** command to verify that the local authentication method was removed:

```
(config)> show auth method
0 tacacs+
1 radius
(config)>
```

Move elements within a list

Use the **move** command to reorder elements in a list.

For example, to reorder the authentication methods:

1. Use the **show** command to display current authentication method configuration:

```
(config)> show auth method
0 local
1 tacacs+
2 radius
(config)>
```

2. To configure the device to use TACACS+ authentication first to authenticate a user, use the **move *index_number_1 index_number_2*** command:

```
(config)> move auth method 1 0
(config)>
```

3. Use the **show** command again to verify the change:

```
(config)> show auth method
0 tacacs+
1 local
2 radius
(config)>
```

The revert command

The **revert** command is used to revert changes to the IX15 device's configuration and restore default configuration settings. The behavior of the revert command varies depending on where in the configuration hierarchy the command is executed, and whether the optional **path** parameter is used. After executing the revert command, you must save the configuration changes by using the **save** command. You can also discard the configuration changes by using the **cancel** command.



CAUTION! The **revert** command reverts all changes to the default configuration, not only unsaved changes.

Revert all configuration changes to default settings

To discard all configuration changes and revert to default settings, use the **revert** command at the config prompt without the optional **path** parameter:

1. At the config prompt, enter **revert**:

```
(config)> revert
(config)>
```

2. Set the password for the admin user prior to saving the changes:

```
(config)> auth user admin password pwd
(config)>
```

3. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Revert a subset of configuration changes to the default settings

There are two methods to revert a subset of configuration changes to the default settings.

- Enter the **revert** command with the **path** parameter. For example, to revert all changes to the authentication methods configuration:
  1. Enter the **revert** command with the **path** set to **auth method**:

```
(config)> revert auth method
(config)>
```

  2. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

  3. Type **exit** to exit the Admin CLI.

     Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.
- Move to the location in the configuration and enter the **revert** command without the **path** parameter. For example:
  1. Change to the auth method node:

```
(config)> auth method
(config auth method)>
```

  2. Enter the **revert** command:

```
(config auth method)> revert
(config auth method)>
```

3. Save the configuration and apply the change:

```
(config auth method)> save
Configuration saved.
>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

- You can also use a combination of both of these methods:

1. Change to the **auth** node:

```
(config)> auth
(config auth)>
```

2. Enter the **revert** command with the **path** set to **method**:

```
(config auth)> revert method
(config auth)>
```

3. Save the configuration and apply the change:

```
(config auth)> save
Configuration saved.
>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Enter strings in configuration commands

For string parameters, if the string value contains a space, the value must be enclosed in quotation marks. For example, to assign a descriptive name for the device using the **system** command, enter:

```
(config)> system description "Digi IX15"
```

Example: Create a new user by using the command line

In this example, you will use the IX15 command line to create a new user, provide a password for the user, and assign the user to authentication groups.

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. At the config prompt, create a new user with the username **user1**:
   - Method one: Create a user at the root of the config prompt:

     ```
     (config)> add auth user user1
     (config auth user user1)>
     ```

   - Method two: Create a user by moving through the configuration:
     a. At the config prompt, enter **auth** to move to the **auth** node:

        ```
        (config)> auth
        (config auth)>
        ```

     b. Enter **user** to move to the **user** node:

        ```
        (config auth)> user
        (config auth user)>
        ```

     c. Create a new user with the username **user1**:

        ```
        (config auth user)> add user1
        (config auth user user1)>
        ```

4. Configure a password for the user:

   ```
   (config auth user user1)> password pwd1
   (config auth user user1)>
   ```

5. List available authentication groups:

   ```
   (config auth user user1)> show .. .. group

   admin
       acl
           admin
               enable true
           nagios
               enable false
           openvpn
               enable false
               no tunnels
           portal
               enable false
               no portals
           serial
               enable false
               no ports
           shell
               enable false

   serial
       acl
           admin
   ```

```
            enable true
        nagios
            enable false
        openvpn
            enable false
            no tunnels
        portal
            enable false
            no portals
        serial
            enable true
                ports
                    0 port1
        shell
            enable false
(config auth user user1)>
```

6. Add the user to the admin group:

```
(config auth user user1)> add group end admin
(config auth user user1)>
```

7. Save the configuration and apply the change:

```
(config auth user user1)> save
Configuration saved.
>
```

8. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Install Python packages

The IX15 device includes support to install Python wheel packages or zip packages coming from the PyCharm IDE. See Develop Python applications for details on how to develop Python applications.

When installing a zip package, the contents are uncompressed in the **/etc/config/scripts** directory and the requirements are automatically installed.

To install a wheel or zip—see Develop an application in PyCharm—follow these steps:

1. Upload your package to a directory in your device. See Upload and download files.
2. Access the command line application and run **pyinstall package <path_to_your_package>**.

Example: pyinstall package

```
> pyinstall package /tmp/AWS-basicPubSub.zip
Archive:  /tmp/AWS-basicPubSub.zip
inflating: AWS-basicPubSub/main.py
inflating: AWS-basicPubSub/requirements.txt
Defaulting to user installation because normal site-packages is not
writeable
Collecting AWSIoTPythonSDK>=1.4.9
|████████████████████████████████| 80 kB 2.0 MB/s
```

```
Preparing metadata (setup.py) ... done
Using legacy 'setup.py install' for AWSIoTPythonSDK, since package 'wheel'
is not installed.
Installing collected packages: AWSIoTPythonSDK
Running setup.py install for AWSIoTPythonSDK ... done
Successfully installed AWSIoTPythonSDK-1.4.9
>save configuration
cfg.commit()
```

File and requirements are installed in **/etc/config/scripts** directory.

```
> ls /etc/config/scripts/AWS-basicPubSub/
-rw-r-----    1 root     root          5091 Nov  2 06:45 main.py
-rw-r-----    1 root     root           103 Nov  2 06:45 requirements.txt
>
```

3. Optionally, you can configure your IX15 device to automatically run the installed application.

See Set up the IX15 to automatically run your applications for more details.

Command line reference

analyzer clear 847
analyzer save 848
analyzer start 848
analyzer stop 848
clear dhcp-lease ip-address 848
clear dhcp-lease mac 849
container create 849
container delete 849
cp 849
help 850
ls 851
mkdir 852
modem at 852
modem at-interactive 852
modem firmware check 852
modem firmware list 852
modem firmware ota check 853
modem firmware ota list 853
modem firmware ota update 853
modem firmware update 853
modem pin change 854
modem pin disable 854
modem pin enable 854
modem pin status 855
modem pin unlock 855
modem puk status 855
modem puk unlock 855
modem reset 856
modem scan 856
modem sim-slot 856
monitoring 856
monitoring metrics upload 857
more 857
mv 857
ping 857
powerctrl 858
pyinstall 858
reboot 859
rm 860
scp 861
show analyzer 861
show arp 861
show cloud 861
show config 862
show containers 862
show dhcp-lease 862
show dns 862
show event 862
show hotspot 863
show ipsec 863
show l2tp lac 863
show l2tp lns 863

show l2tpeth 864
show location 864
show log 864
show manufacture 864
show modbus-gateway 864
show modem 865
show nemo 865
show network 865
show ntp 865
show openvpn client 866
show openvpn server 866
show route 866
show serial 866
show scripts 867
show surelink interface 867
show surelink ipsec 867
show surelink openvpn 867
show system 868
show usb 868
show version 868
show vrrp 868
show web-filter 868
speedtest 869
ssh 869
system backup 869
system disable-cryptography 870
system duplicate-firmware 870
system factory-erase 870
system find-me 870
system firmware ota check 871
system firmware ota list 871
system firmware ota update 871
system firmware update 871
system power ignition off_delay 871
system restore 872
system script start 872
system script stop 872
system serial clear 872
system serial save 873
system serial show 873
system serial start 873
system serial stop 873
system support-report 873
system time set 874
system time sync 874
system time test 874
telnet 874
traceroute 875
xbee 875

analyzer clear

Clears the traffic captured by the analyzer.

Syntax

```
analyzer clear <name>
```

Parameters

name: Name of the capture filter to use.

analyzer save

Saves the current captured traffic to a file.

Syntax

```
analyzer save <name> <filename>
```

Parameters

name: Name of the capture filter to use.

filename: The filename to save captured traffic to. The file will be saved to the device's /etc/config/analyzer directory.

analyzer start

Start a capture session of packets on this devices interfaces.

Syntax

```
analyzer start <name>
```

Parameters

name: Name of the capture filter to use.

analyzer stop

Stops the traffic capture session.

Syntax

```
analyzer stop <name>
```

Parameters

name: Name of the capture filter to use.

clear dhcp-lease ip-address

Clear the DHCP lease for the specified IP address.

Syntax

```
clear dhcp-lease ip-address ADDRESS
```

Parameters

address: An IPv4 or IPv6 address

clear dhcp-lease mac

Clear the DHCP lease for the specified MAC address.

Syntax

```
clear dhcp-lease mac ADDRESS
```

Parameters

address: 12-digit, colon-delimited MAC address [00:11:22:AA:BB:CC]

container create

Create a LXC container from a given image. This process creates a copy of the image, so the orginal image may be deleted after creating the container without breaking the container.

Syntax

```
container create <path>
```

Parameters

path: Filepath for container image to be created. .

container delete

Delete a LXC container. This will remove the LXC container configuration and the container image.

Syntax

```
container delete <container>
```

Parameters

container: Filepath for container image to be created. This process creates a copy of the image, so the orginal image may be deleted after creating the container without breaking the container.

cp

Copy a file or directory.

Syntax

```
cp <source> <destination> [force]
```

Parameters

source: The source file or directory to copy.

destination: The destination path to copy the source file or directory to.

force: Do not ask to overwrite the destination file if it exists.

help

Show CLI editing and navigation commands.

Syntax

```
help
```

Parameters

None

ls

List a directory.

Syntax

```
ls <path> [show-hidden]
```

Parameters

path: List files and directories under this path.

show-hidden: Show hidden files and directories. Hidden filenames begin with '.'.

mkdir

Create a directory. Parent directories are created as needed.

Syntax

```
mkdir <path>
```

Parameters

path: The directory path to create.

modem at

Send an AT command to the modem and display the response.

Syntax

```
modem at <cmd> [name STRING] [imei STRING]
```

Parameters

cmd: The AT command string.

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem at-interactive

Start an AT command session on the modem's AT serial port.

Syntax

```
modem at-interactive [name STRING] [imei STRING]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem firmware check

Inspect /opt/[MODEM_MODEL]/Custom_Firmware/ directory for new modem firmware file.

Syntax

```
modem firmware check [name STRING] [imei STRING]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem firmware list

List modem firmware files found in the /opt/[MODEM_MODEL]/ directory.

Syntax

```
modem firmware list [name STRING] [imei STRING]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem firmware ota check

Query the Digi firmware server for the latest remote modem firmware version.

Syntax

```
modem firmware ota check [name STRING] [imei STRING]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem firmware ota list

Query the Digi firmware server for a list of modem firmware versions.

Syntax

```
modem firmware ota list [name STRING] [imei STRING]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem firmware ota update

Perform FOTA (firmware-over-the-air) update. The modem will be updated to the latest modem firmware image unless a specific firmware version is specified.

Syntax

```
modem firmware ota update [name STRING] [imei STRING] [version STRING]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

version: Firmware version name.

modem firmware update

Update modem firmware using local firmware file. The modem will be updated to the firmware specified in the /opt/[MODEM_MODEL]/Custom_Firmware/ directory unless a specific firmware version is specified.

Syntax

```
modem firmware update [name STRING] [imei STRING] [version STRING]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

version: Firmware version name.

modem pin change

Change the SIM's PIN code.

Warning: Attempting to use an incorrect PIN code may PUK lock the SIM.

Syntax

```
modem pin change <old-pin> <new-pin> [name STRING] [imei STRING]
```

Parameters

old-pin: The SIM's PIN code.

new-pin: The PIN code to change to.

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem pin disable

Disable the PIN lock on the SIM card that is active in the modem.

Warning: Attempting to use an incorrect PIN code may PUK lock the SIM.

Syntax

```
modem pin disable <pin> [name STRING] [imei STRING]
```

Parameters

pin: The SIM's PIN code.

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem pin enable

Enable the PIN lock on the SIM card that is active in the modem. The SIM card will need to be unlocked before each use.

Warning: Attempting to use an incorrect PIN code may PUK lock the SIM.

Syntax

```
modem pin enable <pin> [name STRING] [imei STRING]
```

Parameters

pin: The SIM's PIN code.

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem pin status

Print the PIN lock status and the number of PIN enable/disable/unlock attempts remaining. The SIM will be PUK locked when there are no remaining retries.

Syntax

```
modem pin status [name STRING] [imei STRING]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem pin unlock

Temporarily unlock the SIM card with a PIN code. Set the PIN field in the modem interface's configuration to unlock the SIM card automatically before use.

Warning: Attempting to use an incorrect PIN code may PUK lock the SIM.

Syntax

```
modem pin unlock <pin> [name STRING] [imei STRING]
```

Parameters

pin: The SIM's PIN code.

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem puk status

Print the PUK status and the number of PUK unlock attempts remaining.

Syntax

```
modem puk status [name STRING] [imei STRING]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem puk unlock

Unlock the SIM with a PUK code from the SIM provider.

Syntax

```
modem puk unlock <puk> <new-pin> [name STRING] [imei STRING]
```

Parameters

puk: The SIM's PUK code.

new-pin: The PIN code to change to.

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem reset

Reset the modem hardware (reboot it). This can be useful if the modem has stopped responding to the network or is behaving inconsistently.

Syntax

```
modem reset [name STRING] [imei STRING]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem scan

List of carriers present in the network.

Syntax

```
modem scan [name STRING] [imei STRING]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

modem sim-slot

Show or change the modem's active SIM slot. This applies only to modems with multiple SIM slots.

Syntax

```
modem sim-slot <slot> [name STRING] [imei STRING]
```

Parameters

slot: The SIM slot to change to.

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

monitoring

Commands to clear the device's status or systems.

monitoring metrics

Device metrics commands.

uplaod

Immediately upload current device health metrics. Functions as if a scheduled upload was triggered.

Parameters

None

monitoring metrics upload

Immediately upload current device health metrics. Functions as if a scheduled upload was triggered.

Syntax

```
monitoring metrics upload
```

Parameters

None

more

View a file.

Syntax

```
more <path>
```

Parameters

path: The file to view.

mv

Move a file or directory.

Syntax

```
mv <source> <destination> [force]
```

Parameters

source: The source file or directory to move.
destination: The destination path to move the source file or directory to.
force: Do not ask to overwrite the destination file if it exists.

ping

Ping a host using ICMP echo.

Syntax

```
ping <host> [interface STRING] [source STRING] [ipv6] [size INTEGER] [count
INTEGER] [broadcast]
```

Parameters

host: The name or address of the remote host to send ICMP ping requests to. If broadcast is enabled, can be the broadcast address.

interface: The network interface to send ping packets from when the host is reachable over a default route. If not specified, the system's primary default route will be used.

source: The ping command will send a packet with the source address set to the IP address of this interface, rather than the address of the interface the packet is sent from.

ipv6: If a hostname is defined as the value of the 'host' parameter, use the hosts IPV6 address.

size: The number of bytes sent in the ICMP ping request. (Minimum: 0, Default: 56)

count: The number of ICMP ping requests to send before terminating. (Minimum: 1, Default: 100)

broadcast: Enable broadcast ping functionality.

powerctrl

Power control commands.

powerctrl state poweroff

Enter in poweroff or suspend.

powerctrl state suspend

Enter in suspend.

pyinstall

Pyinstall commands

pyinstall package STRING

Install a Python package.

Parameters

STRING

Absolute path of the .zip or .whl file.

Syntax: **STRING**

reboot

Reboot the system.

Parameters

None

rm

Remove a file or directory.

Syntax

```
rm <path> [force]
```

Parameters

path: The path to remove.

force: Force the file to be removed without asking.

scp

Copy a file or directory over SSH.

Syntax

```
scp <local> <remote> <host> <user> <to> [port INTEGER]
```

Parameters

local: The path and name of the file on the local device to copy to or from.

remote: The path and name of the file on the remote host to copy to or from.

host: The hostname or IP address of the remote host.

user: The username to use when connecting to the remote host.

to: Determine whether to copy the file from the local device to the remote host, or from the remote host to the local device.

port: The SSH port to use to connect to the remote host. (Minimum: 1, Maximum: 65535, Default: 22)

show analyzer

Show packets from a specified analyzer capture.

Syntax

```
show analyzer <name>
```

Parameters

name: Name of the capture filter to use.

show arp

Show ARP tables. If no IP version is specified IPv4 & IPV6 will be displayed.

Syntax

```
show arp [ipv4] [ipv6] [verbose]
```

Parameters

ipv4: Display IPv4 routes. If no IP version is specified IPv4 & IPV6 will be displayed.

ipv6: Display IPv6 routes. If no IP version is specified IPv4 & IPV6 will be displayed.

verbose: Display more information (less concise, more detail).

show cloud

Show drm status & statistics.

Syntax

```
show cloud
```

Parameters

None

show config

Show a summary of changes made to the default configuration. The changes shown are not suitable for pasting into a CLI session.

Syntax

```
show config [cli_format]
```

Parameters

cli_format: Show the exact CLI commands required to configure the device from a default configuration. The changes shown are suitable for pasting into a CLI session, although individual output lines maybe context sensitive and unable to be entered in isolation.

show containers

Show container status & statistics.

Syntax

```
show containers [container STRING]
```

Parameters

container: Display more details and config data for a specific container.

show dhcp-lease

Show DHCP leases.

Syntax

```
show dhcp-lease [all] [verbose]
```

Parameters

all: Show all leases (active and inactive (not in etc/config/dhcp.*lease)).

verbose: Display more information (less concise, more detail).

show dns

Show DNS servers and associated domains.

Syntax

```
show dns
```

Parameters

None

show event

Show event list (high level).

Syntax

```
show event [table <status|error|info>] [number INTEGER]
```

Parameters
table: Type of event log to be displayed (status, error, info).
number: Number of lines to retrieve from log. (Minimum: 1, Default: 20)

show hotspot

Show hotspot statistics.

Syntax

```
show hotspot [name STRING] [ip STRING]
```

Parameters
name: The configured instance name of the hotspot.
ip: IP address of a specific client, to limit the status display to only this client.

show ipsec

Show IPsec status & statistics.

Syntax

```
show ipsec [tunnel STRING] [all] [verbose]
```

Parameters
tunnel: Display more details and config data for a specific IPsec tunnel.
all: Display all tunnels including disabled tunnels.
verbose: Display status of one or all tunnels in plain text.

show l2tp lac

Show L2TP access concentrator status & statistics.

Syntax

```
show l2tp lac [name STRING]
```

Parameters
name: Display more details for a specific L2TP access concentrator.

show l2tp lns

Show L2TP network server status & statistics.

Syntax

```
show l2tp lns [name STRING]
```

Parameters

name: Display more details for a specific L2TP network server.

show l2tpeth

Show L2TPv3 ethernet tunnel session status and statistics.

Syntax

```
show l2tpeth [name STRING]
```

Parameters

name: Display more details for a specific L2TPv3 ethernet tunnel session.

show location

Show location information.

Syntax

```
show location [geofence]
```

Parameters

geofence: Show geofence information.

show log

Show system log (low level).

Syntax

```
show log [number INTEGER] [filter <critical|warning|debug|info>]
```

Parameters

number: Number of lines to retrieve from log. (Minimum: 1, Default: 20)

filter: Filters for type of log message displayed (critical, warning, info, debug). Note, filters from the number of messages retrieved not the whole log (this can be very time consuming). If you require more messages of the filtered type, increase the number of messages retrieved using 'number'.

show manufacture

Show manufacturer information.

Syntax

```
show manufacture [verbose]
```

Parameters

verbose: Display more information (less concise, more detail).

show modbus-gateway

Show modbus gateway status & statistics.

Syntax

```
show modbus-gateway [verbose]
```

Parameters

verbose: Display more information (less concise, more detail).

show modem

Show modem status & statistics.

Syntax

```
show modem [name STRING] [imei STRING] [verbose]
```

Parameters

name: The configured name of the modem to execute this CLI command on.

imei: The IMEI of the modem to execute this CLI command on.

verbose: Display more information (less concise, more detail).

show nemo

Show NEMO status and statistics.

Syntax

```
show nemo [name STRING]
```

Parameters

name: Display more details and configuration data for a specific NEMO instance.

show network

Show network interface status & statistics.

Syntax

```
show network [interface STRING] [all] [verbose]
```

Parameters

interface: Display more details and config data for a specific network interface.

all: Display all interfaces including disabled interfaces.

verbose: Display more information (less concise, more detail).

show ntp

Show NTP status & statistics.

Syntax

```
show ntp
```

Parameters
None

show openvpn client

Show OpenVPN client status & statistics.

Syntax

```
show openvpn client [name STRING] [all]
```

Parameters
name: Display more details and config data for a specific OpenVPN client.
all: Display all clients including disabled clients.

show openvpn server

Show OpenVPN server status & statistics.

Syntax

```
show openvpn server [name STRING] [all]
```

Parameters
name: Display more details and config data for a specific OpenVPN server.
all: Display all servers including disabled servers.

show route

Show IP routing information.

Syntax

```
show route [ipv4] [ipv6] [verbose]
```

Parameters
ipv4: Display IPv4 routes.
ipv6: Display IPv6 routes.
verbose: Display more information (less concise, more detail).

show serial

Show serial status & statistics.

Syntax

```
show serial [port STRING]
```

Parameters

port: Display more details and config data for a specific serial port.

show scripts

Show scheduled system scripts.

Syntax

```
show scripts
```

Parameters

None

show surelink interface

Show SureLink status & statistics for network interfaces.

Syntax

```
show surelink interface [name STRING] [all]
```

Parameters

name: The name of a specific network interface.

all: Show all network interfaces.

show surelink ipsec

Show SureLink status & statistics for IPsec tunnels.

Syntax

```
show surelink ipsec [tunnel STRING] [all]
```

Parameters

tunnel: The name of a specific IPsec tunnel.

all: Show all IPsec tunnels.

show surelink openvpn

Show SureLink status & statistics for OpenVPN clients.

Syntax

```
show surelink openvpn [client STRING] [all]
```

Parameters

client: The name of the OpenVPN client.

all: Show all OpenVPN clients.

show system

Show system status & statistics.

Syntax

```
show system [verbose]
```

Parameters

verbose: Display more information (disk usage, etc).

show usb

Show USB information.

Syntax

```
show usb
```

Parameters

None

show version

Show firmware version.

Syntax

```
show version [verbose]
```

Parameters

verbose: Display more information (build date).

show vrrp

Show VRRP status & statistics.

Syntax

```
show vrrp [name STRING] [all] [verbose]
```

Parameters

name: Display more details and config data for a specific VRRP instance.

all: Display all VRRP instances including disabled instances.

verbose: Display all VRRP status and statistics including disabled instances.

show web-filter

Show web filter status & statistics.

Syntax

```
show web-filter
```

Parameters

None

speedtest

Perform a speed test to a remote host using nuttcp or iPerf. The system's primary default route will be used. The speed test will take approximately 30 seconds to complete.

Syntax

```
speedtest <host> [size INTEGER] [mode <nuttcp|iperf>] [output <text|json>]
```

Parameters

host: The name or address of the remote speed test host/server.

size: The number of kilobytes sent in the speed test packets. (Minimum: 0, Default: 1000)

mode: The type of speed test protocol to run. (Default: nuttcp)

output: The format of output to display the speed test results as. (Default: text)

ssh

Use SSH protocol to log into a remote server.

Syntax

```
ssh <host> <user> [port INTEGER] [command STRING]
```

Parameters

host: The hostname or IP address of the remote host.

user: The username to use when connecting to the remote host.

port: The SSH port to use to connect to the remote host. (Minimum: 1, Maximum: 65535, Default: 22)

command: The command that will be automatically executed once the SSH session to the remote host is established.

system backup

Save the device's configuration to a file. Archives are full backups including generated SSH keys and dynamic DHCP lease information. Command backups are a list of CLI commands required to build the device's configuration.

Syntax

```
system backup [type <custom-defaults|cli-config|archive>] [path STRING]
[passphrase STRING] [remove <custom-defaults>]
```

Parameters

type: The type of backup file to create. Archives are full backups including generated SSH keys and dynamic DHCP lease information. CLI configuration backups are a list of CLI commands used to build

the device's configuration. (Default: archive)

path: The file path to save the backup to. (Default: /var/log/)

passphrase: Encrypt the archive with a passphrase.

remove: Remove a backup file.

system disable-cryptography

Erase the device's configuration and reboot into a limited mode with no cryptography available. The device's shell will be accessible over Telnet (port 23) at IP address 192.168.210.1. To return the device to normal operation, perform the configuration erase procedure with the device's ERASE button twice consecutively.

Syntax

```
system disable-cryptography
```

Parameters

None

system duplicate-firmware

Duplicate the running firmware to the alternate partition so that the device will always boot the same firmware version.

Syntax

```
system duplicate-firmware
```

Parameters

None

system factory-erase

Erase the device to restore to factory defaults. All configuration and automatically generated keys will be erased.

Syntax

```
system factory-erase
```

Parameters

None

system find-me

Find Me function to flash LEDs on this device to help users locate the unit.

Syntax

```
system find-me <state>
```

Parameters

state: Find Me control to flash cellular-related LEDs.

system firmware ota check

Query the Digi firmware server for the latest device firmware version.

Syntax

```
system firmware ota check
```

Parameters

None

system firmware ota list

Query the Digi firmware server for a list of device firmware versions.

Syntax

```
system firmware ota list
```

Parameters

None

system firmware ota update

Perform FOTA (firmware-over-the-air) update. The device will be updated to the latest firmware version unless the version argument is used to specify the firmware version.

Syntax

```
system firmware ota update [version STRING]
```

Parameters

version: Firmware version name.

system firmware update

Update the current firmware image. Upon reboot the new firmware will be run.

Syntax

```
system firmware update <file>
```

Parameters

file: Firmware filename and path.

system power ignition off_delay

Update the current ignition off delay without changing the configuration.

Syntax

```
system power ignition off_delay <off_delay>
```

Parameters

off_delay: Ignition power off delay. Format: number{h|m|s}, Max: 18h. (Minimum: 0s, Maximum: 18h)

system restore

Restore the device's configuration from a backup archive or CLI commands file.

Syntax

```
system restore <path> [passphrase STRING]
```

Parameters

path: The path to the backup file.

passphrase: Decrypt the archive with a passphrase.

system script start

Run a manual script. Scripts that are disabled, not a manual script, or already running can not be run.

Syntax

```
system script start <script>
```

Parameters

script: Script to start.

system script stop

Stop an active running script. Scripts scheduled to run again will still run again (disable a script to prevent it from running again).

Syntax

```
system script stop <script>
```

Parameters

script: Script to stop.

system serial clear

Clears the serial log.

Syntax

```
system serial clear <port>
```

Parameters

port: Serial port.

system serial save

Saves the current serial log to a file.

Syntax

```
system serial save <port> <filename>
```

Parameters

port: Serial port.

filename: The filename to save the serial log. The file will be saved to the device's /etc/config/serial directory.

system serial show

Displays the serial log on the screen.

Syntax

```
system serial show <port>
```

Parameters

port: Serial port.

system serial start

Start logging data on a serial port.

Syntax

```
system serial start <port> [size INTEGER]
```

Parameters

port: Serial port.

size: Maximum size of serial log. (Default: 65536)

system serial stop

Start logging data on a serial port.

Syntax

```
system serial stop <port>
```

Parameters

port: Serial port.

system support-report

Save a support report to a file and include with support requests.

Syntax

```
system support-report [path STRING]
```

Parameters

path: The file path to save the support report to. (Default: /var/log/)

system time set

Set the local date and time using the timezone set in the system.time.timezone config setting.

Syntax

```
system time set <datetime>
```

Parameters

datetime: The date in year-month-day hour:minute:second format (e.g "2021-09-26 12:24:48").

system time sync

Perform a NTP query to the configured server(s) and set the local time to the first server that responds.

Syntax

```
system time sync
```

Parameters

None

system time test

Test the configured NTP server(s) for connectivity. This test will not affect the device's current local date and time.

Syntax

```
system time test
```

Parameters

None

telnet

Use Telnet protocol to log into a remote server.

Syntax

```
telnet <host> [port INTEGER]
```

Parameters

host: The hostname or IP address of the remote host.

port: The telnet port to use to connect to the remote host. (Minimum: 1, Maximum: 65535, Default: 23)

traceroute

Print the route packets trace to network host.

Syntax

```
traceroute <host> [ipv6] [gateway STRING] [interface STRING] [first_ttl
INTEGER] [max_ttl INTEGER] [port INTEGER] [nqueries INTEGER] [src_addr STRING]
[tos INTEGER] [waittime INTEGER] [pausemsecs INTEGER] [packetlen INTEGER]
[debug] [dontfragment] [icmp] [nomap] [bypass]
```

Parameters

host: The host that we wish to trace the route packets for.

ipv6: If a hostname is defined as the value of the 'host' parameter, use the hosts IPV6 address.

gateway: Tells traceroute to add an IP source routing option to the outgoing packet that tells the network to route the packet through the specified gateway.

interface: Specifies the interface through which traceroute should send packets. By default, the interface is selected according to the routing table.

first_ttl: Specifies with what TTL to start. (Minimum: 1, Default: 1)

max_ttl: Specifies the maximum number of hops (max time-to-live value) traceroute will probe. (Minimum: 1, Default: 30)

port: Specifies the destination port base traceroute will use (the destination port number will be incremented by each probe). A value of -1 specifies that no specific port will be used. (Minimum: -1, Default: -1)

nqueries: Sets the number of probe packets per hop. A value of -1 indicated. (Minimum: 1, Default: 3)

src_addr: Chooses an alternative source address. Note that you must select the address of one of the interfaces. By default, the address of the outgoing interface is used.

tos: For IPv4, set the Type of Service (ToS) and Precedence value. Useful values are 16 (low delay) and 8 (high throughput). Note that in order to use some TOS precedence values, you have to be super user. For IPv6, set the Traffic Control value. A value of -1 specifies that no value will be used. (Minimum: -1, Default: -1)

waittime: Determines how long to wait for a response to a probe. (Minimum: 1, Default: 5)

pausemsecs: Minimal time interval between probes. (Minimum: 0, Default: 0)

packetlen: Total size of the probing packet. Default 60 bytes for IPv4 and 80 for Ipv6. A value of -1 specifies that the default value will be used. (Minimum: -1, Default: -1)

debug: Enable socket level debugging.

dontfragment: Do not fragment probe packets.

icmp: Use ICMP ECHO for probes.

nomap: Do not try to map IP addresses to host names when displaying them.

bypass: Bypass the normal routing tables and send directly to a host on an attached network.

xbee

XBee commands.

xbee execute XBEE-ID AT-COMMAND [value]

Execute an AT command in an XBee node.

Parameters

XBEE-ID
The XBee identifier (MAC or node identifier) of the XBee to execute the command.
Syntax: STRING

AT-COMMAND
The AT command to execute.
Syntax: STRING

value
The value of the command to execute.
Syntax: Decimal number, hexadecimal number with prefix 0x, or string between quotes.
Optional: True

xbee get XBEE-ID PARAMETER [decimal]

Read a parameter from an XBee node.

Parameters

XBEE-ID
The XBee identifier (MAC or node identifier) of the XBee to read the parameter from.
Syntax: STRING

PARAMETER
The parameter name to read.
Syntax: STRING

decimal
Show the read value in decimal, otherwise in hexadecimal.
Syntax: BOOLEAN
Default: False
Optional: True

xbee network clear

Clear the XBee network cache.

Parameters
None

xbee network export EXPORTDIR [description STRING] [name STRING]

Export the XBee network to a file.

Parameters

EXPORTDIR
Absolute path of the directory to create the xnet file.
Syntax: STRING

description
A brief description of the network.
Syntax: STRING
Optional: True

name
The name for the XBee network.
Syntax: STRING
Optional: True

xbee network discover [async]

Discover the XBee network.

Parameters

async
Leave the discovery process in background and returns the prompt inmediately
Syntax: BOOLEAN
Default: False
Optional: True

xbee network show

Show network cache in table format.

Parameters
None

xbee set XBEE-ID PARAMETER VALUE [no-apply] [no-save]

Write a parameter value in a XBee node.

Parameters

XBEE-ID
The XBee identifier (MAC or node identifier) of the XBee to write the parameter to.
Syntax: STRING

PARAMETER
The parameter name to write.
Syntax: STRING

VALUE

The new value of the parameter.

Syntax: Decimal number, hexadecimal number with prefix 0x, or string between quotes.

no-apply

Do not apply the change.

Syntax: BOOLEAN

Default: False

Optional: True

no-save

Do not write to flash the change.

Syntax: BOOLEAN

Default: False

Optional: True

xbee update XBEE-ID PROFILE

Update the XBee with a profile.

Parameters

XBEE-ID

The XBee identifier (MAC or node identifier) of the XBee to update.

Syntax: STRING

PROFILE

The path to the profile to update.

Syntax: STRING

Diagnostics

This chapter contains the following topics:

Perform a speedtest 880
Generate a support report 880
View system and event logs 882
Configure syslog servers 886
Configure options for the event and system logs 889
Analyze network traffic 894
Use the ping command to troubleshoot network connections 912
Use the traceroute command to diagnose IP routing problems 912

Perform a speedtest

To perform a speedtest:

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Use the **speedtest** command to generate the report:

   ```
   > speedtest host
   ```

   where *host* is the hostname or IP address of a speedtest host. For example:

   ```
   > speedtest speedtest.accns.com
   Tx (upload) average: 50.1110 Mbps
   Tx latency: 31.45 ms
   Rx (download) average: 44.7588 Mbps
   Rx latency: 30.05 ms
   >
   ```

3. To output the result in json format, use the **output** parameter:

   ```
   > speedtest host output json
   {"tx_avg": "51.8510", "tx_avg_units": "Mbps", "tx_latency": "31.07",
   "tx_latency_units": "ms", "rx_avg": "39.5770", "rx_avg_units": "Mbps",
   "rx_latency": "34.19", "rx_latency_units": "ms" }
   >
   ```

4. To change the size of the speedtest packet, use the **size** parameter:

   ```
   > speedtest host size int
   ```

5. By default, the speedtest uses nuttcp for the mode. You can change this to iperf with the **mode** parameter:

   ```
   > speedtest host mode iperf
   ```

6. Type **exit** to exit the Admin CLI.

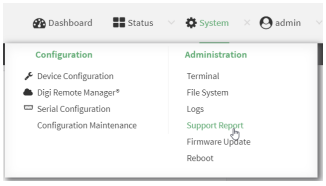
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Generate a support report

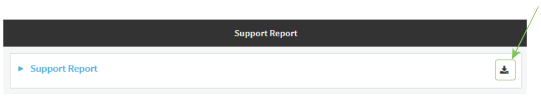
To generate and download a support report:

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **System**. Under **Administration**, click **Support Report**.



3. Click to generate and download the support report.



Attach the support report to any support requests.

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Use the **system support-report** command to generate the report:

```
> system support-report /var/log/
Saving support report to /var/log/support-report-0040D0133536-22-03-03-
10:16:23.bin
Support report saved.
>
```

3. Use the **scp** command to transfer the report to a remote host:

```
> scp host 192.168.4.1 user admin remote /home/admin/temp/ local
/var/log/support-report-00:40:D0:13:35:36-22-03-03-10:16:23.bin to remote
admin@192.168.4.1's password: adminpwd
support-report-0040D0133536-22-03-03-10:16:23.bin
>
```

4. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

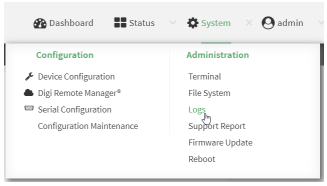
View system and event logs

See Configure options for the event and system logs for information about configuring the information displayed in event and system logs.

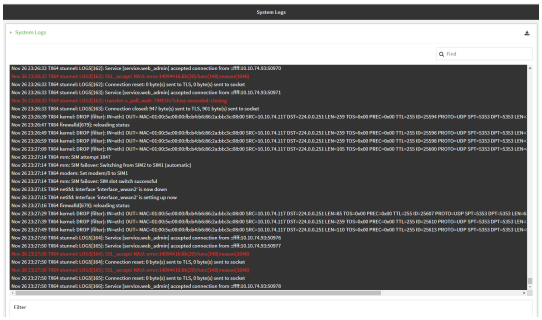
View System Logs

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **System** > **Logs**.



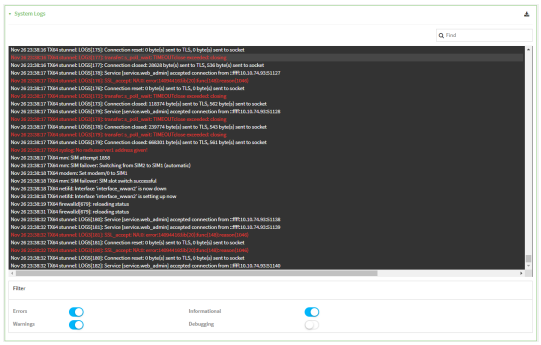
The system log displays:



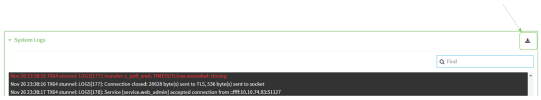
3. Limit the display in the system log by using the **Find** search tool.



4. Use filters to configure the types of information displayed in the system logs.



5. Click to download the system log.



Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Use the show log command at the Admin CLI prompt:

```
> show log

Timestamp        Message
--------------  ------------------------------------------------------------
----
Nov 26 21:54:34  IX15 netifd: Interface 'interface_wan' is setting up now
Nov 26 21:54:35  IX15 firewalld[621]: reloading status
...
>
```

3. (Optional) Use the **show log number *num*** command to limit the number of lines that are displayed. For example, to limit the log to the most recent ten lines:

```
> show log number 10

Timestamp        Message
--------------  ------------------------------------------------------------
----
Nov 26 21:54:34  IX15 netifd: Interface 'interface_wan' is setting up now
Nov 26 21:54:35  IX15 firewalld[621]: reloading status
...
>
```

4. (Optional) Use the **show log filter *value*** command to limit the number of lines that are displayed. Allowed values are **critical**, **warning**, **info**, and **debug**. For example, to limit the event list to only info messages:

```
> show log filter info

Timestamp        Type    Category  Message
---------------- ------- --------- -------------------------------------
-----
Nov 26 22:01:26  info    user
name=admin~service=cli~state=opened~remote=192.168.1.2
Nov 26 22:01:25  info    user
name=admin~service=cli~state=closed~remote=192.168.1.2
...
>
```

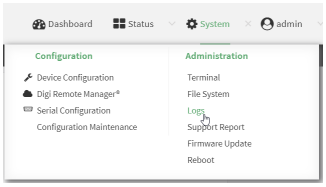
5. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

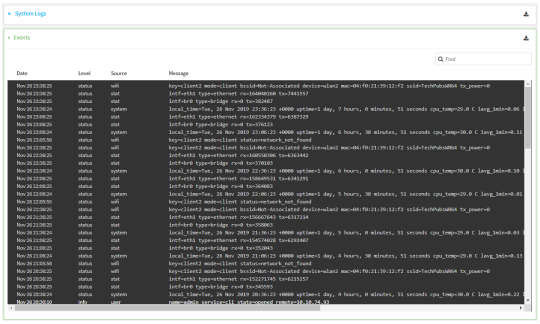
View Event Logs

WebUI

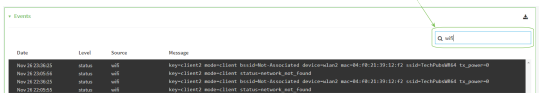
1. Log into the IX15 WebUI as a user with Admin access.
2. On the main menu, click **System** > **Logs**.



3. Click ▾ **System Logs** to collapse the system logs viewer, or scroll down to **Events**.
4. Click ▸ **Events** to expand the event viewer.



5. Limit the display in the event log by using the **Find** search tool.



6. Click to download the event log.



Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Use the show event command at the Admin CLI prompt:

```
> show event

Timestamp        Type    Category  Message
---------------- ------- --------- ------------------------------------
-----
Nov 26 21:42:37  status  stat
intf=eth1~type=ethernet~rx=11332435~tx=5038762
Nov 26 21:42:35  status  system    local_time=Thu, 08 Aug 2019 21:42:35
+0000~uptime=3 hours, 0 minutes, 48 seconds
...
>
```

3. (Optional) Use the **show event number *num*** command to limit the number of lines that are displayed. For example, to limit the event list to the most recent ten lines:

```
> show event number 10

Timestamp        Type    Category  Message
---------------- ------- --------- ------------------------------------
-----
Nov 26 21:42:37  status  stat
intf=eth1~type=ethernet~rx=11332435~tx=5038762
Nov 26 21:42:35  status  system    local_time=Thu, 08 Aug 2019 21:42:35
+0000~uptime=3 hours, 0 minutes, 48 seconds
...
>
```

4. (Optional) Use the **show event table *value*** command to limit the number of lines that are displayed. Allowed values are **error**, **info**, and **status**. For example, to limit the event list to only info messages:

```
> show event table info

Timestamp        Type    Category  Message
---------------- ------- --------- ------------------------------------
-----
Nov 26 22:01:26  info    user
name=admin~service=cli~state=opened~remote=192.168.1.2
Nov 26 22:01:25  info    user
name=admin~service=cli~state=closed~remote=192.168.1.2
...
>
```

5. Type **exit** to exit the Admin CLI.

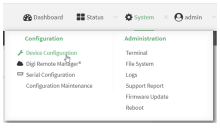
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure syslog servers

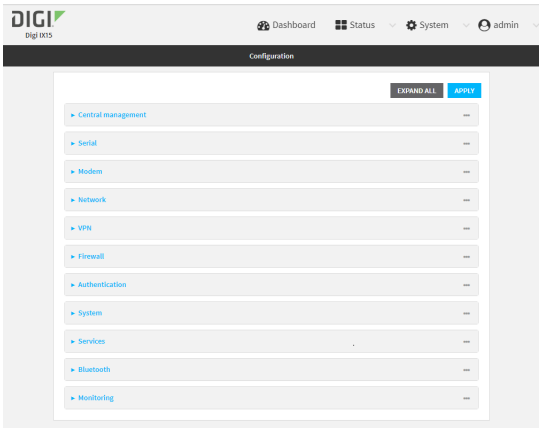
You can configure remote syslog servers for storing event and system logs.

WebUI

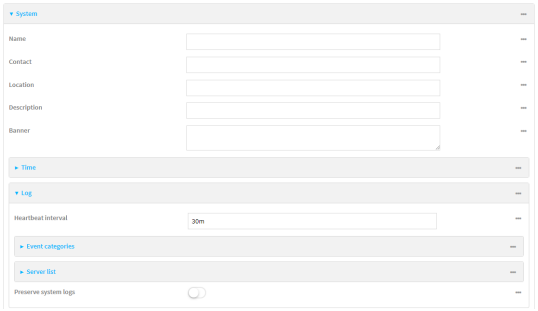
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



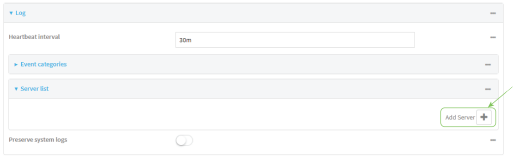
The **Configuration** window is displayed.



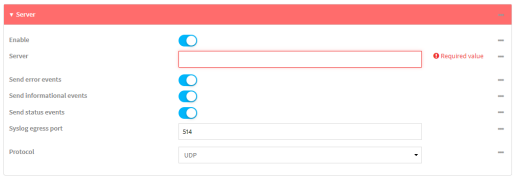
3. Click **System** > **Log**.



4. Add and configure a remote syslog server:
   a. Click to expand **Server list**.
   b. For **Add Server**, click ✚.



   The log server configuration window is displayed.



   Log servers are enabled by default. To disable, click to toggle off **Enable**.
   c. Type the host name or IP address of the **Server**.
   d. Select the event categories that will be sent to the server. By default, all event categories are enabled. You can disable logging for error, informational, and status event categories by clicking to toggle off the category.
   e. For **Syslog egress port**, type the port number to use for the syslog server. The default is **514**.
   f. For **Protocol**, select the IP protocol to use for communication with the syslog server. Available options are **TCP** and **UPD**. The default is **UPD**.
5. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. (Optional) To configure remote syslog servers:
   a. Add a remote server:

```
(config)> add system log remote end
(config system log remote 0)>
```

   b. Enable the server:

```
(config system log remote 0)> enable true
(config system log remote 0)>
```

   c. Set the host name or IP address of the server:

```
(config system log remote 0)> server hostname
(config system log remote 0)>
```

   d. The event categories that will be sent to the server are automatically enabled when the server is enabled.
      - To disable informational event messages:

```
(config system log remote 0)> info false
(config system log remote 0)>
```

      - To disable status event messages:

```
(config system log remote 0)> status false
(config system log remote 0)>
```

      - To disable informational event messages:

```
(config system log remote 0)> error false
(config system log remote 0)>
```

4. Set the port number to use for the syslog server:

```
(config system log remote 0)> port value
(config system log remote 0)>
```

   where *value* is any integer between **1** and **65535**. The default is **514**.
5. Set the IP protocol to use for communication with the syslog server:

```
(config system log remote 0)> protocol value
(config system log remote 0)>
```

   where *value* is either **tcp** or **udp**. The default is **udp**.
6. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

7. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Configure options for the event and system logs

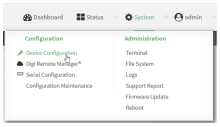
The default configuration for event and system logging is:

- The heartbeat interval, which determines the amount of time to wait before sending a heartbeat event if no other events have been sent, is set to 30 minutes.
- All event categories are enabled.

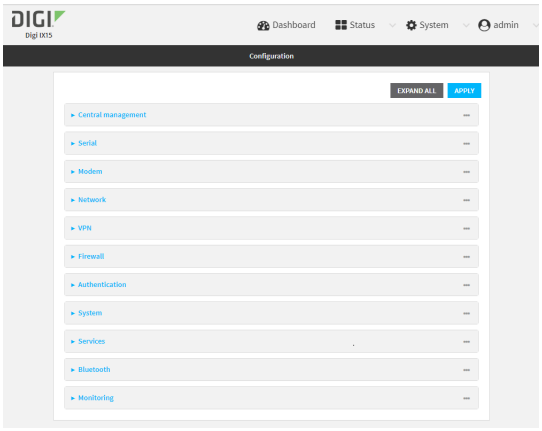
To change or disable the heartbeat interval, or to disable event categories, and to perform other log configuration:

WebUI

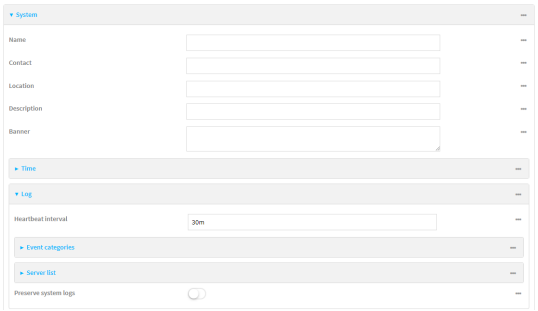
1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



   The **Configuration** window is displayed.



3. Click **System** > **Log**.



4. (Optional) To change the **Heartbeat interval** from the default of 30 minutes, type a new value. The heartbeat interval determines the amount of time to wait before sending a heartbeat event if no other events have been sent.

   Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w|d|h|m|s**}.

   For example, to set **Heartbeat interval** to ten minutes, enter **10m** or **600s**.

   To disable the **Heartbeat interval**, enter **0s**.

5. (Optional) To disable event categories, or to enable them if they have been disabled:
   a. Click to expand **Event Categories**.
   b. Click an event category to expand.
   c. Depending on the event category, you can enable or disable informational events, status events, and error events. Some categories also allow you to set the **Status interval**, which is the time interval between periodic status events.

6. (Optional) See Configure syslog servers for information about configuring remote syslog servers to which log messages will be sent.

7. Enable **Preserve system logs** to save the current session's system log after a reboot.

   By default, the IX15 device erases system logs each time the device is powered off or rebooted.

   Note You should only enable **Preserve system logs** temporarily to debug issues. Once you are finished debugging, immediately disable **Preserve system logs** to avoid unnecessary wear to the flash memory.

8. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. At the command line, type **config** to enter configuration mode:

```
> config
(config)>
```

3. (Optional) To change the heartbeat interval from the default of 30 minutes, set a new value. The heartbeat interval determines the amount of time to wait before sending a heartbeat event if no other events have been sent.

```
(config)> system log heartbeat_interval value
(config)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **the heartbeat interval** to ten minutes, enter either **10m** or **600s**:

```
(config)> system log heartbeat_interval 600s
(config)>
```

To disable the heartbeat interval, set the value to **0s**

4. Enable preserve system logs functionality to save the current session's system log after a reboot. By default, the IX15 device erases system logs each time the device is powered off or rebooted.

Note You should only enable **Preserve system logs** temporarily to debug issues. Once you are finished debugging, immediately disable **Preserve system logs** to avoid unnecessary wear to the flash memory.

```
(config)> system log persistent true
(config)>
```

5. (Optional) To disable event categories, or to enable them if they have been disabled:
   a. Use the question mark (**?**) to determine available event categories:

```
(config)> system log event ?

Event categories: Settings to enable individual event categories.

 Additional Configuration
 ------------------------------------------------------------------------
----------
 arping                    ARP ping
 config                    Configuration
 dhcpserver                DHCP server
 firmware                  Firmware
 location                  Location
 modem                     Modem
 netmon                    Active recovery
 network                   Network interfaces
 openvpn                   OpenVPN
```

```
portal                    Captive portal
remote                    Remote control
restart                   Restart
serial                    Serial
sms                       SMS commands
speed                     Speed
stat                      Network statistics
user                      User
wol                       Wake-On-LAN

(config)> system log event
```

b. Depending on the event category, you can enable or disable informational events, status events, and error events. Some categories also allow you to set the status interval, which is the time interval between periodic status events. For example, to configure DHCP server logging:

   i. Use the question mark (**?**) to determine what events are available for DHCP server logging configuration:

```
(config)> system log event dhcpserver ?
...
DHCP server: Settings for DHCP server events. Informational events
are generated
when a lease is obtained or released. Status events report the
current list of
leases.

 Parameters               Current Value
 ------------------------------------------------------------------
--------------
 info                     true          Enable informational
events
 status                   true          Enable status events
 status_interval          30m           Status interval

(config)> system log event dhcpserver
```

   ii. To disable informational messages for the DHCP server:

```
(config)> system log event dhcpserver info false
(config)>
```

   iii. To change the status interval:

```
(config)> system log event dhcpserver status_interval value
(config)>
```

   where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

   For example, to set **the status interval** to ten minutes, enter either **10m** or **600s**:

```
(config)> system log event dhcpserver status_interval 600s
(config)>
```

6. (Optional) See Configure syslog servers for information about configuring remote syslog servers to which log messages will be sent.
7. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

8. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Analyze network traffic

The IX15 device includes a network analyzer tool that captures data traffic on any interface and decodes the captured data traffic for diagnostics. You can capture data traffic on multiple interfaces at the same time and define capture filters to reduce the captured data. You can capture up to 10 MB of data traffic in two 5 MB files per interface.

To perform a more detailed analysis, you can download the captured data traffic from the device and view it using a third-party application.

Note Data traffic is captured to RAM and the captured data is lost when the device reboots unless you save the data to a file. See Save captured data traffic to a file.

This section contains the following topics:

Configure packet capture for the network analyzer 895
Example filters for capturing data traffic 904
Capture packets from the command line 905
Stop capturing packets 906
Show captured traffic data 907
Save captured data traffic to a file 908
Download captured data to your PC 909
Clear captured data 910

Configure packet capture for the network analyzer

To use the network analyzer, you must create one or more packet capture configuration.

Required configuration items

- The interface used by this packet capture configuration.

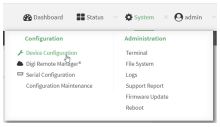
Additional configuration items

- The filter expression for this packet capture configuration.
- Schedule the analyzer to run based on a specified event or at a particular time:
  - The events or time that will trigger the analyzer to run, using this capture configuration.
  - The amount of time that the analyzer session will run.
  - The frequency with which captured events will be saved.

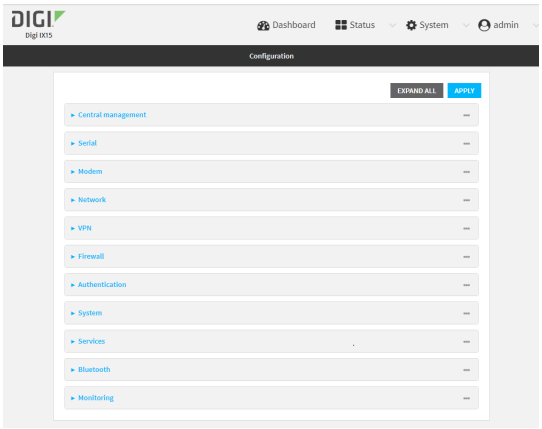
To configure a packet capture configuration:

WebUI

1. Log into the IX15 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

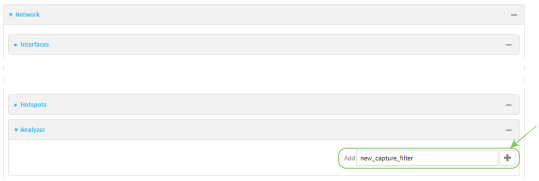


The **Configuration** window is displayed.

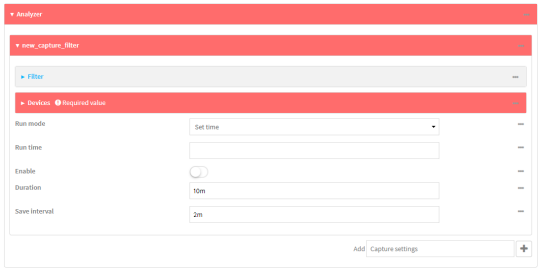


3. Click **Network** > **Analyzer**.

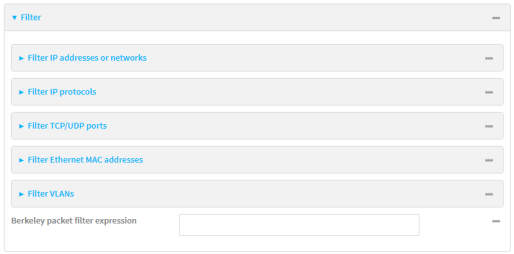
4. For **Add Capture settings**, type a name for the capture filter and click ✚.



The new capture filter configuration is displayed.

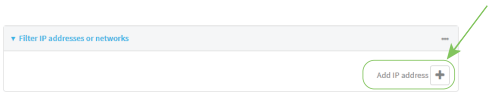


5. (Optional) Add a filter type:
   a. Click to expand **Filter**.



   You can select from preconfigured filters to determine which types of packets to capture or ignore, or you can create your own Berkeley packet filter expression.

   b. To create a filter that either captures or ignores packets from a particular IP address or network:
      i. Click to expand **Filter IP addresses or networks**.
      ii. Click ✚ to add an IP address/network.



      iii. For **IP address or network**, type the IPv4 or IPv6 address (and optional netmask).
      iv. For **Source or destination IP address**, select whether the filter should apply to packets when the IP address/network is the source, the destination, or both.
      v. Click **Ignore this IP address or network** if the filter should ignore packets from this IP address/network. By default, is option is disabled, which means that the filter will capture packets from this IP address/network.
      vi. Click ✚ to add additional IP address/network filters.

c. To create a filter that either captures or ignores packets that use a particular IP protocol:
   i. Click to expand **Filter IP protocols**.
   ii. Click ✚ to add an IP protocol.
   iii. For **IP protocol to capture or ignore**, select the protocol. If **Other protocol** is selected, type the number of the protocol.
   iv. Click **Ignore this protocol** if the filter should ignore packets that use this protocol. By default, is option is disabled, which means that the filter will capture packets that use this protocol.
   v. Click ✚ to add additional IP protocols filters.

d. To create a filter that either captures or ignores packets from a particular port:
   i. Click to expand **Filter TCP/UDP port**.
   ii. Click ✚ to add a TCP /UDP port.
   iii. For **IP TCP/UDP port to capture or ignore**, type the number of the port to be captured or ingored.
   iv. For **TCP or UDP port**, select the type of transport protocol.
   v. For **Source or destination TCP/UDP port**, select whether the filter should apply to packets when the port is the source, the destination, or both.
   vi. Click **Ignore this TCP/UDP port** if the filter should ignore packets that use this port. By default, is option is disabled, which means that the filter will capture packets that use this port.
   vii. Click ✚ to add additional port filters.

e. To create a filter that either captures or ignores packets from one or more specified MAC addresses:
   i. Click to expand **Filter Ethernet MAC addresses**.
   ii. Click ✚ to add a MAC address.
   iii. For **Ethernet MAC address**, type the MAC address to be captured or ingored.
   iv. For **Source or destination Ethernet MAC address**, select whether the filter should apply to packets when the Ethernet MAC address is the source, the destination, or both.
   v. Click **Ignore this MAC address** if the filter should ignore packets that use this port. By default, is option is disabled, which means that the filter will capture packets that use this port.
   vi. Click ✚ to add additional MAC address filters.

f. To create a filter that either captures or ignores packets from one or more VLANs:
   i. Click to expand **Filter VLANs**.
   ii. Click ✚ to add a VLAN.
   iii. For **The VLAN to capture or ignore**, type the number of the VLAN.
   iv. Click **Ignore this VLAN** if the filter should ignore packets that use this port. By default, is option is disabled, which means that the filter will capture packets that use this port.
   v. Click ✚ to add additional VLAN filters.

g. For **Berkeley packet filter expression**, type a filter using Berkeley Packet Filter (BPF) syntax. See Example filters for capturing data traffic for examples of filters using BPF syntax.

6. Add one or more interface to the capture filter:
   a. Click to expand **Device**.
   b. Click ✚ to add an interface to the capture setting instance.



   c. For **Device**, select an interface.
   d. Repeat to add additional interfaces to the capture filter.
7. (Optional) For **Berkeley packet filter expression**, type a filter using Berkeley Packet Filter (BPF) syntax. See Example filters for capturing data traffic for examples of filters using BPF syntax.
8. (Optional) Schedule the analyzer to run, using this capture filter, based on a specified event or at a particular time:
   a. For **Run mode**, select the mode that will be used to run the capture filter. Available options are:
      - **On boot**: The capture filter will run once each time the device boots.
      - **Interval**: The capture filter will start running at the specified interval, within 30 seconds after the configuration change is saved.
        - If **Interval** is selected, in **Interval**, type the interval.

          Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w**|**d**|**h**|**m**|**s**}.

          For example, to set **Interval** to ten minutes, enter **10m** or **600s**.
      - **Set time**: Runs the capture filter at a specified time of the day.
        - If **Set Time** is selected, specify the time that the capture filter should run in **Run time**, using the format *HH*:*MM*.
      - **During system maintenance**: The capture filter will run during the system maintenance time window.
   b. **Enable** the capture filter schedule.
   c. For **Duration**, type the amount of time that the scheduled analyzer session will run.

      Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w**|**d**|**h**|**m**|**s**}.

      For example, to set **Duration** to ten minutes, enter **10m** or **600s**.
   d. For **Save interval**, type the frequency with which captured events will be saved.

      Allowed values are any number of weeks, days, hours, minutes, or seconds, and take the format ***number***{**w**|**d**|**h**|**m**|**s**}.

      For example, to set **Save interval** to ten minutes, enter **10m** or **600s**.
9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX15 command line as a user with full Admin access rights.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the command line, type **config** to enter configuration mode:

   ```
   > config
   (config)>
   ```
3. Add a new capture filter:

   ```
   (config)> add network analyzer name
   (config network analyzer name)>
   ```
4. Add an interface to the capture filter:

   ```
   (config network analyzer name)> add device end device
   (config network analyzer name)>
   ```

   Determine available devices and the proper syntax.

   To determine available devices and proper syntax, use the space bar autocomplete feature:

   ```
   (config network analyzer name)> add device end <space>
   (config network analyzer name)> add interface end /network/
   ```

   Repeat to add additional interfaces.
5. (Optional) Set a filter for the capture filter:
   a. To create a filter that either captures or ignores packets from a particular IP address or network:
      i. Add a new IP address/network filter:

      ```
      (config network analyzer name)> add filter address end
      (config network analyzer name filter address 0)>
      ```

      ii. Set the IPv4 or IPv6 address (and optional netmask):

      ```
      (config network analyzer name filter address 0)> address ip_
      address[/netmask]
      (config network analyzer name filter address 0)>
      ```

      iii. Set whether the filter should apply to packets when the IP address/network is the source, the destination, or both:

```
(config network analyzer name filter address 0)> match value
(config network analyzer name filter address 0)>
```

where *value* is one of:

- **source**: The filter will apply to packets when the IP address/network is the source.
- **destination**: The filter will apply to packets when the IP address/network is the destination.
- **either**: The filter will apply to packets when the IP address/network is either the source or the destination.

iv. (Optional) Set the filter should ignore packets from this IP address/network:

```
(config network analyzer name filter address 0)> ignore true
(config network analyzer name filter address 0)>
```

By default, is option is set to **false**, which means that the filter will capture packets from this IP address/network.

v. Repeat these steps to add additional IP address filters.

b. To create a filter that either captures or ignores packets that use a particular IP protocol:

i. Add a new IP protocol filter:

```
(config network analyzer name)> add filter protocol end
(config network analyzer name filter protocol 0)>
```

ii. Use the **?** to determine available protocols and the appropriate format:

```
(config network analyzer name filter protocol 0)> protocol ?

IP protocol to capture or ignore: IP protocol to capture or
ignore.
Format:
  ah
  esp
  gre
  icmp
  icmpv6
  igmp
  ospf
  other
  tcp
  udp
  vrrp
Current value:

(config network analyzer name filter protocol 0)>
```

iii. Set the protocol:

```
(config network analyzer name filter protocol 0)> protocol value
(config network analyzer name filter protocol 0)>
```

iv. If other is set for the protocol, set the number of the protocol:

```
(config network analyzer name filter protocol 0)> protocol_other
value
(config network analyzer name filter protocol 0)>
```

where *value* is an integer between 1 and 255 and represents the the number of the protocol.

v. (Optional) Set the filter should ignore packets from this protocol:

```
(config network analyzer name filter protocol 0)> ignore true
(config network analyzer name filter protocol 0)>
```

By default, is option is set to **false**, which means that the filter will capture packets from this protocol.

vi. Repeat these steps to add additional protocol filters.

c. To create a filter that either captures or ignores packets from a particular port:

i. Add a new port filter:

```
(config network analyzer name)> add filter port end
(config network analyzer name filter port 0)>
```

ii. Set the transport protocol that should be filtered for the port:

```
(config network analyzer name filter port 0)> protocol value
(config network analyzer name filter port 0)>
```

where *value* is one of **tcp**, **udp**, or **either**. The default is either.

iii. Set whether the filter should apply to packets when the port is the source, the destination, or both:

```
(config network analyzer name filter port 0)> match value
(config network analyzer name filter port 0)>
```

where *value* is one of:

- **source**: The filter will apply to packets when the port is the source.
- **destination**: The filter will apply to packets when the port is the destination.
- **either**: The filter will apply to packets when the port is either the source or the destination.

iv. (Optional) Set the filter should ignore packets from this port:

```
(config network analyzer name filter port 0)> ignore true
(config network analyzer name filter port 0)>
```

By default, is option is set to **false**, which means that the filter will capture packets from this port.

v. Repeat these steps to add additional port filters.

d. To create a filter that either captures or ignores packets from one or more specified MAC addresses:

i. Add a new MAC address filter:

```
(config network analyzer name)> add filter mac_address end
(config network analyzer name filter mac_address 0)>
```

ii. Set the MAC address that should be be captured or ignored:

```
(config network analyzer name filter mac_address 0)> address value
(config network analyzer name filter mac_address 0)>
```

where *value* is the MAC address to be filtered, using colon-hexadecimal notation with lower case, for example, **00:aa:11:bb:22:cc**.

iii. Set whether the filter should apply to packets when the MAC address is the source, the destination, or both:

```
(config network analyzer name filter mac_address 0)> match value
(config network analyzer name filter mac_address 0)>
```

where *value* is one of:

- **source**: The filter will apply to packets when the MAC address is the source.
- **destination**: The filter will apply to packets when the MAC address is the destination.
- **either**: The filter will apply to packets when the MAC address is either the source or the destination.

iv. (Optional) Set the filter should ignore packets from this port:

```
(config network analyzer name filter mac_address 0)> ignore true
(config network analyzer name filter mac_address 0)>
```

By default, is option is set to **false**, which means that the filter will capture packets from this MAC address.

v. Repeat these steps to add additional MAC addresses.

e. To create a filter that either captures or ignores packets from one or more specified VLANs:

i. Add a new VLAN filter:

```
(config network analyzer name)> add filter vlan end
(config network analyzer name filter vlan 0)>
```

ii. Set the VLAN that should be be captured or ignored:

```
(config network analyzer name filter vlan 0)> vlan value
(config network analyzer name filter vlan 0)>
```

where *value* is number o the VLAN.

iii. (Optional) Set the filter should ignore packets from this VLAN:

```
(config network analyzer name filter vlan 0)> ignore true
(config network analyzer name filter vlan  0)>
```

By default, is option is set to **false**, which means that the filter will capture packets from this MAC address.

iv. Repeat these steps to add additional VLANs.

f. To create a filter using Berkeley Packet Filter (BPF) syntax:

```
(config network analyzer name)> filter custom value
(config network analyzer name)>
```

where *value* is a filter using Berkeley Packet Filter (BPF) syntax. Values that contain spaces must be enclosed in double quotes (**"**).

See Example filters for capturing data traffic for examples of filters using BPF syntax.

6. (Optional) Schedule the analyzer to run, using this capture filter, based on a specified event or at a particular time:

a. Enable scheduling for this capture filter:

```
(config network analyzer name)> schedule enable true
(config network analyzer name)>
```

b. Set the mode that will be used to run the capture filter:

```
(config network analyzer name)> when mode
(config network analyzer name)>
```

where *mode* is one of the following:

- **boot**: The script will run once each time the device boots.
- **interval**: The script will start running at the specified interval, within 30 seconds after the configuration change is saved. If **interval** is selected, set the interval:

```
(config add network analyzer name)> on_interval value
(config add network analyzer name)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **on_interval** to ten minutes, enter either **10m** or **600s**:

```
(config network analyzer name)> on_interval  600s
(config network analyzer name)>
```

- **set_time**: Runs the script at a specified time of the day. If **set_time** is set, set the time that the script should run, using the format *HH*:*MM*:

```
(config network analyzer name)> run_time HH:MM
(config network analyzer name)>
```

- **maintenance_time**: The script will run during the system maintenance time window.

c. Set the amount of time that the scheduled analyzer session will run:

```
(config network analyzer name)> duration value
(config network analyzer name)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **duration** to ten minutes, enter either **10m** or **600s**:

```
(config network analyzer name)> save_interval 600s
(config network analyzer name)>
```

d. Set the frequency with which captured events will be saved:

```
(config network analyzer name)> save_interval value
(config network analyzer name)>
```

where *value* is any number of weeks, days, hours, minutes, or seconds, and takes the format ***number***{**w|d|h|m|s**}.

For example, to set **save_interval** to ten minutes, enter either **10m** or **600s**:

```
(config network analyzer name)> save_interval 600s
(config network analyzer name)>
```

7. Save the configuration and apply the change:

```
(config)> save
Configuration saved.
>
```

8. Type **exit** to exit the Admin CLI.

Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Example filters for capturing data traffic

The following are examples of filters using Berkeley Packet Filter (BPF) syntax for capturing several types of network data. See https://biot.com/capstats/bpf.html for detailed information about BPF syntax.

Example IPv4 capture filters

- Capture traffic to and from IP host 192.168.1.1:

```
ip host 192.168.1.1
```

- Capture traffic from IP host 192.168.1.1:

```
ip src host 192.168.1.1
```

- Capture traffic to IP host 192.168.1.1:

```
ip dst host 192.168.1.1
```

- Capture traffic for a particular IP protocol:

```
ip proto protocol
```

  where *protocol* is a number in the range of **1** to **255** or one of the following keywords: **icmp**, **icmp6**, **igmp**, **pim**, **ah**, **esp**, **vrrp**, **udp**, or **tcp**.
- Capture traffic to and from a TCP port 80:

```
ip proto tcp and port 80
```

- Capture traffic to UDP port 53:

```
ip proto udp and dst port 53
```

- Capture traffic from UDP port 53:

```
ip proto udp and src port 53
```

- Capture to and from IP host 10.0.0.1 but filter out ports 22 and 80:

```
ip host 10.0.0.1 and not (port 22 or port 80)
```

Example Ethernet capture filters

- Capture Ethernet packets to and from a host with a MAC address of 00:40:D0:13:35:36:

```
ether host 00:40:D0:13:35:36
```

- Capture Ethernet packets from host 00:40:D0:13:35:36:

```
ether src 00:40:D0:13:35:36:
```

- Capture Ethernet packets to host 00:40:D0:13:35:36:

```
ether dst 00:40:D0:13:35:36
```

Capture packets from the command line

You can start packet capture at the command line with the analyzer start command. Alternatively, you can schedule the network analyzer to run based on a specified event or at a particular time. See Configure packet capture for the network analyzer for information about scheduling packet capturing.

Additional analyzer commands allow you to:

- Stop capturing packets.
- Save captured data traffic to a file.
- Clear captured data.

Required configuration items

- A configured packet capture. See Configure packet capture for the network analyzer for packet capture configuration information.

To start packet capture from the command line:

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Type the following at the Admin CLI prompt:

```
> analyzer start name capture_filter
>
```

   where *capture_filter* is the name of a packet capture configuration. See Configure packet capture for the network analyzer for more information.

   To determine available packet capture configurations, use the **?**:

```
> analyzer start name ?

name: Name of the capture filter to use.
Format:
  test_capture
  capture_ping

> analyzer start name
```

You can capture up to 10 MB of data traffic in two 5 MB files per interface.

Note Data traffic is captured to RAM and the captured data is lost when the device reboots unless you save the data to a file. See Save captured data traffic to a file.

Stop capturing packets

You can stop packet capture at the command line with the analyzer stop command.

To stop packet capture from the command line:

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Type the following at the Admin CLI prompt:

```
> analyzer stop name capture_filter
>
```

   where *capture_filter* is the name of a packet capture configuration. See Configure packet capture for the network analyzer for more information.

   To determine available packet capture configurations, use the **?**:

```
> analyzer stop name ?
```

```
name: Name of the capture filter to use.
Format:
  test_capture
  capture_ping

> analyzer stop name
```

Show captured traffic data

To view captured data traffic, use the show analyzer command. The command output show the following information for each packet:

- The packet number.
- The timestamp for when the packet was captured.
- The length of the packet and the amount of data captured.
- Whether the packet was sent or received by the device.
- The interface on which the packet was sent or received.
- A hexadecimal dump of the packet of up to 256 bytes.
- Decoded information of the packet.

To show captured data traffic:

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Type the following at the Admin CLI prompt:

```
> show analyzer name capture_filter

Packet 1 : Mar-03-2022 10:16:23.287682, Length 60 bytes (Captured Length
60 bytes)

Received on interface eth1

      00 40 ff 80 01 20 b4 b6  86 21 b5 73 08 00 45 00   .@... .. 
.!.s..E.
      00 28 3d 36 40 00 80 06  14 bc 0a 0a 4a 82 0a 0a   .(=6@... ....J..
      4a 48 cd ae 00 16 a4 4b  ff 5f ee 1f d8 23 50 10   JH.....K 
._...#P.
      08 02 c7 40 00 00 00 00  00 00 00 00               ...@.... ....

    Ethernet Header
      Destination MAC Addr : 00:40:D0:13:35:36
      Source MAC Addr      : fb:03:53:05:11:2f
      Ethernet Type        : IP (0x0800)
    IP Header
      IP Version           : 4
```

```
      Header Length        : 20 bytes
      ToS                  : 0x00
      Total Length         : 40 bytes
      ID                   : 15670 (0x3d36)
      Flags                : Do not fragment
      Fragment Offset      : 0 (0x0000)
      TTL                  : 128 (0x80)
      Protocol             : TCP (6)
      Checksum             : 0x14bc
      Source IP Address    : 10.10.74.130
      Dest. IP Address     : 10.10.74.72
    TCP Header
      Source Port          : 52654
      Destination Port     : 22
      Sequence Number      : 2756443999
      Ack Number           : 3995064355
      Data Offset          : 5
      Flags                : ACK
      Window               : 2050
      Checksum             : 0xc740
      Urgent Pointer       : 0
    TCP Data
      00 00 00 00 00 00                                    ......

>
```

where *capture_filter* is the name of a packet capture configuration. See Configure packet capture for the network analyzer for more information.

To determine available packet capture configurations, use the **?**:

```
> show anaylzer name ?

name: Name of the capture filter to use.
Format:
  test_capture
  capture_ping

> show anaylzer name
```

Save captured data traffic to a file

Data traffic is captured to RAM and when the device reboots, the data is lost. To retain the captured data, first save the data to a file and then upload the file to a PC.

To save captured traffic data to a file, use the analyzer save command:

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.

2. Type the following at the Admin CLI prompt:

```
> analyzer save filename filename name capture_filter
>
```

where:

- *filename* is the name of the file that the captured data will be saved to.

  Determine filenames already in use:

  Use the tab autocomplete feature to determine filenames that are currently in use:

```
> analyzer save name <tab>
test1_analyzer_capture     test2_analyzer_capture
> analyzer save name
```

- *capture_filter* is the name of a packet capture configuration. See Configure packet capture for the network analyzer for more information.

  To determine available packet capture configurations, use the **?**:

```
> analyzer save name ?

name: Name of the capture filter to use.
Format:
  test_capture
  capture_ping

> analyzer save name
```

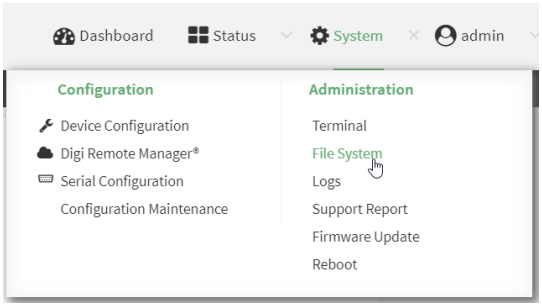
The file is stored in the **/etc/config/analyzer** directory. To transfer the file to your PC, see Download captured data to your PC.

Download captured data to your PC

After saving captured data to a file (see Save captured data traffic to a file), you can download the file from the WebUI or from the command line by using the scp (secure copy file) command.

WebUI

1. Log into the IX15 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Administration**, click **File System**.



The **File System** page appears.



3. Highlight the **analyzer** directory and click ↱ to open the directory.
4. Select the saved analyzer report you want to download and click 🖫 (download).

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Type **scp** to use the Secure Copy program to copy the file to your PC:

```
> scp host hostname-or-ip user username remote remote-path local local-
path to remote
```

   where:

   - *hostname-or-ip* is the hostname or ip address of the remote host.
   - *username* is the name of the user on the remote host.
   - *remote-path* is the location on the remote host where the file will be copied.
   - *local-path* is the path and filename on the IX15 device.

   For example:

   To download the traffic saved in the file **/etc/config/analyzer/eth0.pcpng** to a PC with the IP **192.168.210.2**, for a user named **maria**, to the **/home/maria** directory:

```
> scp host 192.168.210.2 user maria remote /home/maria local
/etc/config/analyzer/eth0.pcpng to remote

maria@192.168.210.2's password:
eth0.pcpng                                    100%   11KB 851.3KB/s
00:00
```

Clear captured data

To clear captured data traffic in RAM, use the analyzer clear command:

Command line

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. Type the following at the Admin CLI prompt:

```
> analyzer clear name capture_filter
>
```

where *capture_filter* is the name of a packet capture configuration. See Configure packet capture for the network analyzer for more information.

To determine available packet capture configurations, use the **?**:

```
> anaylzer clear name ?

name: Name of the capture filter to use.
Format:
  test_capture
  capture_ping

> anaylzer clear name
```

Note You can remove data traffic saved to a file using the rm command.

Use the ping command to troubleshoot network connections

Use the ping command troubleshoot connectivity problems.

Ping to check internet connection

To check your internet connection:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the Admin CLI prompt, type the ping command followed by the host name or IP address of the server to be pinged:

```
> ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=54 time=11.1 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=54 time=10.8 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=54 time=10.7 ms
...
>
```

3. Type **exit** to exit the Admin CLI.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Stop ping commands

To stop pings when the number of pings to send (the **count** parameter) has been set to a high value, enter **Ctrl+C**.

Use the traceroute command to diagnose IP routing problems

Use the **traceroute** command to diagnose IP routing problems. This command traces the route to a remote IP host and displays results. The **traceroute** command differs from ping in that traceroute shows where the route fails, while ping simply returns a single error on failure.

See the traceroute command description for command syntax and examples. The **traceroute** command has several parameters. Only **host** is required.

- **host**: The IP address of the destination host.
- **bypass**: Send directly to a host on an attached network.
- **debug**: Enable socket level debugging.
- **dontfragment**: Do not fragment probe packets.
- **first_ttl**: Specifies with what TTL to start. (Default: 1)
- **gateway**: Route the packet through a specified gateway.
- **icmp**: Use ICMP ECHO for probes.
- **interface**: Specifies the interface.

- **ipchecksums**: Calculate ip checksums.
- **max_ttl**: Specifies the maximum number of hops. (Default: 30)
- **nomap**: Do not map IP addresses to host names
- **nqueries**: Sets the number of probe packets per hop. (Default: 3)
- **packetlen**: Total size of the probing packet. (Default: -1)
- **pausemsecs**: Minimal time interval between probes (Default: 0)
- **port**: Specifies the destination port. (Default: -1)
- **src_addr**: Chooses an alternative source address.
- **tos**: Set Type of Service. (Default: -1)
- **verbose**: Verbose output.
- **waittime**: Max wait for a response to a probe. (Default: 5)

Example

This example shows using **traceroute** to verify that the Digi IX15 Gateway device can route to host **8.8.8.8** (www.google.com) through the default gateway. The command output shows that **15** routing hops were required to reach the host:

1. Log into the IX15 command line as a user with Admin access.

   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **admin** to access the Admin CLI.
2. At the Admin CLI prompt, use the **traceroute** command to view IP routing information:

```
> traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 52 byte packets
 1  192.168.8.1 (192.168.8.1)  0 ms  0 ms  0 ms
 2  10.10.10.10 (10.10.10.10)  0 ms  2 ms  2 ms
 3  * 10.10.8.23 (10.10.8.23)  1 ms  1 ms
 4  96.34.84.22 (96.34.84.22)  1 ms  1 ms  1 ms
 5  96.34.81.190 (96.34.81.190)  2 ms  2 ms  2 ms
 6  * * *
 7  96.34.2.12 (96.34.2.12)  11 ms  11 ms  11 ms
 8  * * *
 9  8.8.8.8 (8.8.8.8)  11 ms  11 ms  11 ms
>
```

By entering a **whois** command on a Unix device, the output shows that the route is as follows:

1. **192/8**: The local network of the IX15 device.
2. **192.168.8.1**: The local network gateway to the Internet.
3. **96/8**: Charter Communications, the network provider.
4. **216/8**: Google Inc.

Stop the traceroute process

To stop the traceroute process, enter **Ctrl-C**.

Server Command Interface (SCI)

This section describes using SCI with the IX15.

Discover and retrieve the XBee network 915
Retrieve XBee device settings 916
Configure XBee device settings 918
Retrieve XBee device settings 918
Execute arbitrary commands 919
Update an XBee device profile 922
Reset an XBee device to factory defaults 923

Discover and retrieve the XBee network

The Digi Remote Manager API allows you to remotely retrieve the XBee network of your IX15. You can retrieve the current discovered network, perform a new discovery or clear the network before discovering it. To do so:

1. **Login** to Digi Remote Manager: remotemanager.digi.com/ui/login.
2. Click the **API Explorer** button located in the left menu.
3. In the right panel, configure the **request type** to be **POST** and use **/ws/sci** as the request **URL**.
4. Fill the **Request Body** field with the following content:

```
<sci_request version="1.0">
  <send_message cache="false">
    <targets>
      <device id="DEVICE_ID"/>
    </targets>
    <rci_request version="1.1">
      <do_command target="zigbee">
        <discover start="1" size="255" option="discover"/>
      </do_command>
    </rci_request>
  </send_message>
</sci_request>
```

   Where:

   - **DEVICE_ID** is the Device ID of the IX15 registered in your Digi Remote Manager account.
   - The **discover attributes** are configured as follows:
     - **addr** attribute specifies the 64-bit address of the XBee device to discover/retrieve. If not specified, a full discover of the network is performed.
     - **start** attribute specifies the starting index of the XBee devices to retrieve.
     - **size** attribute specifies the maximum number of XBee devices to retrieve.
     - **option** attribute specifies the type of discover to execute:
     - **clear** to clear the network cache before performing the discover.
     - **discover** to perform a standard network discover.
     - **current** to retrieve the current network information without running the discover.
5. Click the **Send** button next to the **URL** field.
6. After a while, the **Response Body** field is populated with the response from the IX15 containing a list of devices in the network:

```
<sci_reply version="1.0">
  <send_message>
    <device id="DEVICE_ID">
      <rci_reply version="1.1">
        <do_command target="zigbee">
          <discover>
            <device index="1">
              <type>0</type>
              <ext_addr>00:13:A2:00:41:AF:13:43!</ext_addr>
              <net_addr>0x0000</net_addr>
              <node_id>COORDINATOR</node_id>
```

```
            <hw_version>0x41</hw_version>
            <fw_version>0x100a</fw_version>
          </device>
          <device index="2">
            <type>1</type>
            <ext_addr>00:13:A2:00:41:95:C6:58!</ext_addr>
            <net_addr>0xC93A</net_addr>
            <node_id>ROUTER_1</node_id>
            <hw_version>0x42</hw_version>
            <fw_version>0x100a</fw_version>
          </device>
          <device index="3">
            <type>1</type>
            <ext_addr>00:13:A2:00:41:AE:30:3D!</ext_addr>
            <net_addr>0x87F4</net_addr>
            <node_id>ROUTER_2</node_id>
            <hw_version/>
            <fw_version/>
          </device>
        </discover>
      </do_command>
    </rci_reply>
  </device>
 </send_message>
</sci_reply>
```

Retrieve XBee device settings

The Digi Remote Manager API allows you to remotely retrieve the value of a predefined set of settings for an XBee device in a network. To do so:

1. **Login** to Digi Remote Manager: remotemanager.digi.com/ui/login.
2. Click the **API Explorer** button located in the left menu.
3. In the right panel, configure the **request type** to be **POST** and use **/ws/sci** as the request **URL**.
4. Fill the **Request Body** field with the following content:

```
<sci_request version="1.0">
  <send_message>
    <targets>
      <device id="DEVICE_ID"/>
    </targets>
    <rci_request version="1.1">
      <do_command target="zigbee">
        <query_setting addr="0013A2004195C658"/>
      </do_command>
    </rci_request>
  </send_message>
</sci_request>
```

Where:

- **DEVICE_ID** is the Device ID of the IX15 registered in your Digi Remote Manager account.

- The **query_setting attributes** are configured as follows:
  - **addr** attribute specifies the 64-bit address of the XBee device to retrieve its settings. If omitted, the address of the IX15's XBee device is used.

5. Click the **Send** button next to the **URL** field.
6. After a while, the **Response Body** field is populated with the response from the IX15 containing the list of settings and values for the XBee device:

```
<sci_reply version="1.0">
  <send_message>
    <device id="DEVICE_ID">
      <rci_reply version="1.1">
        <do_command target="zigbee">
          <query_setting addr="0013A2004195C658">
            <radio>
              <aggregation>255</aggregation>
              <api_enable>1</api_enable>
              <api_options>0</api_options>
              <channel>0x17</channel>
              <channel_mask/>
              <coord_enable>0</coord_enable>
              <dest_addr_high>0x00000000</dest_addr_high>
              <dest_addr_low>0x00000000</dest_addr_low>
              <device_options>0x40</device_options>
              <device_type>0x00120000</device_type>
              <dio0_config>1</dio0_config>
              <dio1_config>0</dio1_config>
              <dio2_config>0</dio2_config>
              <dio3_config>0</dio3_config>
              <dio4_config>0</dio4_config>
              <dio5_config>1</dio5_config>
              <dio6_config>0</dio6_config>
              <dio7_config>1</dio7_config>
              <dio8_config>1</dio8_config>
              <dio9_config>1</dio9_config>
              <dio10_config>1</dio10_config>
              <dio11_config>0</dio11_config>
              <dio12_config>0</dio12_config>
              <dio13_config>1</dio13_config>
              <dio14_config>1</dio14_config>
              <dio15_config>0</dio15_config>
              <dio16_config>0</dio16_config>
              <dio17_config>0</dio17_config>
              <dio18_config>0</dio18_config>
              <dio19_config>0</dio19_config>
              <dio_detect>0x0000</dio_detect>
              <dio_sample_rate>0</dio_sample_rate>
              <discover_options>0x00</discover_options>
              <discover_timeout>60</discover_timeout>
              <encrypt_enable>0</encrypt_enable>
              <encrypt_options>0x02</encrypt_options>
              <join_time>255</join_time>
              <join_notification>0</join_notification>
              <join_verification>1</join_verification>
              <link_key>0x00</link_key>
              <network_key>0x00</network_key>
              <net_addr>0xc93a</net_addr>
              <node_id>ROUTER_1</node_id>
```

```
            <pan_id>0x00000000deadbeef</pan_id>
            <power_level>4</power_level>
            <pwm0_config>0</pwm0_config>
            <pwm1_config>0</pwm1_config>
            <scan_channels>0x7fff</scan_channels>
            <scan_duration>3</scan_duration>
            <serial_parity>0</serial_parity>
            <serial_rate>7</serial_rate>
            <serial_stop_bits>0</serial_stop_bits>
            <sleep_count>1</sleep_count>
            <sleep_mode>0</sleep_mode>
            <sleep_options>0x00</sleep_options>
            <sleep_time>32</sleep_time>
            <sleep_wake_time>5000</sleep_wake_time>
          </radio>
        </query_setting>
      </do_command>
    </rci_reply>
  </device>
 </send_message>
</sci_reply>
```

The settings returned by this API execution are a subset of all the settings contained in the XBee firmware. Some devices may not have all these settings available, in that case, the unavailable settings return an empty value. To retrieve the value of a specific setting not contained in this list, see Execute arbitrary commands.

Configure XBee device settings

Retrieve XBee device settings

The Digi Remote Manager API allows you to remotely set the value of a predefined set of settings for an XBee device of your network. To do so:

1. **Login** to Digi Remote Manager: remotemanager.digi.com/ui/login.
2. Click the **API Explorer** button located in the left menu.
3. In the right panel, configure the **request type** to be **POST** and use **/ws/sci** as the request **URL**.
4. Fill the **Request Body** field with the following content:

```
<<sci_request version="1.0">
  <send_message>
    <targets>
      <device id="DEVICE_ID"/>
    </targets>
    <rci_request version="1.1">
      <do_command target="zigbee">
        <set_setting addr="0013A2004195C658">
          <radio>
            <node_id>NEW_NODE_ID_VALUE</node_id>
          </radio>
        </set_setting>
      </do_command>
    </rci_request>
  </send_message>
</sci_request>
```

Where:

- **DEVICE_ID** is the Device ID of the IX15 registered in your Digi Remote Manager account.
- **NEW_NODE_ID_VALUE** is the new value of the **node_id** setting.
- The **set_setting attributes** are configured as follows:
  - **addr** attribute specifies the 64-bit address of the XBee device to set its settings. If omitted, the address of the IX15's XBee device is used.
  - You can add new setting entries to the request inside the **radio** tag following this pattern:

    <SETTING_TAG>NEW_SETTING_VALUE</SETTING_TAG>

    The available tags of the settings that can be changed are listed in the answer of the **query_setting** request, explained in Retrieve XBee device settings.

5. Click the **Send** button next to the **URL** field.
6. After a while, the **Response Body** field is populated with the response from the IX15 with an empty **radio** tag if the request was successful:

```
<sci_reply version="1.0">
  <send_message>
    <device id="DEVICE_ID">
      <rci_reply version="1.1">
        <do_command target="zigbee">
          <set_setting addr="0013A2004195C658">
            <radio/>
          </set_setting>
        </do_command>
      </rci_reply>
    </device>
  </send_message>
</sci_reply>
```

Execute arbitrary commands

You can remotely execute arbitrary commands in an XBee network by using the IX15 API. The command to execute can be one of the following types:

- Read the value of any XBee setting.
- Change the value of any XBee setting.
- Execute a special command in the XBee and get the output.

To remotely execute any arbitrary command in an XBee device from the network using IX15, login to Digi Remote Manager and access the API explorer:

1. **Login** to Digi Remote Manager: remotemanager.digi.com/ui/login.
2. Click the **API Explorer** button located in the left menu.
3. In the right panel, configure the **request type** to be **POST** and use **/ws/sci** as the request **URL**.

Depending on the command to execute, you need to fill the **Request Body** field with different content.

Read the value of an XBee setting

Fill the **Request Body** field with the following content:

```
<sci_request version="1.0">
  <send_message>
    <targets>
      <device id="DEVICE_ID"/>
    </targets>
    <rci_request version="1.1">
      <do_command target="zigbee">
        <radio_command addr="0013A2004195C658" id="NI" format="string"
timeout="10" />
      </do_command>
    </rci_request>
  </send_message>
</sci_request>
```

Where:

- **DEVICE_ID** is the Device ID of the IX15 registered in your Digi Remote Manager account.
- The **radio_command** attributes are configured as follows:
  - **addr** attribute specifies the 64-bit address of the XBee device to read the setting from. If omitted, the address of the IX15's XBee device is used.
  - **id** attribute is the identifier of the setting to read, in this case **NI**.
  - **format** is the format in which the current setting value will be returned:
    - **string** to receive the setting value in string format.
    - **integer** to receive the setting value in integer format.
    - **binary** to receive the setting value in binary format with hexadecimal representation. If the format attribute is omitted, this is the default format used.
  - **timeout** is the time in seconds to wait for answer. If omitted, the configured XBee device read timeout is used.

Click the **Send** button next to the **URL** field.

After a while, the **Response Body** field is populated with the response from the IX15 containing the value of the requested setting:

```
<sci_reply version="1.0">
  <send_message>
    <device id="DEVICE_ID">
      <rci_reply version="1.1">
        <do_command target="zigbee">
          <radio_command addr="0013A2004195C658" format="string"
id="NI">ROUTER_1</radio_command>
        </do_command>
      </rci_reply>
    </device>
  </send_message>
</sci_reply>
```

Set the value of an XBee setting

Fill the **Request Body** field with the following content:

```
<sci_request version="1.0">
  <send_message>
    <targets>
```

```
      <device id="DEVICE_ID"/>
    </targets>
    <rci_request version="1.1">
      <do_command target="zigbee">
        <radio_command addr="0013A2004195C658" id="NI" format="string"
timeout="10">NEW_NODE_NI_VALUE</radio_command>
      </do_command>
    </rci_request>
  </send_message>
</sci_request>
```

Where:

- **DEVICE_ID** is the Device ID of the IX15 registered in your Digi Remote Manager account.
- The **set_setting attributes** are configured as follows:
  - **addr** attribute specifies the 64-bit address of the XBee device to set the setting to. If omitted, the address of the gateway's XBee device is used.
  - **id** attribute is the identifier of the setting to write, in this case **NI**.
  - **format** is the format in which the setting value is written:
    - **string** if the setting value is written in string format.
    - **integer** if the setting value is written in integer format.
    - **binary** if the setting value is written in binary format with hexadecimal representation.
  - **timeout** is the time in seconds to wait for answer. If omitted, the configured XBee device read timeout is used.

Click the **Send** button next to the **URL** field.

After a while, the **Response Body** field populates with the response from the IX15 containing an empty **radio_command** tag if the request was successful:

```
<sci_reply version="1.0">
  <send_message>
    <device id="DEVICE_ID">
      <rci_reply version="1.1">
        <do_command target="zigbee">
          <radio_command addr="0013A2004195C658" format="string" id="NI"/>
        </do_command>
      </rci_reply>
    </device>
  </send_message>
</sci_reply>
```

Execute a special XBee command

Fill the **Request Body** field with the following content:

```
<sci_request version="1.0">
  <send_message>
    <targets>
      <device id="DEVICE_ID"/>
    </targets>
    <rci_request version="1.1">
      <do_command target="zigbee">
        <radio_command addr="0013A2004195C658" id="ND" timeout="10" />
      </do_command>
```

```
    </rci_request>
  </send_message>
</sci_request>
```

Where:

- **DEVICE_ID** is the Device ID of the IX15 registered in your Digi Remote Manager account.
- The **set_setting attributes** are configured as follows:
  - **addr** attribute specifies the 64-bit address of the XBee device to execute the command in. If omitted, the address of the IX15's XBee device is used.
  - **id** attribute is the identifier of the command to execute, in this case **ND**.
  - **timeout** is the time in seconds to wait for answer. If omitted, the configured XBee device read timeout is used.

In this case, there is not **format** attribute and the command output is always returned in **binary** format represented in hexadecimal.

Click the **Send** button next to the **URL** field.

After a while, the **Response Body** field populates with the response from the IX15 containing the result of the command execution:

```
<sci_reply version="1.0">
  <send_message>
    <device id="DEVICE_ID">
      <rci_reply version="1.1">
        <do_command target="zigbee">
          <radio_command addr="0013A2004195C658"
id="ND">0x00000013a20041af134344415649445f434f4f5244494e41544f5200fffe0000c1051
01e</radio_command>
        </do_command>
      </rci_reply>
    </device>
  </send_message>
</sci_reply>
```

Update an XBee device profile

It is possible to remotely update the profile of an XBee device on your netwok using the Digi Remote Manager API. Before updating the profile, you need to make sure that the profile to update is located in the IX15 file system so that it can be applied. To remotely update the profile of an XBee device:

1. **Login** to Digi Remote Manager: remotemanager.digi.com/ui/login.
2. Click the **API Explorer** button located in the left menu.
3. In the right panel, configure the **request type** to be **POST** and use **/ws/sci** as the request **URL**.
4. Fill the **Request Body** field with the following content:

```
<sci_request version="1.0">
  <send_message>
    <targets>
      <device id="DEVICE_ID"/>
    </targets>
    <rci_request version="1.1">
      <do_command target="zigbee">
        <fw_update target="0013A2004195C658" file="/etc/config/xbee-
```

```
profiles/ROUTER_1.xpro"/>
      </do_command>
    </rci_request>
  </send_message>
</sci_request>
```

Where:

- **DEVICE_ID** is the Device ID of the IX15 registered in your Digi Remote Manager account.
- The **fw_update attributes** are configured as follows:
  - **target** attribute specifies the 64-bit address of the XBee device to apply the profile to. If omitted, the address of the IX15's XBee device is used.
  - **file** attribute specifies the full path in the IX15 of the profile to apply to the XBee device.

5. Click the **Send** button next to the **URL** field.
6. After a while, the **Response Body** field will be filled with the response from the IX15 containing an empty **fw_update** tag if the process succeeds:

```
<sci_reply version="1.0">
  <send_message>
    <device id="DEVICE_ID">
      <rci_reply version="1.1">
        <do_command target="zigbee">
          <fw_update target="0013A2004195C658"/>
        </do_command>
      </rci_reply>
    </device>
  </send_message>
</sci_reply>
```

Reset an XBee device to factory defaults

The Digi Remote Manager API allows you to remotely reset any XBee device of a network to its factory default settings. To do so:

1. **Login** to Digi Remote Manager: remotemanager.digi.com/ui/login.
1. Click the **API Explorer** button located in the left menu.
2. In the right panel, configure the **request type** to be **POST** and use **/ws/sci** as the request **URL**.
3. Fill the **Request Body** field with the following content:

```
<sci_request version="1.0">
  <send_message>
    <targets>
      <device id="DEVICE_ID"/>
    </targets>
    <rci_request version="1.1">
      <do_command target="zigbee">
        <set_factory_default addr="0013A2004195C658"/>
      </do_command>
    </rci_request>
  </send_message>
</sci_request>
```

Where:

DEVICE_ID is the Device ID of your XBee Gateway registered in your Digi Remote Manager account.

The **fw_update attributes** are configured as follows:

addr attribute specifies the 64-bit address of the XBee device to reset to factory defaults. If omitted, the address of the gateway's XBee device is used.

Click the **Send** button next to the **URL** field.

After a while, the **Response Body** field will be filled with the response from the IX15 contianing an empty **set_factory_default** tag if the process succeed:

```
<sci_reply version="1.0">
  <send_message>
    <device id="DEVICE_ID">
      <rci_reply version="1.1">
        <do_command target="zigbee">
          <set_factory_default addr="0013A2004195C658"/>
        </do_command>
      </rci_reply>
    </device>
  </send_message>
</sci_reply>
```

Troubleshooting

This section covers common issues and troubleshooting information for the IX15.

System log 926
Recover the local XBee 927

System log

Configure XBee log level

You can configure the logging level of the XBee interface from the **Device Configuration** page:

1. Access the IX15 local web interface.
   a. Use an Ethernet cable to connect the IX15 to your local laptop or PC.
   The factory default IP address is **192.168.2.1**.
   b. Login to the IX15 WebUI as a user with full admin access rights.
   The default user name is **admin** and the default password is the unique password printed on the label packaged with your device.
2. Go to the **Configuration** window.
   a. On the menu, click **System**.
   b. Under **Configuration**, click **Device Configuration**. The Configuration window is displays.
3. Set the log level.
   a. Click **XBee**.
   b. Select the log level from the list. The default is **info**.
4. Click **Apply** to save the configuration and apply the changes.
   The **Apply** button is located at the top of the WebUI page. You may need to scroll to the top of the page to locate it.

Display the system log

To view system log from the WebUI, follow these steps:

1. Access the IX15 local web interface.
   a. Use an Ethernet cable to connect the IX15 to your local laptop or PC.
   The factory default IP address is **192.168.2.1**.
   b. Login to the IX15 WebUI as a user with full admin access rights.
   The default user name is **admin** and the default password is the unique password printed on the label packaged with your device.
2. Go to the Logs window.
   a. On the menu, click **System**.
   b. Under **Administration**, click **Logs**.
3. Under the **System Logs** category you can:
   a. Review the complete log.
   b. Filter by level.
   c. Search for specific entries.
   d. Download the log file.

It is also possible to access the log from the CLI with the **show log** command.

Note For more information, see Configure options for the event and system logs, View system and event logs and Display status and statistics using the show command.

Recover the local XBee

If the local XBee of an IX15 does not respond because, for example, it has damaged firmware or the update process failed, the IX15 automatically tries to restore communication at startup. **To recover a non-responding local XBee, reboot your IX15**.

This recovery process:

1. Tries to restore the communication with the XBee.
2. If it is not possible, the IX15 applies a default profile called **xbee_gateway_default** located at **/etc/config/xbee-profiles/xbee_gateway_default.xpro**.

This default profile programs a Zigbee firmware, restores XBee settings to factory defaults, and applies some settings values, such as the network ID—0x1234—the node identifier—GATEWAY—or the XBee role—coordinator. If this is not suitable, you can:

- Replace this profile with your own, so it is applied if the XBee is not responsive. You can use the WebUI or upload from the CLI.
- Once the recovery process finishes, apply your custom profile manually or programmatically, from the WebUI or upload from the CLI or using the XBee API.

xbeemgmt tool

The **xbeemgmt** tool can also recover the local XBee with a custom profile. This tool is only accessible from the shell:

1. Upload the custom profile to use in the recovery process to the IX15. You can use the WebUI or upload from the CLI.
2. Access the CLI. See Access the command line interface.
3. Disable the XBee service.

```
config xbee enable false
```

4. If you do not have shell access, run the following commands:

```
config auth group admin acl shell enable true
exit
```

5. Depending on the device configuration, you may be presented with another menu, for example:

```
Access selection menu:

    a: Admin CLI
    s: Shell
    q: Quit

Select access or quit [admin] :
```

   Type **s** or **shell** to access the IX15 shell.

6. Enable the XBee interface:

```
# xbee on
```

7. Launch the **xbeemgmt** tool with the port that the XBee is attached to—**/dev/ttyXBee**—and the absolute path of the custom profile to use in the process:

```
# xbeemgmt recover /dev/ttyXBee /etc/config/xbee-profiles/my_custom_
profile.xpro
```

8. When the process finishes, reboot the IX15.

FAQ

This section contains the following topics:

Get the IX15 IP 930
A remote XBee is not listed in the IX15 network 930
PyCharm: My IX15 is not listed in Digi Device Selector 930

Get the IX15 IP

Use the CLI over the serial port to learn the IP assigned to your IX15:

1. Access to the CLI via serial. See Access the command line interface.
2. At the CLI prompt, you can get:

- The Ethernet interface IP:

```
show network interface eth
```

- The Cellular IP:

```
show network interface modem
```

- The default IP:

```
show network interface defaultip
```

Note By default, the default IP address of the device is **192.168.2.1** on the WAN/ETH1 .
The device is also accessible at the default IP address of **192.168.210.1**. However, because this IP address does not use a DHCP server, to connect to this address you must configure your local PC with an appropriate static IP address—for example, **192.168.210.2**.

A remote XBee is not listed in the IX15 network

If a particular remote XBee device is not listed as a known node, you can:

- If the remote XBee is a sleeping node, ensure that it is properly configured. Follow the recommendations in Configure a sleeping network to work with the IX15.
- Perform a network discovery from the WebUI or the CLI.
- Configure active discovery from the WebUI or the CLI.
- Configure the remote XBee to periodically send I/O samples to the IX15. See:
  - XBee 3 Zigbee: Periodic I/O sampling
  - XBee 3 DigiMesh: Periodic I/O sampling
  - XBee 3 802.15.4: Periodic I/O sampling
- Program a MicroPython application on the remote node that sends data to the IX15. See the *Digi MicroPython Programming Guide*.
- For Zigbee, configure the remote XBee to announce itself when joining. See JN (Join Notification).

PyCharm: My IX15 is not listed in Digi Device Selector

If an IX15 does not appear on the list of the Digi Device Selector:

- Ensure that your device has the mDNS service enabled and is on the same network as the computer. See Set up the IX15 for Python development.
- Or click the link **Click here to add it manually** to specify the IP address, port, username, and password. To determine the IX15 IP address, see Get the IX15 IP.


Digi Device Selector
Digi Device Selector
Before running a project or opening an SSH connection, you need to specify the target Digi device.
Select the Digi device you want to communicate with:
Device type: IX15
Device name: Digi IX15 [0004f30e47c0] (ssh)
IP address: 10.101.2.224
169.254.100.100
192.168.2.1
Device type: IX15
Device name: Digi IX15 [0004f40e4924] (ssh)
IP address: 192.168.1.10
Device type: IX15
Device name: Digi IX15 [0004F30E48A0] (ssh)
IP address: 10.101.2.252
Your device is not listed? Click here to add it manually
OK
Cancel

Digi IX15 regulatory and safety statements

This section contains the following topics:

RF exposure statement 933
FCC (USA) exposure notice 933
FCC Part 15 Class A 933
Radio Frequency Interference (RFI) (FCC 15.105) 933
European Community - CE Mark Declaration of Conformity (DoC) 933
Maximum transmit power for radio frequencies 934

RF exposure statement

In order to comply with RF exposure limits established in the ANSI C95.1 standards, the distance between the antenna or antennas and the user should not be less than 20 cm in the United States, 30 cm in Canada, and 23 cm in European countries.

FCC (USA) exposure notice

This equipment complies with FCC radiation exposure limits prescribed for an uncontrolled environment for fixed and mobile use conditions.



This equipment should be installed and operated with a minimum distance of 20 cm between the radiator and the body of the user or nearby persons. This transmitter must not be co-located or operating in conjunction with any other antenna or transmitter except in accordance with FCC procedures and as authorized in the module certification filing.

FCC Part 15 Class A

Radio Frequency Interference (RFI) (FCC 15.105)

The Digi IX15 has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area may cause harmful interference in which case the user will be required to correct the interference at their own expense.

Labeling Requirements (FCC 15.19)

IX15 complies with Part 15 of FCC rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

If the FCC ID is not visible when installed inside another device, then the outside of the device into which the module is installed must also display a label referring to the enclosed module FCC ID.

Modifications (FCC 15.21)

Changes or modifications to this equipment not expressly approved by Digi may void the user's authority to operate this equipment.

European Community - CE Mark Declaration of Conformity (DoC)

Digi has issued Declarations of Conformity for the IX15 concerning emissions, EMC, and safety. For more information, see www.digi.com/resources/certifications.

Important note

Digi customers assume full responsibility for learning and meeting the required guidelines for each country in their distribution market. Refer to the radio regulatory agency in the desired countries of operation for more information.

Maximum transmit power for radio frequencies

The following tables show the maximum transmit power for frequency bands.

Cellular frequency bands

Frequency bands	Maximum transmit power
Cellular LTE 700 MHz Cellular LTE 800 MHz Cellular LTE 850 MHz Cellular LTE 900 MHz Cellular LTE 1700 MHz Cellular LTE 1800 MHz Cellular LTE 1900 MHz Cellular LTE 2100 MHz	200 mW
Cellular LTE 2600 MHz Cellular LTE 2300 MHz Cellular LTE 2500 MHz	158.49 mW

XBee frequency bands

Frequency bands	Maximum transmit power
XBee 3 Zigbee/DigiMesh/802.15.4 RF Modules operate in the ISM 2.4 – 2.4835 GHz frequency band.	XBee 3-Pro 79 mW (+19 dBm) XBee 3 6.3 mW (+8 dBm)

RoHS compliance statement

All Digi International Inc. products that are compliant with the RoHS Directive (EU Directive 2002/95/EC and subsequent amendments) are marked as **RoHS COMPLIANT**. RoHS COMPLIANT means that the substances restricted by the EU Directive 2002/95/EC and subsequent amendments of the European Parliament are not contained in a finished product above threshold limits mandated by EU Directive 2002/95/EC and subsequent amendments, unless the restrictive substance is subject of an exemption contained in the RoHS Directive. Digi International Inc., cannot guarantee that inventory held by distributors or other third parties is RoHS compliant.

ISED (Innovation, Science and Economic Development Canada)

This device complies with Industry Canada license-exempt RSS standard(s). Operation is subject to the following two conditions: (1) this device may not cause interference, and (2) this device must accept any interference, including interference that may cause undesired operation of the device.

Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes: (1) l'appareil ne doit pas produire de brouillage, et (2) l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

RF Exposure



CAUTION! This equipment is approved for mobile and base station transmitting devices only. Antenna(s) used for this transmitter must be installed to provide a separation distance of at least 30 cm from all persons and must not be co-located or operating in conjunction with any other antenna or transmitter.



ATTENTION! Cet équipement est approuvé pour la mobile et la station base dispositifs d'émission seulement. Antenne(s) utilisé pour cet émetteur doit être installé pour fournir une distance de séparation d'au moins 30 cm à partir de toutes les personnes et ne doit pas être situé ou fonctionner en conjonction avec tout autre antenne ou émetteur.

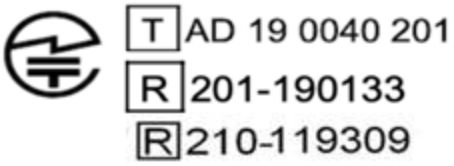
Antennas

See Antenna specifications for the XBee RF Module and Antenna specifications for the cellular modem.

Voir les spécifications d'antenne pour le module RF XBee et les spécifications d'antenne pour le modem cellulaire.

Japan (TELEC)

The IX15 complies with Japan MIC Article 2 Paragraph 1, Item 19.



Safety notices

- Read all instructions before installing and powering the router. You should keep these instructions in a safe place for future reference.
- If the power supply shows signs of damage or malfunction, stop using it immediately, turn off the power and disconnect the power supply before contacting your supplier for a repair or replacement.
- Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment. Use only the accessories, attachments, and power supplies provided by the manufacturer-connecting non-approved antennas or power supplies may damage the router, cause interference or create an electric shock hazard, and will void the warranty.
- Do not attempt to repair the product. The router contains no electronic components that can be serviced or replaced by the user. Any attempt to service or repair the router by the user will void the product warranty.
- Ports that are capable of connecting to other apparatus are defined as SELV ports. To ensure conformity with IEC60950 ensure that these ports are only connected to ports of the same type on other apparatus.

Safety statements



WARNING! For ambient temperatures above 60° C, this equipment must be installed in a Restricted Access Location only.



AVERTISSEMENT! Cet équipement est destiné à être installé dans un lieu d'accès restreint uniquement.

CAUTION! Hot surface!
To avoid burns when handling the device surface, wait at least one half hour after switching off the device before handling the surface.

PRUDENCE! Surface chaude!
Pour éviter les brûlures lors de la manipulation de la surface de l'appareil, attendez au moins une demi-heure après avoir éteint l'appareil avant de manipuler la surface.

Digi IX15 Gateway Hazardous Locations information

Special conditions for safe use

Hazardous locations

- ANSI/ISA UL 121201-2015
- CAN/CSA C22.2 NO.213-17

Ordinary locations

- UL 60950-1, 2nd Edition, 2014-10-14
- UL 62368: 2014, 2nd Edition

Class I Division 2, Groups A,B,C,D Temperature Code: T4



WARNING! The Digi IX15 Gateway is suitable for use in Class I, Division 2, Groups A, B, C, and D or Non-hazardous locations only. These devices are open-type devices that are to be installed in a tool only accessible enclosure suitable for the environment.



WARNING! EXPLOSION HAZARD - DO NOT DISCONNECT WHILE THE CIRCUIT IS LIVE OR UNLESS THE AREA IS FREE OF IGNITIBLE CONCENTRATIONS.



Avertissement: Risque d'Exlposion - Avant de déconnecter l'equipment, couper le courant ou s'assurer que l'emplacement est désigné non dangereux.



WARNING! Antennas intended for use in Class I, Division 2 Hazardous Locations must be installed within the end use enclosure. For remote mounting in an unclassified location, routing and installation of the antennas shall be in accordance with the National Electrical Codes requirements.

Use 24-12 AWG (.2-3.3mm2) with a minimum temperature rating of 80° C (recommended 105° C) for connecting to terminal blocks. Torque all screws from 5 to 8 inches per pound. To connect to Earth Ground, tie a 24-12 AWG (.2-3.3mm2) wire from the -neg DC input to grounded earth.

Power input	12-24VDC, 9W min
Power consumption	Typical 4 W (max 6 W)
Full operating temperature range	-40C to +74C (-40F to +165F)

For additional information on installation and operation of this product, visit www.digi.com/support.

Manufacturer: Digi International Inc. www.digi.com

Special safety notes for wireless routers

Digi International products are designed to the highest standards of safety and international standards compliance for the markets in which they are sold. However, cellular-based products contain radio devices which require specific consideration. Take the time to read and understand the following guidance. Digi International assumes no liability for an end user's failure to comply with these precautions.



Wireless routers incorporate a wireless radio module. Users should ensure that the antenna(s) is (are) positioned at least 1 meter away from themselves and other persons in normal operation.
When in a hospital or other health care facility, observe the restrictions on the use of mobile phones. Do not use the router in areas where guidelines posted in sensitive areas instruct users to switch off mobile phones. Medical equipment may be sensitive to RF energy.
The operation of cardiac pacemakers, other implanted medical equipment and hearing aids can be affected by interference from cellular terminals such as the wireless routers when places close to the device. If in doubt about potential danger, contact the physician or the manufacturer of the device to verify that the equipment is properly shielded. Pacemaker patients are advised to keep the wireless router away from the pacemaker while it is on.



Wireless routers must NOT be operated on aircraft. The operation of wireless appliances in an aircraft is forbidden to prevent interference with communications systems. Failure to observe these instructions may lead to the suspension or denial of cellular services to the offender, legal action, or both.



As with any electrical equipment, do not operate the router in the presence of flammable gases, fumes or potentially explosive atmospheres. Do not use radio devices anywhere that blasting operations occur.



Wireless routers receive and transmit radio frequency energy when power is on. Interference can occur when using the router close to TV sets, radios, computers or inadequately shielded equipment. Follow any special regulations and always power off your router wherever forbidden or when it may cause interference or danger.



SOS IMPORTANT! Wireless routers operate using radio signals and cellular networks cannot be guaranteed to connect in all possible conditions. Therefore, never rely solely upon any wireless device for life critical communications.

Product disposal instructions

The WEEE (Waste Electrical and Electronic Equipment: 2002/96/EC) directive has been introduced to ensure that electrical/ electronic products are recycled using the best available recovery techniques to minimize the impact on the environment.



This product contains high quality materials and components which can be recycled. At the end of its life this product MUST NOT be mixed with other commercial waste for disposal. Check with the terms and conditions of your supplier for disposal information.

Digi International Ltd WEEE Registration number: WEE/HF1515VU

Safety warnings

English
Bulgarian--български
Croatian--Hrvatski
French--Français
Greek--Ελληνικά
Hungarian--Magyar
Italian--Italiano
Latvian--Latvietis
Lithuanian--Lietuvis
Polish--Polskie
Portuguese--Português
Slovak--Slovák
Slovenian--Esloveno
Spanish--Español

English



Ensure that the power cord is connected to a socket-outlet with earthing connection.



To comply with FCC/IC RF exposure limits at least 20 cm separation distance must be maintained between any antenna of the unit and any part of the user at all times.



This appliance does not contain any user-serviceable parts. Never open the equipment. For safety reasons, the equipment should be opened only by qualified personnel.



The unit must be powered off where blasting is in progress, where explosive atmospheres are present, or near medical or life support equipment. Do not power on the unit in any aircraft.



Operation of this equipment in a residential environment could cause radio interference.



For ambient temperatures above 60° C, this equipment must be installed in a Restricted Access Location only.

Bulgarian--български



Уверете се, че захранващият кабел е свързан към контакт със заземителна връзка.



За да се спази FCC / IC границите на излагане на радиочестота, трябва да се поддържа поне 20 cm разстояние на разделяне между която и да е антена на устройството и която и да е част от потребителя по всяко време.



Този уред не съдържа части, които обслужват потребителя. Никога не отваряйте оборудването. От съображения за безопасност оборудването трябва да се отваря само от квалифициран персонал.



Уредът трябва да се изключи там, където се извършва взривяване, където има експлозивна атмосфера или в близост до медицинско оборудване или оборудване за поддържане на живота. Не включвайте устройството в самолет.



Работата с това оборудване в жилищна среда може да причини радиосмущения.



За околни температури над 60 ° C, това оборудване трябва да се инсталира само на място с ограничен достъп.

Croatian--Hrvatski



Provjerite je li kabel za napajanje spojen na utičnicu s uzemljenjem.



Da bi se udovoljilo FCC / IC ograničenjima izlaganja RF, mora se održavati najmanje 20 cm udaljenosti odvojenosti od bilo koje antene uređaja i bilo kojeg dijela korisnika u svakom trenutku.



Ovaj uređaj ne sadrži dijelove koje korisnik može servisirati. Nikada ne otvarajte opremu. Iz sigurnosnih razloga opremu bi trebalo otvarati samo kvalificirano osoblje.



Uređaj se mora isključiti tamo gdje je u tijeku miniranje, gdje su prisutne eksplozivne atmosfere ili u blizini medicinske opreme ili opreme za održavanje života. Nemojte uključivati jedinicu ni u jednom zrakoplovu.



Rad ove opreme u stambenom okruženju mogao bi prouzročiti radio smetnje.



Za okolne temperature iznad 60 ° C, ova oprema mora biti instalirana samo na mjestu s ograničenim pristupom.

French--Français



Assurez-vous que le cordon d'alimentation est connecté à une prise de courant avec mise à la terre.



Pour se conformer aux limites d'exposition RF FCC/IC, une distance de séparation d'au moins 20 cm doit être maintenue entre toute antenne de l'unité et toute partie de l'utilisateur à tout moment.



Cet appareil ne contient aucune pièce réparable par l'utilisateur. Ne jamais ouvrir l'équipement. Pour des raisons de sécurité, l'équipement ne doit être ouvert que par du personnel qualifié.



L'unité doit être éteinte là où le dynamitage est en cours, où des atmosphères explosives sont présentes, ou à proximité d'équipements médicaux ou de survie. N'allumez pas l'appareil dans un avion.



L'utilisation de cet équipement dans un environnement résidentiel peut provoquer des interférences radio.



Pour des températures ambiantes supérieures à 60 °C, cet équipement doit être installé uniquement dans un emplacement à accès restreint.

Greek--Ελληνικά



Βεβαιωθείτε ότι το καλώδιο τροφοδοσίας είναι συνδεδεμένο σε πρίζα με σύνδεση γείωσης.



Για συμμόρφωση με τα FCC / IC RF όρια έκθεσης πρέπει να διατηρείται τουλάχιστον 20 cm απόσταση διαχωρισμού μεταξύ οποιασδήποτε κεραίας της μονάδας και οποιουδήποτε μέρους του χρήστη ανά πάσα στιγμή.



Αυτή η συσκευή δεν περιέχει εξαρτήματα που μπορούν να επισκευαστούν από το χρήστη. Μην ανοίγετε ποτέ τον εξοπλισμό. Για λόγους ασφαλείας, ο εξοπλισμός πρέπει να ανοίγει μόνο από εξειδικευμένο προσωπικό.



Η μονάδα πρέπει να είναι απενεργοποιημένη όταν βρίσκεται σε εξέλιξη η έκρηξη, όπου υπάρχουν εκρηκτικές ατμόσφαιρες ή κοντά σε ιατρικό εξοπλισμό ή εξοπλισμό υποστήριξης της ζωής. Μην ενεργοποιείτε τη μονάδα σε κανένα αεροσκάφος.



Η λειτουργία αυτού του εξοπλισμού σε οικιστικό περιβάλλον μπορεί να προκαλέσει παρεμβολές ραδιοφώνου.



Για θερμοκρασίες περιβάλλοντος άνω των 60 ° C, αυτός ο εξοπλισμός πρέπει να εγκατασταθεί μόνο σε θέση περιορισμένης πρόσβασης

Hungarian--Magyar



Győződjön meg arról, hogy a tápkábel csatlakozik egy földelő csatlakozóaljzathoz.



Az FCC / IC rádiófrekvenciás expozíciós határértékeinek betartása érdekében a berendezés bármely antennája és a felhasználó bármely része között legalább 20 cm távolságot kell tartani.



Ez a készülék nem tartalmaz a felhasználó által javítható alkatrészeket. Soha ne nyissa ki a berendezést. Biztonsági okokból a berendezést csak szakképzett személyzet nyithatja meg.



Az egységet ki kell kapcsolni, ha robbantás folyik, ahol robbanásveszélyes környezet van, vagy orvosi vagy életmentő berendezések közelében. Semmilyen repülőgépen ne kapcsolja be az egységet.



A berendezés lakókörnyezetben történő működtetése rádiózavarokat okozhat.



60 ° C feletti környezeti hőmérséklet esetén ezt a berendezést csak korlátozott hozzáférésű helyre kell telepíteni.



Az EZ04-IAG4-EXT és EZ04-IA00-EXT készletekhez mellékelt kiterjesztett hőmérsékletű, dugaszolható tápegység (76002079 /24000141) nem C1D2 tanúsítvánnyal rendelkezik, és nem használható C1D2 besorolású veszélyes helyeken.

Italian--Italiano



Assicurarsi che il cavo di alimentazione sia collegato ad una presa con messa a terra.



Per rispettare i limiti di esposizione RF FCC/IC è necessario mantenere sempre una distanza di separazione di almeno 20 cm tra qualsiasi antenna dell'unità e qualsiasi parte dell'utente.



Questo apparecchio non contiene parti riparabili dall'utente. Non aprire mai l'apparecchiatura. Per motivi di sicurezza, l'apparecchiatura deve essere aperta solo da personale qualificato.



L'unità deve essere spenta dove sono in corso esplosioni, dove sono presenti atmosfere esplosive o vicino ad apparecchiature mediche o di supporto vitale. Non accendere l'unità in nessun aereo.



Il funzionamento di questa apparecchiatura in un ambiente residenziale potrebbe causare interferenze radio.



Per temperature ambiente superiori a 60° C, questa apparecchiatura deve essere installata solo in un luogo ad accesso limitato.

Latvian--Latvietis



Pārliecinieties, ka strāvas vads ir pievienots kontaktligzdai ar zemējuma savienojumu.



Lai ievērotu FCC / IC radiofrekvenču iedarbības robežas, vienmēr jābūt vismaz 20 cm attālumam starp jebkuru ierīces antenu un jebkuru lietotāja daļu.



Šajā ierīcē nav nevienas lietotāja apkalpojamas daļas. Nekad neatveriet aprīkojumu. Drošības apsvērumu dēļ aprīkojumu drīkst atvērt tikai kvalificēts personāls.



Iekārtai jābūt izslēgtai, ja notiek spridzināšana, sprādzienbīstama vide vai medicīnas vai dzīvības uzturēšanas aprīkojuma tuvumā. Nevienā lidmašīnā neieslēdziet ierīci.



Šīs ierīces darbība dzīvojamā vidē var izraisīt radio traucējumus.



Ja apkārtējā temperatūra pārsniedz 60 ° C, šī iekārta jāuzstāda tikai ierobežotas piekļuves vietā.

Lithuanian--Lietuvis



Įsitikinkite, kad maitinimo laidas yra prijungtas prie lizdo su įžeminimu.



Kad būtų laikomasi FCC / IC radijo dažnių apšvitos ribų, tarp bet kurios įrenginio antenos ir bet kurios vartotojo dalies visada turi būti išlaikytas bent 20 cm atstumas.



Šiame prietaise nėra naudotojui prižiūrimų dalių. Niekada neatidarykite įrangos. Saugumo sumetimais įrangą turėtų atidaryti tik kvalifikuotas personalas.



Įrenginys turi būti išjungtas ten, kur vyksta sprogdinimas, sprogi aplinka arba šalia medicinos ar gyvybės palaikymo įrangos. Neįjunkite įrenginio jokiuose orlaiviuose.



Naudojant šią įrangą gyvenamojoje aplinkoje, gali kilti radijo trukdžių.



Esant aukštesnei nei 60 ° C aplinkos temperatūrai, ši įranga turi būti montuojama tik riboto patekimo vietoje.

Polish--Polskie



Upewnij się, że przewód zasilający jest podłączony do gniazdka z uziemieniem.



Aby zachować zgodność z limitami ekspozycji FCC/IC RF, między anteną urządzenia a jakąkolwiek częścią użytkownika musi być zachowana odległość co najmniej 20 cm.



To urządzenie nie zawiera żadnych części, które mogą być naprawiane przez użytkownika. Nigdy nie otwieraj urządzenia. Ze względów bezpieczeństwa urządzenie powinno być otwierane wyłącznie przez wykwalifikowany personel.



Urządzenie musi być wyłączone w miejscach, w których trwają prace wybuchowe, w atmosferze wybuchowej lub w pobliżu sprzętu medycznego lub podtrzymującego życie. Nie włączaj urządzenia w żadnym samolocie.



Praca tego sprzętu w środowisku mieszkalnym może powodować zakłócenia radiowe.



W przypadku temperatur otoczenia powyżej 60°C urządzenie to należy instalować wyłącznie w miejscach o ograniczonym dostępie.

Portuguese--Português



Certifique-se de que o cabo de alimentação esteja conectado a uma tomada com conexão de aterramento.



Para cumprir os limites de exposição à RF da FCC / IC, pelo menos 20 cm de distância de separação deve ser mantida entre qualquer antena da unidade e qualquer parte do usuário o tempo todo.



Este aparelho não contém peças cuja manutenção possa ser feita pelo usuário. Nunca abra o equipamento. Por razões de segurança, o equipamento deve ser aberto apenas por pessoal qualificado.



A unidade deve ser desligada onde houver detonações em andamento, onde houver presença de atmosferas explosivas ou próximo a equipamentos médicos ou de suporte à vida. Não ligue a unidade em nenhuma aeronave.



A operação deste equipamento em um ambiente residencial pode causar interferência de rádio.



Para temperaturas ambientes acima de 60 ° C, este equipamento deve ser instalado apenas em locais de acesso restrito.

Slovak--Slovák



Uistite sa, že je napájací kábel pripojený k zásuvke so zemniacim pripojením.



Aby boli dodržané limity vystavenia vysokofrekvenčným lúčom FCC / IC, musí byť medzi anténou jednotky a akoukoľvek časťou používateľa neustále udržiavaná vzdialenosť najmenej 20 cm.



Toto zariadenie neobsahuje žiadne diely opraviteľné používateľom. Nikdy neotvárajte zariadenie. Z bezpečnostných dôvodov by malo zariadenie otvárať iba kvalifikovaný personál.



Jednotka musí byť vypnutá tam, kde prebiehajú trhacie práce, kde je prítomné výbušné prostredie, alebo v blízkosti lekárskych prístrojov alebo zariadení na podporu života. Jednotku nezapínajte v žiadnom lietadle.



Prevádzka tohto zariadenia v obytnom prostredí by mohla spôsobiť rádiové rušenie.



Pri teplotách okolia nad 60 ° C musí byť toto zariadenie inštalované iba na mieste s obmedzeným prístupom.

Slovenian--Esloveno



Prepričajte se, da je napajalni kabel priključen v vtičnico z ozemljitvenim priključkom.



Da bi izpolnili omejitve izpostavljenosti FCC / IC RF, mora biti med katero koli anteno enote in katerim koli delom uporabnika ves čas vzdrževana najmanj 20 cm razdalja.



Ta naprava ne vsebuje nobenih delov, ki bi jih lahko uporabljal uporabnik. Nikoli ne odpirajte opreme. Iz varnostnih razlogov naj opremo odpira samo usposobljeno osebje.



Enoto je treba izklopiti tam, kjer poteka razstreljevanje, kjer so prisotne eksplozivne atmosfere ali v bližini medicinske opreme ali opreme za vzdrževanje življenja. Enote ne vklopite v nobenem letalu.



Delovanje te opreme v stanovanjskem okolju lahko povzroči radijske motnje.



Pri temperaturah okolice nad 60 ° C mora biti ta oprema nameščena samo na lokaciji z omejenim dostopom.

Spanish--Español



Asegúrese de que el cable de alimentación esté conectado a una toma de corriente con conexión a tierra.



Para cumplir con los límites de exposición a RF de la FCC / IC, se debe mantener una distancia de separación de al menos 20 cm entre cualquier antena de la unidad y cualquier parte del usuario en todo momento.



Este aparato no contiene ninguna pieza que pueda reparar el usuario. Nunca abra el equipo. Por razones de seguridad, el equipo debe ser abierto únicamente por personal calificado.



La unidad debe estar apagada donde se estén realizando explosiones, cuando haya atmósferas explosivas o cerca de equipos médicos o de soporte vital. No encienda la unidad en ningún avión.



El funcionamiento de este equipo en un entorno residencial puede provocar interferencias de radio.



Para temperaturas ambiente superiores a 60 ° C, este equipo debe instalarse únicamente en una ubicación de acceso restringido.

Digi IX15 certifications

International EMC (Electromagnetic Compatibility) and safety standards

This product complies with the requirements of the following Electromagnetic Compatibility standards.

There are no user-serviceable parts inside the product. Contact your Digi representative for repair information.

Certification category	Standards
Electromagnetic Compatibility (EMC) compliance standards	■ EN 301 489-1 V2.1.1 (2017-02) ■ EN 301 489-52 V1.1.0:2016 ■ EN 301 489-17 V3.1.1(2017-02)
Safety compliance standards	ANSI/ISA 121201 CAN/CSA C22.2 No213-17 Class I, Division 2, Groups A, B, C, and D Temperature Code – T4 EN 62368-1:2014
E-UTRA CA, E-UTRA FDD, E-UTRA TDD, UMTS FDD	PTCRB
Cellular carriers	See the current list of carriers on the IX15 Certifications page.
Electrical safety compliance	The IX15 shall be powered using a DC power source Approved in its country of use as per ES1 [IEC 62368-1:2014(Ed.2.0)] or SELV [Safety Extra Low Voltage as per IEC 60950-1:2005(ED 2) + A1, A2.