IX10

User Guide

User Guide

Firmware version 21.5

Revision history—90002399

Revision	Date	Description
A	October 2020	Initial release of the *IX10 User Guide*.

Revision	Date	Description
B	December 2020	Release of Digi IX10 firmware version 20.11: ■ Modem firmware update commands added to the Admin CLI. ■ Network bridging enhanced to use the MAC address of the first active device listed in **Network** > **Bridges** > ***Bridge name*** > **Devices** as the MAC address for the bridged interface. ■ IPsec enhancements: • Support for full IPsec IPv6 tunnels: IPv6-over-IPv4, or IPv4-over-IPv6 tunnels. • IPsec tunnels are now treated like network interfaces, which allows tunnels to be selected for routing and routing priority and access control lists. • IPSec tunnels now wait for Surelink tests, if configured, to pass prior to initiating outbound tunnels. ■ Modbus gateway enhancements: • **Status** > **Modbus gateway** page in the Web UI displays information about modbus clients and servers connected to the gateway. • **show modbus** command displays information about the the Modbus gateway service. ■ Serial port enhancements: • Serial port logging options added for starting, stopping, and viewing serial port activity. • Option added to the **System** > **Serial configuration** page to copy serial port settings. ■ Policy-based routing enhancements: • Added a **DSCP** option to match the routing rule by the type of DSCP field in the packet. • Added a **Defaultroute** option for matching policy-based routes to the device's active default route. ■ Python pip support for installing external modules and libraries. ■ Link speed and duplex options added to Ethernet port configuration. ■ ssh and telnet commands added to Admin CLI. ■ **Services** > **Ping responder** allows you to control the interfaces and firewall zones on which the DAL device will respond to ICMP requests. ■ Enhanced policy-based routing:

Revision	Date	Description
		• Added a **DSCP** option to match the routing rule by the type of DSCP field in the packet. • Added a **Defaultroute** option for matching policy-based routes to the device's active default route. ■ Added a link to User Guide under the User menu in the Web UI.
C	March 2021	Release of Digi IX10 firmware version 21.2: ■ Location services added, including: • The ability to define a static latitude and longitude as a location for the device. • Reporting location information as health metrics to Digi Remote Manager. • Geo-fencing: Allow you to define one or more circular or polygonal geo-fence areas and then perform a set of actions when the device enters or leaves that area. • Python support for location information through the digidevice.location python module. ■ Cellular modem carrier scanning and locking: • New modem scan CLI command for listing available carriers for the current modem and SIM. • Manual carrier selection option to allow you to lock the SIM to a specific carrier. ■ Enhanced serial support: • Certificate management control for TCP and autoconnect serial port setups. • Autoconnect. ■ Local REST API for automated configuration of the device. ■ Support for remote CLI commands through Digi Remote Manager. ■ Support for automatically checking for device and modem firmware updates.

Revision	Date	Description
D	June 2021	Release of Digi IX10 firmware version 21.5: ■ Wi-Fi enhancements: • Added support for WPA3 Wi-Fi encryption: ◦ WPA2/WPA3 Personal ◦ WPA3 Enhanced Open ◦ WPA3 Personal • Added support for WPA and WPA/WPA2 mised mode with TKIP. ■ Cellular enhancements: • Added support for modem firmware update to the Admin CLI. • Added support for over-the-air (OTA) modem firmware update to check, list, and update to new modem firmware from the Digi firmware server. • Added the ability to scan for cellular carriers on the **Modem status** page and the ability select a particular PLMN/network to use. ■ Added comands for over-the-air (OTA) system firmware update to check, list, and update to new firmware from the Digi firmware server. ■ Added a **show dns** command to the Admin CLI to display active DNS servers and their associated interface. ■ Added a **show ntp** command to the Admin CLI to display the status of the NTP service. ■ Expanded Port forwarding option to support a range of ports, including one-to-one and many-to-one port mappings. ■ Added options to control packet filtering for the network analyzer. ■ VPN enhancements: • IPsec enhancements: ◦ Added support for multiple remote endpoints and the ability to use round-robin or to randomly select an endpoint to establish a tunnel to. ◦ Added configurable options to control IKE transmit interval, tunnel retry interval, and tunnel retry timeout. • LDAP enhancements: ◦ Added a login attribute to provide the ability to

Revision	Date	Description
		match the attribute set on an Active Directory server. ■ SureLink enhancements • Added the ability to configure how many times a SureLink test must run, and must fail, before the interface is restarted or the device is rebooted. • Added the ability to configure how many times a SureLink test must pass before an interface is considered to be working. • Added the ability to test another interface's status. ■ SNMPv2 supported added. ■ Simple Certificate Enrollment Protocol (SCEP) supported added. ■ Updated python to version 3.6.13. ■ Added the default **digi.device** local domain.

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.

© 2021 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 (IX10 User Guide, 90002399 D) in the subject line of your email.

Contents

Revision history—90002399 2

What's new in Digi IX10 version 21.5

Digi IX10 Quick start

Step 1: What's in the box 18
Step 2: Gather accessories 18
Step 3: Connect 19
Step 4: Configure 21

Digi IX10 hardware reference

Digi IX10 features and specifications 22
IX10 accessories 22
IX10 front and side views 22
IX10 LEDs 23
- Power (PWR) 24
- SIM 24
- LTE 24
- Signal quality indicators 25
- Ethernet Link and Activity 25
- Signal quality bars explained 26

IX10 power supply requirements 27
Digi IX10 serial connector pinout 27
- 10-pin serial cabling options 28

Hardware setup

Install SIM cards 30
- SIM removal 30
- Tips for improving cellular signal strength 30

Connect data cables 31
Mount the IX10 device 31
- Attach to a mounting surface by using the mounting tabs 31
- Attach to DIN rail with clip 31

Configuration and management

Review IX10 default settings 35
Local WebUI 35
Digi Remote Manager 35
Default interface configuration 35
Other default configuration settings 36
Change the default password for the admin user 36
Configuration methods 38
Using Digi Remote Manager 39
Access Digi Remote Manager 39
Using the web interface 39
Log out of the web interface 40
Using the command line 41
Access the command line interface 41
Log in to the command line interface 41
Exit the command line interface 42

Interfaces

Wireless Wide Area Networks (WWANs) 44
Configure SureLink active recovery to detect modem failures 44
Configure the device to reboot when a failure is detected 51
Disable SureLink 58
Using cellular modems in a Wireless WAN (WWAN) 62
Configure a Wireless Wide Area Network (WWAN) 80
Show WWAN status and statistics 90
Delete a WWAN. 91
Local Area Networks (LANs) 94
About Local Area Networks (LANs) 95
Configure a LAN 95
Show LAN status and statistics 101
Delete a LAN 103
DHCP servers 104
Create a Virtual LAN (VLAN) route 120

Serial port

Configure the serial port 123
Configure UDP serial mode 135
Show serial status and statistics 138
Log serial port messages 138

Routing

IP routing 141
Configure a static route 142
Delete a static route 145
Policy-based routing 146
Configure a routing policy 147
Routing services 154
Configure routing services 154
Show the routing table 157

Dynamic DNS ... 158
Configure dynamic DNS ... 158
Virtual Router Redundancy Protocol (VRRP) ... 163
VRRP+ ... 164
Configure VRRP ... 164
Configure VRRP+ ... 167
Example: VRRP/VRRP+ configuration ... 175
Configure device one (master device) ... 175
Configure device two (backup device) ... 180
Show VRRP status and statistics ... 186

Virtual Private Networks (VPN)

IPsec ... 189
IPsec data protection ... 189
IPsec modes ... 189
Internet Key Exchange (IKE) settings ... 189
Authentication ... 190
Configure an IPsec tunnel ... 190
Configure IPsec failover ... 213
Configure SureLink active recovery for IPsec ... 216
Show IPsec status and statistics ... 223
Debug an IPsec configuration ... 224
Configure a Simple Certificate Enrollment Protocol client ... 225
Example: SCEP client configuration with Fortinet SCEP server ... 229
OpenVPN ... 235
Configure an OpenVPN server ... 236
Configure an OpenVPN Authentication Group and User ... 245
Configure an OpenVPN client by using an .ovpn file ... 249
Configure an OpenVPN client without using an .ovpn file ... 252
Configure SureLink active recovery for OpenVPN ... 256
Show OpenVPN server status and statistics ... 264
Show OpenVPN client status and statistics ... 265
Generic Routing Encapsulation (GRE) ... 267
Configuring a GRE tunnel ... 267
Show GRE tunnels ... 272
Example: GRE tunnel over an IPSec tunnel ... 273
NEMO ... 288
Configure a NEMO tunnel ... 288
Show NEMO status ... 293

Services

Allow remote access for web administration and SSH ... 296
Configure the web administration service ... 300
Configure SSH access ... 310
Use SSH with key authentication ... 316
Generating SSH key pairs ... 316
Configure telnet access ... 319
Configure DNS ... 324
Show DNS server ... 330
Simple Network Management Protocol (SNMP) ... 331
SNMP Security ... 331
Configure Simple Network Management Protocol (SNMP) ... 331

Download MIBs ... 336
Location information ... 337
Configure the location service ... 338
Configure the device to use a user-defined static location ... 340
Configure the device to accept location messages from external sources ... 342
Forward location information to a remote host ... 346
Configure geofencing ... 352
Show location information ... 364
Modbus gateway ... 365
Configure the Modbus gateway ... 366
Show Modbus gateway status and statistics ... 379
System time ... 382
Configure the system time ... 382
Network Time Protocol ... 385
Show status and statistics of the NTP server ... 385
Configure the device as an NTP server ... 386
Configure a multicast route ... 391
Enable service discovery (mDNS) ... 393
Use the iPerf service ... 397
Example performance test using iPerf3 ... 401
Configure the ping responder service ... 402
Example performance test using iPerf3 ... 406

Applications

Configure applications to run automatically ... 408
Task one: Upload the application ... 408
Task two: Configure the application to run automatically ... 410
Run a Python application at the shell prompt ... 414
Start an interactive Python session ... 416
Digidevice module ... 418
Use digidevice.cli to execute CLI commands ... 419
Use digidevice.datapoint to upload custom datapoints to Digi Remote Manager ... 420
Use digidevice.config for device configuration ... 422
Use Python to respond to Digi Remote Manager SCI requests ... 425
Use digidevice runtime to access the runtime database ... 433
Use Python to upload the device name to Digi Remote Manager ... 435
Use Python to access the device location data ... 437
Use Python to send and receive SMS messages ... 441
Use Python to access serial ports ... 444
Use the Paho MQTT python library ... 445
Use the local REST API to configure the IX10 device ... 448
Use the GET method to return device configuration information ... 448
Use the POST method to modify device configuration parameters and add items to a list array ... 450
Use the DELETE method to remove items from a list array ... 451
Stop a script that is currently running ... 452
Show script information ... 452

User authentication

IX10 user authentication ... 455
User authentication methods ... 455
Add a new authentication method ... 457

Delete an authentication method ... 459
Rearrange the position of authentication methods ... 461
Authentication groups ... 463
Change the access rights for a predefined group ... 465
Add an authentication group ... 467
Delete an authentication group ... 471
Local users ... 473
Change a local user's password ... 474
Configure a local user ... 476
Delete a local user ... 483
Terminal Access Controller Access-Control System Plus (TACACS+) ... 486
TACACS+ user configuration ... 487
TACACS+ server failover and fallback to local authentication ... 488
Configure your IX10 device to use a TACACS+ server ... 488
Remote Authentication Dial-In User Service (RADIUS) ... 492
RADIUS user configuration ... 493
RADIUS server failover and fallback to local configuration ... 493
Configure your IX10 device to use a RADIUS server ... 494
LDAP ... 498
LDAP user configuration ... 499
LDAP server failover and fallback to local configuration ... 500
Configure your IX10 device to use an LDAP server ... 500
Configure serial authentication ... 505
Disable shell access ... 507
Set the idle timeout for IX10 users ... 509
Example user configuration ... 512
Example 1: Administrator user with local authentication ... 512
Example 2: RADIUS, TACACS+, and local authentication for one user ... 514

Firewall

Firewall configuration ... 522
Create a custom firewall zone ... 522
Configure the firewall zone for a network interface ... 524
Delete a custom firewall zone ... 526
Port forwarding rules ... 527
Configure port forwarding ... 527
Delete a port forwarding rule ... 532
Packet filtering ... 535
Configure packet filtering ... 535
Enable or disable a packet filtering rule ... 539
Delete a packet filtering rule ... 541
Configure custom firewall rules ... 543
Configure Quality of Service options ... 544

System administration

Review device status ... 556
Configure system information ... 557
Update system firmware ... 559
Manage firmware updates using Digi Remote Manager ... 559
Certificate management for firmware images ... 560
Downgrading ... 560
Dual boot behavior ... 563

Update cellular module firmware 565
Update modem firmware over the air (OTA) 565
Update modem firmware by using a local firmware file 567
Reboot your IX10 device 568
Reboot your device immediately 568
Schedule reboots of your device 569
Erase device configuration and reset to factory defaults 571
Configure the IX10 device to use custom factory default settings 574
Configuration files 576
Save configuration changes 576
Save configuration to a file 577
Restore the device configuration 578
Schedule system maintenance tasks 581
Disable device encryption 588
Re-enable cryptography after it has been disabled. 589
Configure the speed of your Ethernet port 591

Monitoring

intelliFlow 595
Enable intelliFlow 595
Use intelliFlow to display average CPU and RAM usage 598
Use intelliFlow to display top data usage information 599
Use intelliFlow to display data usage by host over time 601
Configure NetFlow Probe 602

Central management

Digi Remote Manager support 608
Configure Digi Remote Manager 608
Collect device health data and set the sample interval 615
Log into Digi Remote Manager 618
Use Digi Remote Manager to view and manage your device 619
Add a device to Digi Remote Manager 620
View Digi Remote Manager connection status 620
Configure multiple devices using profiles 621
Learn more 621

File system

The IX10 local file system 623
Display directory contents 623
Create a directory 624
Display file contents 625
Copy a file or directory 625
Move or rename a file or directory 626
Delete a file or directory 627
Upload and download files 628
Upload and download files by using the WebUI 628
Upload and download files by using the Secure Copy command 629
Upload and download files using SFTP 630

Diagnostics

Generate a support report 633
View system and event logs 634
- View System Logs 634
- View Event Logs 636

Configure syslog servers 638
Configure options for the event and system logs 641
Analyze network traffic 646
- Configure packet capture for the network analyzer 647
- Example filters for capturing data traffic 656
- Capture packets from the command line 657
- Stop capturing packets 658
- Show captured traffic data 659
- Save captured data traffic to a file 660
- Download captured data to your PC 661
- Clear captured data 662

Use the ping command to troubleshoot network connections 664
- Ping to check internet connection 664
- Stop ping commands 664

Use the traceroute command to diagnose IP routing problems 664

Digi IX10 regulatory and safety statements

RF exposure statement 666
Federal Communication (FCC) Part 15 Class B 666
- Radio Frequency Interference (RFI) (FCC 15.105) 666

European Community - CE Mark Declaration of Conformity (DoC) 666
Maximum transmit power for radio frequencies 668
Innovation, Science, and Economic Development Canada (IC) certifications 668
RoHS compliance statement 669
Safety statements 669
Special safety notes for wireless routers 670
Product disposal instructions 671

DigiIX10 Certifications

International EMC (Electromagnetic Compatibility) and safety standards 671

Command line interface

Access the command line interface 674
Log in to the command line interface 674
Exit the command line interface 675
Execute a command from the web interface 675
Display help for commands and parameters 676
- The help command 676
- The question mark (?) command 676
- Display help for individual commands 677
- Use the Tab key or the space bar to display abbreviated help 677

Auto-complete commands and parameters 678
Available commands 679
Use the scp command 680

Display status and statistics using the show command ... 681
show config ... 681
show system ... 682
show network ... 682
Device configuration using the command line interface ... 682
Execute configuration commands at the root Admin CLI prompt ... 683
Display help for the config command from the root Admin CLI prompt ... 683
Configuration mode ... 685
Enable configuration mode ... 685
Enter configuration commands in configuration mode ... 685
Save changes and exit configuration mode ... 686
Exit configuration mode without saving changes ... 686
Configuration actions ... 686
Display command line help in configuration mode ... 687
Move within the configuration schema ... 689
Manage elements in lists ... 690
The revert command ... 692
Enter strings in configuration commands ... 694
Example: Create a new user by using the command line ... 694
Command line reference ... 697
analyzer ... 698
cp ... 699
help ... 700
ls ... 701
mkdir ... 702
modem ... 703
modem puk status [imei STRING] [name STRING] ... 708
modem scan [imeiSTRING] [nameSTRING] ... 709
more ... 711
mv ... 712
ping ... 713
reboot ... 715
rm ... 716
scp ... 717
show ... 718
ssh ... 725
system ... 727
traceroute ... 731

What's new in Digi IX10 version 21.5

Release of Digi IX10 firmware version 21.5:

- Wi-Fi enhancements:
  - Added support for WPA3 Wi-Fi encryption:
    - WPA2/WPA3 Personal
    - WPA3 Enhanced Open
    - WPA3 Personal
  - Added support for WPA and WPA/WPA2 mised mode with TKIP.
- Cellular enhancements:
  - Added support for modem firmware update to the Admin CLI.
  - Added support for over-the-air (OTA) modem firmware update to check, list, and update to new modem firmware from the Digi firmware server.
  - Added the ability to scan for cellular carriers on the **Modem status** page and the ability select a particular PLMN/network to use.
- Added comands for over-the-air (OTA) system firmware update to check, list, and update to new firmware from the Digi firmware server.
- Added a **show dns** command to the Admin CLI to display active DNS servers and their associated interface.
- Added a **show ntp** command to the Admin CLI to display the status of the NTP service.
- Expanded Port forwarding option to support a range of ports, including one-to-one and many-to-one port mappings.
- Added options to control packet filtering for the network analyzer.
- VPN enhancements:
  - IPsec enhancements:
    - Added support for multiple remote endpoints and the ability to use round-robin or to randomly select an endpoint to establish a tunnel to.
    - Added configurable options to control IKE transmit interval, tunnel retry interval, and tunnel retry timeout.
  - LDAP enhancements:
    - Added a login attribute to provide the ability to match the attribute set on an Active Directory server.
- SureLink enhancements

    - Added the ability to configure how many times a SureLink test must run, and must fail, before the interface is restarted or the device is rebooted.
    - Added the ability to configure how many times a SureLink test must pass before an interface is considered to be working.
    - Added the ability to test another interface's status.
- SNMPv2 supported added.
- Simple Certificate Enrollment Protocol (SCEP) supported added.
- Updated python to version 3.6.13.
- Added the default **digi.device** local domain.

Digi IX10 Quick start

The following steps guide you through the setup of your DigiIX10 device.

Step 1: What's in the box

When you open the IX10 package, look for the following:

- **Digi IX10 device**



  The Digi IX10 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 IX10 also includes a terminal connector for the power supply installed in the power input.
- **Insert cards**
- **Digi IX10 label**

  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

Accessories may be purchased with the IX10 device, or you can provide your own.

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



Antennas
Two cellular antennas, ordered separately.



Power supply
Ordered separately.



DIN rail clip
Ordered separately. For optionally clipping the IX10 to a DIN rail.



Laptop or personal computer
Use an Ethernet cable to connect your IX10 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 APN (Access Point Name), and SIM pin (if any) for each card.



Ethernet cable



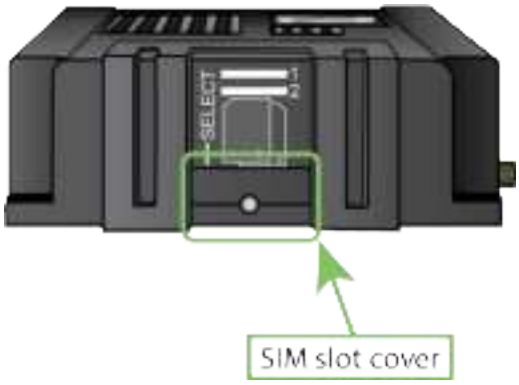
Smart phone or tablet
Optional: Use a smart phone or table to to automatically register your IX10 in your Digi Remote Manager account and connect to your cellular network. See Digi IX10 Quick start.

Step 3: Connect

1. **Insert SIM card(s)**

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

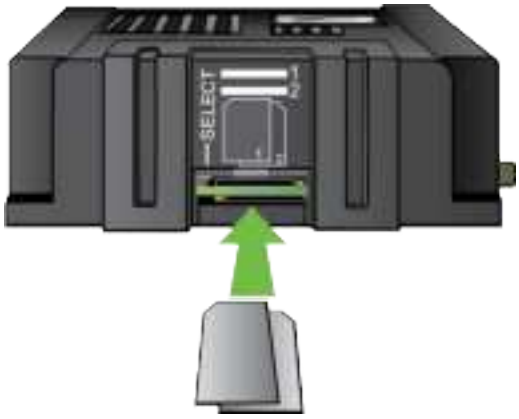
a. Use a screw driver to remove the SIM slot cover.



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

Note If the IX10 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.

c. 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.



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

2. **Attach cellular antennas**

   Securely tighten each antenna using the barrel connector at the base of the antenna.

3. **Connect DC power**



4. Wait until a cellular connection has been established.

   When connecting for the first time, it could take several minutes for the IX10 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 IX10 shows either green or blue (solid or flashing) for proper operation.
   - Verify that the signal strength indicator on the front of the IX10 shows 2 or more bars.

   Note If your SIM card has an APN that is not recognized by the IX10 device, skip this step and configure the APN following the procedure at Configure cellular modem APNs.

5. If you intend to configure Ethernet WAN access at this time, use an Ethernet cable to connect the IX10's **WAN/ETH1** port to a hub with access to the Internet.
6. Use an Ethernet cable to connect the IX10 **ETH** port to your PC.

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 RM device configuration to automatically update the device. See the Digi Remote Manager User Guide.

1. On the PC connected to the IX10, open a browser and go to **192.168.210.1**.
2. Log into the IX10:

   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).

Digi IX10 hardware reference

Digi IX10 features and specifications

The Digi IX10 key features include:

- Industrial grade components.
- Operating temperatures:

  -40C to +70C/-40F to +158F.
- LTE CAT 4 modem with two SIM slots.
- 10/100 BaseT Ethernet port for high-speed connectivity.

For a detailed list of IX10 hardware specifications, see https://www.digi.com/products/networking/cellular-routers/industrial/digi-ix10#specifications.

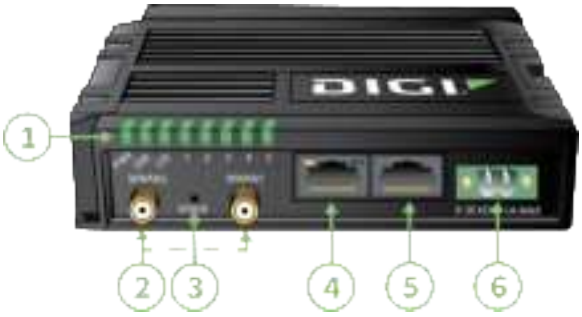
IX10 accessories

When accessories are purchased with the IX10 device, the following are provided:

- Cellular antennas.
- Power supply.
- Ethernet cable.
- DIN rail mounting clip.

IX10 front and side views

The following figure shows the front view of the IX10.



Item	Decription
1. **LEDs**	See IX10 LEDs.
2. **Antenna connectors**	Main (WWAN1) and auxillary (WWAN2) cellular antenna connectors.
3. **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.
4.**Ethernet port**	LAN-enabled by default.
5. **Serial port**	See Digi IX10 serial connector pinout for information about the serial port pin-out.
6. **Power suppy**	IX10 power supply requirements.



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

IX10 LEDs

The IX10 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.

SIM

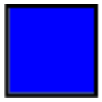
Indicates that a SIM is in use:

	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.		

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.
- **Solid amber**: 1000 Mbps link detected.

Signal quality bars explained

The signal status bars for the Digi IX10 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 IX10 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 IX10 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 IX10 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.

IX10 power supply requirements

IX10 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.

- Use the included power supply (part number 76002104).
- If you are providing the DC power source with a non-Digi power supply, you must use a certified LPS power supply rated at either 12 VDC/0.75 A or 24 VDC/0.375 A minimum. The voltage tolerance supports +/- 10% (9 VDC to 30 VDC) at 9 Watts minimum.
- 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.

Digi IX10 serial connector pinout

The IX10 is a DTE device. The pinout for the 10 pin RJ-50 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.

Hardware setup

This chapter contains the following topics:

Install SIM cards 30
Connect data cables 31
Mount the IX10 device 31

Install SIM cards

To install SIM cards:

1. Use a screw driver to remove the SIM slot cover.



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

   Note If the IX10 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.

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 SIM cards are installed, replace the SIM slot cover.

SIM removal

The IX10 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 IX10 provides two types of data ports:

- **Ethernet** (RJ-45): Use a Cat 5e or Cat 6 Ethernet cable.
- **Serial** (RJ-50): Use a serial cable with an RJ-50 connector to connect to the IX10 device. See 10-pin serial cabling options for information about Digi's 10-pin RJ-50 cables.

Mount the IX10 device

There are two options for mounting the IX10 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 IX10 is purchased with accessories. You can attach the din rail clip directly to the device either on the back or the bottom of the device.

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 IX10 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 IX10.



b. Set the IX10 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 IX10 default settings 35
Change the default password for the admin user 36
Configuration methods 38
Using Digi Remote Manager 39
Access Digi Remote Manager 39
Using the web interface 39
Using the command line 41
Access the command line interface 41
Log in to the command line interface 41
Exit the command line interface 42

Review IX10 default settings

You can review the default settings for your IX10 device by using the local WebUI or Digi Remote Manager:

Local WebUI

1. Log into the IX10 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 IX10.

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.

WebUI

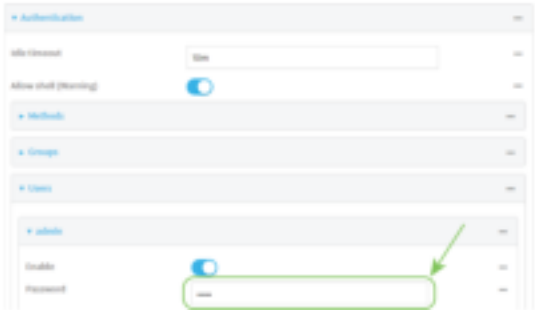
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



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 IX10 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 IX10 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 IX10 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 IX10 device.
  - The local web interface. See Using the web interface for more information about using the local web interface to manage and configure your IX10 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 IX10 device. See Using the command line for more information about using the command line to manage and configure your IX10 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 IX10 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 IX10 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 IX10 local WebUI:

1. Use an Ethernet cable to connect the IX10's **ETH** port to a laptop or PC.
2. Open a browser and go to **192.168.210.1**.
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.
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 IX10 device's status, statistics, and identifying information.

Dashboard area	Description
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).

Log out of the web interface

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



Using the command line

The Digi IX10 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 IX10 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: Configure the 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 IX10 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**192.168.210.1** on the .
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 IX10 command line.

You will now be connected to the Admin CLI:

```
Connecting now, 'exit' to disconnect from 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.

Interfaces

IX10 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) 44
Local Area Networks (LANs) 94

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 IX10 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 IX10 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 IX10 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, either:
  - 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 or HTTPS test: Requires the URL of the host to be tested.
  - 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 IX10 device upon test failure:
  - The default behavior, which is to restart the **Modem** interface.
  - 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.

To configure the IX10 device to regularly probe connections through the WWAN:

WebUI

SureLink can be configured for both IPv4 and IPv6.

1. Log into the IX10 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. Click to expand **Test targets**.
8. For **Add Test Target**, click ✚.



9. 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.

10. Optional active recovery configuration parameters:
    a. If **Reboot device** is enabled, 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**.
    b. 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.
    c. 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.
    d. 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.
    e. For **Failed attempts**, type the number of probe attempts before the WAN is considered to have failed.
    f. 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.
11. (Optional) Repeat this procedure for IPv6.
12. 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 IX10 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. Add a test target:

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

6. 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
    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.

7. 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. To configure the device to restart the interface when its connection is considered to have failed:

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

   This is useful for interfaces that may regain connectivity after restarting, such as a cellular modem.

   c. To configure the device to reboot when the interface is considered to have failed:

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

   Note If both the **restart** and **reboot** parameters are enabled, the **reboot** parameter takes precedence.

   d. 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.

   e. 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**.

f. 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**.

g. 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.

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

```
(config network interface my_wwan ipv4 surelink)> 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 reboot when a failure is detected

Using SureLink, you can configure the IX10 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 IX10 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, either:
  - 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 or HTTPS test: Requires the URL of the host to be tested.

- 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 IX10 device to reboot when an interface has failed:

WebUI

SureLink can be configured for both IPv4 and IPv6.

1. Log into the IX10 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. Enable **Reboot device**.

   Note If both the **Restart interface** and **Reboot device** parameters are enabled, the **Reboot device** parameter takes precedence.

8. (Optional) 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. If **Reboot device** is enabled, 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**.
    b. 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.
    c. 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.
    d. 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.
    e. For **Failed attempts**, type the number of probe attempts before the WAN is considered to have failed.
    f. 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 IX10 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. 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 both the **restart** and **reboot** parameters are enabled, the **reboot** parameter takes precedence.

6. (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
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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device cannot connect to the network using SIM1, it automatically fails over to SIM2. IX10 devices automatically use the correct cellular module firmware for each carrier when switching SIMs.

Configure cellular modem

Configuring the IX10'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 IX10 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 IX10 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 IX10 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

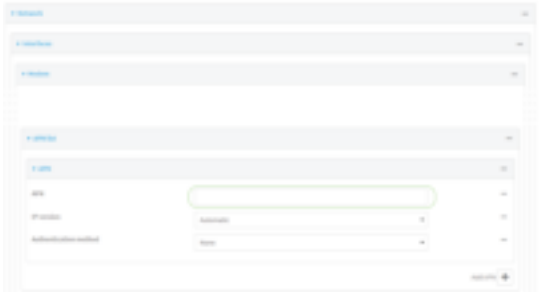
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

   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) **Authentication method**:

   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 IX10 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 IX10 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 IX10 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
 Manufacturer              : Telit
 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 IX10 device cannot make a cellular connection.

Command line

To unlock a SIM card:

1. Log into the IX10 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 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 IX10 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 IX10 command line:

Command line

1. Log into the IX10 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 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 IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



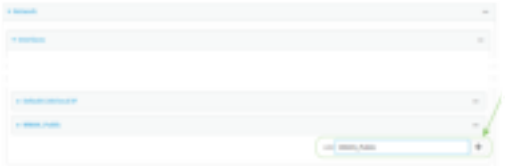
The **Configuration** window is displayed.



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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 WebUI as a user with Admin access.
2. From the menu, click **Status**.
3. Under **Networking**, click **Interfaces**.

Command line

1. Log into the IX10 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 IX10 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 IX10 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.

Local Area Networks (LANs)

The IX10 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) 95
Configure a LAN 95
Show LAN status and statistics 101
Delete a LAN 103
DHCP servers 104
Create a Virtual LAN (VLAN) route 120

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 IX10 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. 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.
9. See Configure DHCP relay for information about configuring **DHCP relay**.
10. (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**.
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 there 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 IX10 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.

(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.

(Optional) Configure the MAC address allowlist.

If there 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.

8. Save the configuration and apply the change:

```
(config network interface my_lan)> 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 LAN status and statistics

WebUI

1. Log into the IX10 WebUI as a user with Admin access.
2. From the menu, click **Status**.
3. Under **Networking**, click **Interfaces**.

Command line

1. Log into the IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device and to diagnose DHCP issues.

WebUI

1. Log into the IX10 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 IX10 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 IX10 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 IX10 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.

Serial port

IX10 devices have a single serial port that provides access to the command-line interface.

Use an RS-232 serial cable to establish a serial connection from your IX10 to your local laptop or PC. Use a terminal emulator program to establish the serial connection. The terminal emulator's serial connection must be configured to match the configuration of the IX10 device's serial port. 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 the serial port

By default, the IX10 serial port is configured as follows:

- **Enabled**
- **Serial mode**: Remote
- **Label**: None
- **Baud rate**: 9600
- **Data bits**: 8
- **Parity**: None
- **Stop bits**: 1
- **Flow control**: None

To change the configuration to match the serial configuration of the device to which you want to connect:

WebUI

1. Log into the IX10 WebUI as a user with Admin access.
2. On the menu, click **System**. Under **Configuration**, click **Serial Configuration**.



The **Serial Configuration** page is displayed.



3. **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.
4. Click to expand **Port 1**.



The serial port is enabled by default. To disable, toggle off **Enable**.

5. For **Mode**, select one of the following:
   - **Login**: Allows the user to log into the device through the serial port.
   - **Remote access**: Allows for remote access to another device that is connected to the serial port.
   - **Application**: Provides access to the serial device from Python applications. See Use Python to access serial ports for information about creating Python applications that access the serial port.
   - **Modbus**: Allows you to use the serial port for Modbus. See Modbus gateway.
   - **UDP serial**: Provides access to the device through a UDP serial port. See Configure UDP serial mode.

   The default is **Remote**.
6. (Optional) For **Label**, enter a label that will be used when referring to this port.

7. If **Login**, **Remote Access**, or **Modbus** is selected for **Mode**:
   a. Click to expand **Serial Settings**.



   b. For **Baud rate**, select the baud rate used by the device to which you want to connect.
   c. For **Data bits**, select the number of data bits used by the device to which you want to connect.
   d. For **Parity**, select the type of parity used by the device to which you want to connect.
   e. For **Stop bits**, select the number of stop bits used by the device to which you want to connect.
   f. For **Flow control**, select the type of flow control used by the device to which you want to connect.
8. (Optional) If Remote Access is selected for **Mode**:
   a. Click to 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.
   b. Click to expand **Session Settings**.



   c. Enable **Exclusive access** to limit access to the serial port to a single active session.
   d. 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**.
   e. 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.
   f. For **Idle timeout**, type the amount of time to wait before disconnecting due to user inactivity.
1. Click to 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.

9. (Optional) Copy the serial port's configuration by clicking the (copy) icon.

   The **Copy Configuration** window displays.



   a. For **Copy Port 1 configuration to these ports:**, type the names of the ports that the configuration should be copied to.
   b. For **Copy these settings**, select the types of settings that should be copied to the selected ports.
   c. Click **Copy**.

10. 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 IX10 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 mode
(config)>
```

where *mode* is either:

- **login**: Allows the user to log into the device through the serial port.
- **remote**: Allows for remote access to another device that is connected to the serial port.
- **application**: Provides access to the serial device from Python applications. See Use Python to access serial ports for information about creating Python applications that access the serial port.
- **modbus**: Allows you to use the serial port for Modbus. See Modbus gateway.
- **udpserial**: Provides access to the device through a UDP serial port. See Configure UDP serial mode.

The default is **login**.

5. (Optional) Set a label that will be used when referring to this port.

```
(config)> serial port1 label label
(config)>
```

6. If **mode** is set to **login** or **remote**:
   a. Set the baud rate used by the device to which you want to connect:

```
(config)> serial port1 baudrate rate
(config)>
```

   b. Set the number of data bits used by the device to which you want to connect:

```
(config)> serial port1 databits bits
(config)>
```

   c. Set the type of parity used by the device to which you want to connect:

```
(config)> serial port1 parity parity
(config)>
```

Allowed values are:

- **even**
- **odd**
- **none**

The default is **none**.

d. Set the stop bits used by the device to which you want to connect:

```
(config)> serial port1 stopbits bits
(config)>
```

e. Set the type of flow control used by the device to which you want to connect:

```
(config)> serial port1 flow type
(config)
```

Allowed values are:

- **none**
- **rts/cts**
- **xon/xoff**

The default is **none**.

7. If **mode** is set to **remote**:

a. Set the characters used to start an escape sequence:

```
(config)> serial port1 escape 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)> serial port1 exclusive 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)> serial port1 history bytes
(config)
```

The default is **4000** bytes.

d. Set the amount of time to wait before disconnecting due to user inactivity:

```
(config)> serial port1 idle_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)> serial port1 idle_timeout 600s
(config)
```

The default is **15m**.

e. (Optional) Enable monitoring of CTS (Clear to Send) changes on this port:

```
(config)> serial port1 monitor cts true
(config)
```

f. (Optional) Enable monitoring of DCD (Data Carrier Detect) changes on this port:

```
(config)> serial port1 monitor dcd true
(config)
```

g. Configure TCP access to this port:

i. Set the connection type:

```
(config serial USB_port)> service tcp conn_type value
(config serial USB_port)>
```

where *value* is one of:

i. **tcp**: The TCP connection is unencrypted.

ii. **tls**: The TCP connection uses Transport Layer Security (TLS) encryption.

iii. **tls_auth**: The TCP connection uses TLS encryption with authentication.

ii. Enable TCP access:

```
(config serial USB_port)> service tcp enable true
(config serial USB_port)>
```

iii. Set the TCP port:

```
(config serial USB_port)> service tcp port port
(config serial USB_port)>
```

iv. (Optional) Configure the access control list to limit access to the TCP connection:

- To limit access to specified IPv4 addresses and networks:

```
(config serial USB_port)> add service tcp acl address end
value
(config serial USB_port)>
```

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 tcp port.

Repeat this step to list additional IP addresses or networks.

- To limit access to specified IPv6 addresses and networks:

```
(config serial USB_port)> add service tcp acl address6 end
value
(config serial USB_port)>
```

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 tcp port.

Repeat this step to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX10 device:

```
(config serial USB_port)> add service tcp acl interface end
value
(config serial USB_port)>
```

Where *value* is an interface defined on your device.

Display a list of available interfaces:

Use **... network interface ?** to display interface information:

```
(config serial USB_port)> ... network interface ?

Interfaces

Additional Configuration
-----------------------------------------
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config serial USB_port)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config serial USB_port)> add service tcp 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 serial USB_port)> ... 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 serial USB_port)>
```

Repeat this step to list additional firewall zones.

v. (Optional) Enable mDNS. mDNS is a protocol that resolves host names in small networks that do not have a DNS server.

```
(config serial USB_port)> service tcp mdns enable true
(config serial USB_port)>
```

h. Configure telnet access to this port:



CAUTION! This connection is not authenticated or encrypted.

i. Enable telnet access:

```
(config serial USB_port)> service telnet enable false
(config serial USB_port)>
```

ii. Set the telnet port:

```
(config serial USB_port)> service telnet port port
(config serial USB_port)>
```

iii. (Optional) Configure the access control list to limit access to the telnet connection:

- To limit access to specified IPv4 addresses and networks:

```
(config serial USB_port)> add service telnet acl address end
value
(config serial USB_port)>
```

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 port.

Repeat this step to list additional IP addresses or networks.

- To limit access to specified IPv6 addresses and networks:

```
(config serial USB_port)> add service telnet acl address6 end
value
(config serial USB_port)>
```

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 port.

Repeat this step to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX10 device:

```
(config serial USB_port)> add service telnet acl interface
end value
(config serial USB_port)>
```

Where *value* is an interface defined on your device.

Display a list of available interfaces:

Use **... network interface ?** to display interface information:

```
(config serial USB_port)> ... network interface ?

Interfaces

Additional Configuration
-----------------------------------------
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config serial USB_port)>
```

Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config serial USB_port)> 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 serial USB_port)> ... 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 serial USB_port)>
```

Repeat this step to list additional firewall zones.

iv. (Optional) Enable **mDNS**. mDNS is a protocol that resolves host names in small networks that do not have a DNS server.

```
(config serial USB_port)> service telnet mdns enable true
(config serial USB_port)>
```

i. Configure ssh access to this port:

i. Enable ssh access:

```
(config serial USB_port)> service ssh enable false
(config serial USB_port)>
```

ii. Set the ssh port:

```
(config serial USB_port)> service ssh port port
(config serial USB_port)>
```

iii. (Optional) Configure the access control list to limit access to the ssh connection:

- To limit access to specified IPv4 addresses and networks:

```
(config serial USB_port)> add service ssh acl address end
value
(config serial USB_port)>
```

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 port.

Repeat this step to list additional IP addresses or networks.

- To limit access to specified IPv6 addresses and networks:

```
(config serial USB_port)> add service ssh acl address6 end
value
(config serial USB_port)>
```

  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 port.

  Repeat this step to list additional IP addresses or networks.

- To limit access to hosts connected through a specified interface on the IX10 device:

```
(config serial USB_port)> add service ssh acl interface end
value
(config serial USB_port)>
```

  Where *value* is an interface defined on your device.

  Display a list of available interfaces:

  Use **... network interface ?** to display interface information:

```
(config serial USB_port)> ... network interface ?

Interfaces

Additional Configuration
----------------------------------------
 defaultip              Default IP
 defaultlinklocal       Default Link-local IP
 eth                    ETH
 loopback               Loopback
 modem                  Modem

config serial USB_port)>
```

  Repeat this step to list additional interfaces.

- To limit access based on firewall zones:

```
(config serial USB_port)> 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 serial USB_port)> ... 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 serial USB_port)>
```

Repeat this step to list additional firewall zones.

iv. (Optional) Enable **mDNS**. mDNS is a protocol that resolves host names in small networks that do not have a DNS server.

```
(config serial USB_port)> service ssh mdns enable true
(config serial USB_port)>
```

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 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 IX10 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.



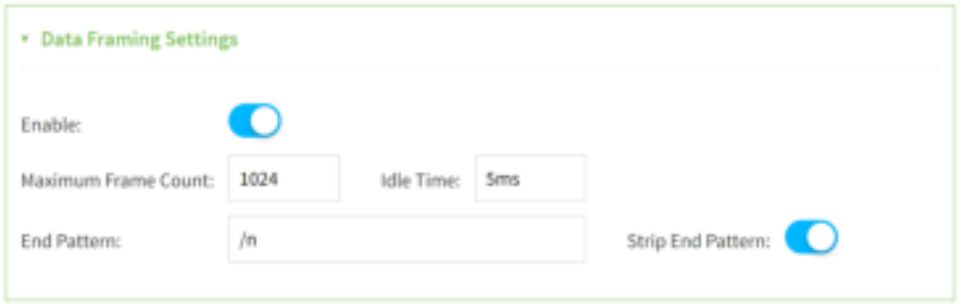
The serial port is enabled by default. To disable, toggle off **Enable**.

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**.



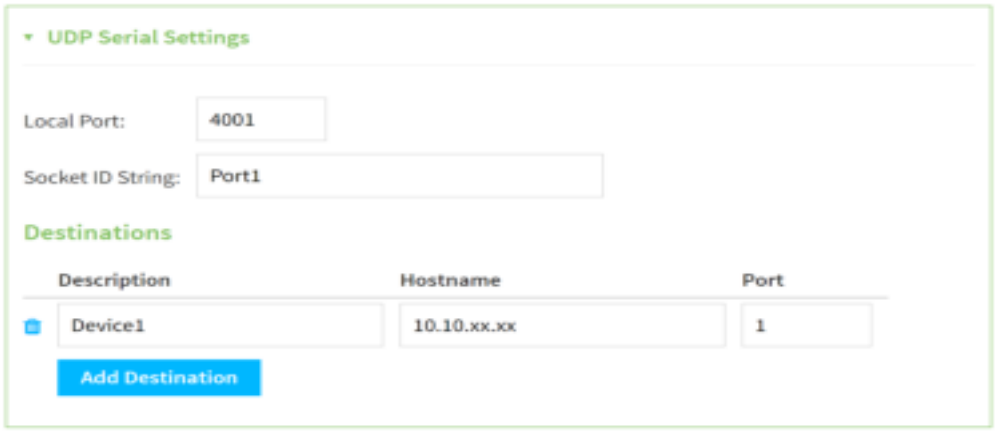
   a. For **Baud rate**, select the baud rate used by the device to which you want to connect.
   b. For **Data bits**, select the number of data bits used by the device to which you want to connect.
   c. For **Parity**, select the type of parity used by the device to which you want to connect.
   d. For **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.

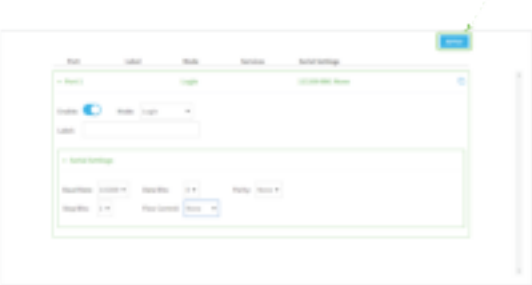
8. Expand **UDP Serial Settings**.



   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 IX10 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.



Show serial status and statistics

To show the status and statistics for the serial port:

WebUI

1. Log into the IX10 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 IX10 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 IX10 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 141
Show the routing table 157
Dynamic DNS 158
Virtual Router Redundancy Protocol (VRRP) 163

IP routing

The IX10 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 142
Delete a static route 145
Policy-based routing 146
Configure a routing policy 147
Routing services 154
Configure routing services 154

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 IX10 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.

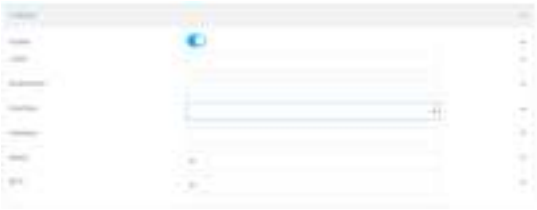


3. Click **Network** > **Routes** > **Static routes**.

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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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

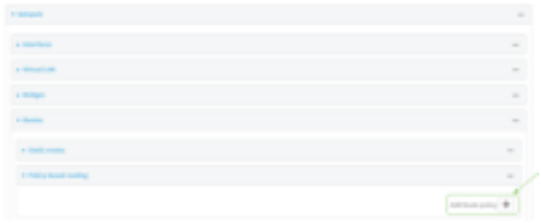
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 devices:



Configure device one (master device)

WebUI

Task 1: Configure VRRP on device one

1. Log into the IX10 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.



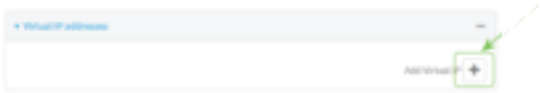
3. Click **Network** > **VRRP**.
4. For **Add VRRP instance**, type a name for the VRRP instance and click ✚.



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**.

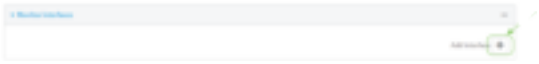
9. Click to expand **Virtual IP addresses**.
10. Click ✚ to add a virtual IP address.



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 IX10 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 IX10 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.

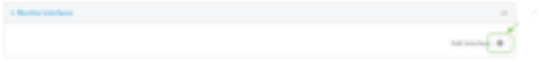
5. Click **Enable**.
6. For **Interface**, select **Interface: ETH**.
7. For **Router ID**, leave at the default setting of **50**.
8. For **Priority**, type **80**.
9. Click to expand **Virtual IP addresses**.
10. Click ✚ to add a virtual IP address.



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 IX10 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 IX10 device. VRRP status is available from the Web UI only.

WebUI

1. Log into the IX10 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 IX10 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 189
OpenVPN 235
Generic Routing Encapsulation (GRE) 267
NEMO 288

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 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 IX10 device can be configured to authenticate with the remote peer as an XAUTH client.

RSA Signatures

With RSA signatures authentication, the IX10 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 IX10 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 IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



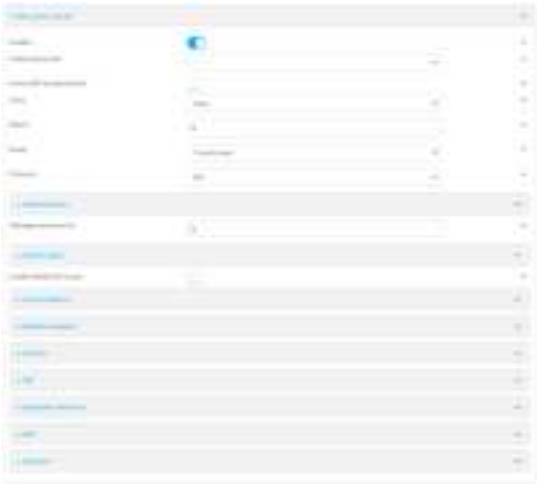
The **Configuration** window is displayed.



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.

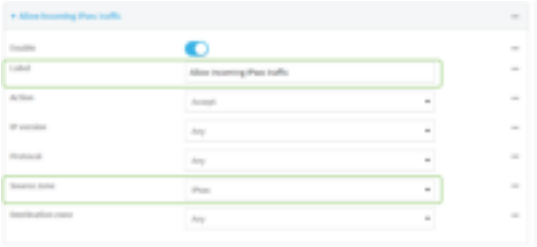


6. The IPsec tunnel is enabled by default. To disable, click **Enable**.
7. (Optional) **Preferred tunnel** provides an optional mechanism for IPsec failover behavior. See Configure IPsec failover for more information.
8. (Optional) Enable **Force UDP encapsulation** to force the tunnel to use UDP encapsulation 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. 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.
12. Select the **Protocol**, either:
    - **ESP** (Encapsulating Security Payload): Provides encryption as well as authentication and integrity.
    - **AH** (Authentication Header): Provides authentication and integrity only.
13. 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 IX10 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.

14. (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.
15. (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.
16. (Optional) Click **Enable MODECFG client** to receive configuration information, such as the private IP address, from the remote peer.
17. 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.

18. 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.

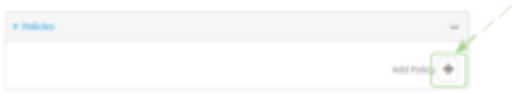
- **KeyID**: The ID will be interpreted as a Key ID and sent as an ID_KEY_ID IKE identity.

  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.

19. Click to expand **Policies**.

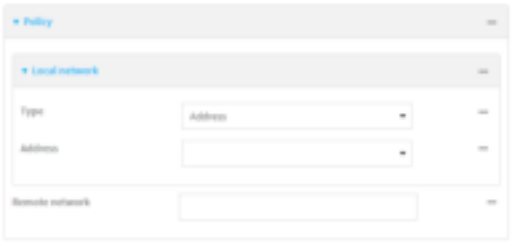
    Policies define the network traffic that will be encapsulated by this tunnel.

    a. Click ✚ to create a new policy.

    The new policy configuration is displayed.

    b. Click to expand **Local network**.

    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. The keyword **any** can also be used.
    - **Request a network**: Requests a network from the remote peer.

    d. For **Remote network**, enter the IP address and optional netmask of the remote network. The keyword **any** can also be used. .

20. 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 **Enable padding**, click to disable the padding of IKE packets. This should normally not be disabled except for compatibility purposes.
    e. 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**.
    f. 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**.
    g. 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**.
    h. 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**.

i. 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**.

21. (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.
22. (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.
23. See Configure SureLink active recovery for IPsec for information about IPsec **Active recovery**.
24. (Optional) Click **Advanced** to set various IPsec-related time out, keep alive, and related values.
25. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX10 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. 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)>
```

    e. 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.

    f. 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.

g. 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.

h. 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:

```
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)> dh_
group value
(config vpn ipsec tunnel ipsec_example ike phase1_proposal 0)>
```

where *value* is one of **ecp384**, **modp768**, **modp1024**, **modp1536**, **modp2048**, **modp3072**, **modp4096**, **modp6144**, or **modp8192**, . The default is **modp1024**.

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.

i. 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:

```
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)> dh_
group value
(config vpn ipsec tunnel ipsec_example ike phase2_proposal 0)>
```

where *value* is one of **ecp384**, **modp768**, **modp1024**, **modp1536**, **modp2048**, **modp3072**, **modp4096**, **modp6144**, or **modp8192**, . The default is **modp1024**.

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 network policy:

```
(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.

    d. Set the IP address and optional netmask of the remote network. The keyword **any** can also be used.

```
(config vpn ipsec tunnel ipsec_example policy 0)> remote network value
(config vpn ipsec tunnel ipsec_example policy 0)>
```

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.

```
(config)> vpn ipsec advanced ?

Advanced: Advanced configuration that applies to all IPsec tunnels.

 Parameters                Current Value
 ------------------------------------------------------------------------
----------
 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.
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 IX10 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 IX10 device to regularly probe IPsec client connections 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 IX10 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 IX10 device to regularly probe the IPsec connection:

WebUI

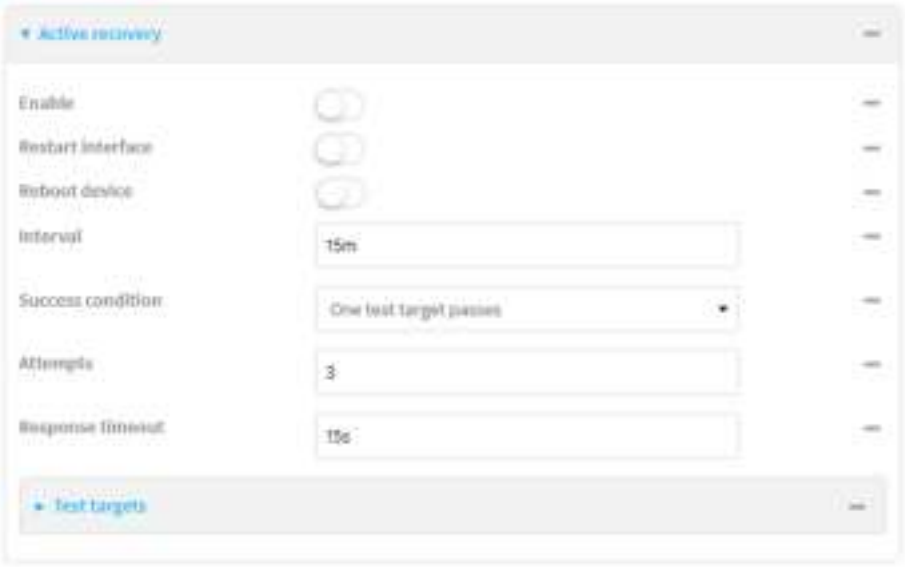
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



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**}.

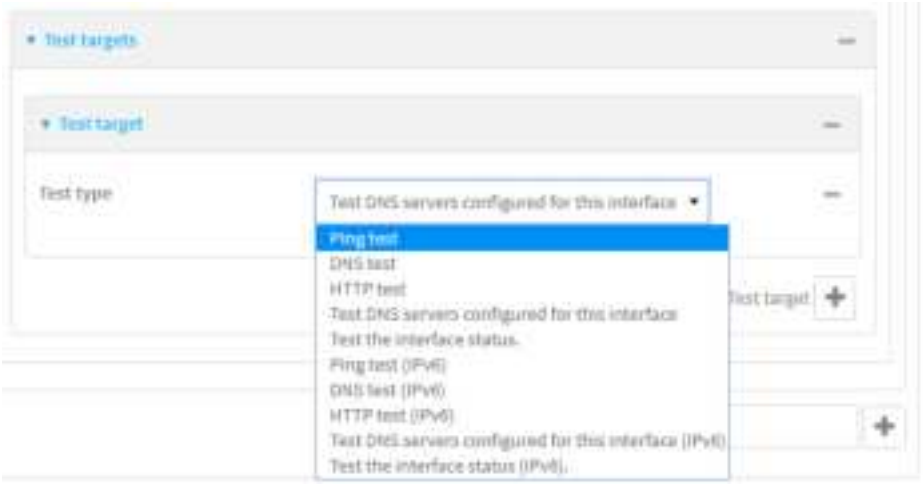
   For example, to set **Interval** to ten minutes, enter **10m** or **600s**.

   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**:
       - **Ping test** or **Ping test (IPv6)**: 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** or **DNS test (IPv6)**: Tests connectivity by sending a DNS query to the specified **DNS server**.

- **HTTP test HTTP test (IPv6)**: 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** or **Test DNS servers configured for this interface (IPv6)**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
- **Test the interface status** or **Test the interface status IPv6**: 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 IX10 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)> connection_monitor 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)> connection_monitor 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)> connection_monitor reboot enable
(config vpn ipsec tunnel ipsec_example)>
```

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

```
(config vpn ipsec tunnel ipsec_example)> connection_monitor 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)> connection_monitor 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)> connection_monitor 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)> connection_monitor 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)> connection_monitor 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)> connection_monitor interval 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 connection_monitor target
    end
    (config vpn ipsec tunnel ipsec_example connection_monitor target 0)>
    ```

    b. Set the test type:

    ```
    (config vpn ipsec tunnel ipsec_example connection_monitor target 0)>
    test value
    (config vpn ipsec tunnel ipsec_example connection_monitor target 0)>
    ```

    where *value* is one of:

    - **ping** (IPv4) or **ping6** (IPv6): Tests connectivity by sending an ICMP echo request to a specified hostname or IP address.
      - Specify the hostname or IP address by using **ping_host** or **ping_host6**:

      ```
      (config vpn ipsec tunnel ipsec_example connection_monitor
      target 0)> ping_host host
      (config vpn ipsec tunnel ipsec_example connection_monitor
      target 0)>
      ```

      - (Optional) Set the size, in bytes, of the ping packet by using **ping_size** or **ping_size6**:

      ```
      (config vpn ipsec tunnel ipsec_example connection_monitor
      target 0)> ping_size [num]
      (config vpn ipsec tunnel ipsec_example connection_monitor
      target 0)>
      ```

    - **dns** (IPv4) or **dns6** (IPv6): 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 connection_monitor
      target 0)> dns_server ip_address
      (config vpn ipsec tunnel ipsec_example connection_monitor
      target 0)>
      ```

    - **dns_configured** (IPv4) or **dns_configured6** (IPv6): Tests connectivity by sending a DNS query to the DNS servers configured for this interface.

- **http** (IPv4) or **http6** (IPv6): Tests connectivity by sending an HTTP or HTTPS GET request to the specified URL.
  - Specify the url. Allowed value uses the format **http[s]://*hostname*/[*path*]**.

    ```
    (config vpn ipsec tunnel ipsec_example connection_monitor
    target 0)> http_url url
    (config vpn ipsec tunnel ipsec_example connection_monitor
    target 0)>
    ```

- **interface_up** (IPv4) or **interface_up6** (IPv6): : 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 connection_monitor
    target 0)> interface_down_time value
    (config vpn ipsec tunnel ipsec_example connection_monitor
    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 vpn ipsec tunnel ipsec_example connection_monitor
    target 0)> interface_down_time 600s
    (config vpn ipsec tunnel ipsec_example connection_monitor
    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 connection_monitor
    target 0)> interface_timeout value
    (config vpn ipsec tunnel ipsec_example connection_monitor
    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 vpn ipsec tunnel ipsec_example connection_monitor
    target 0)> interface_timeout 600s
    (config vpn ipsec tunnel ipsec_example connection_monitor
    target 0)>
    ```

    The default is 60 seconds.

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 IX10 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 IX10 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.

There are two methods to enable IPsec debug messages:

- From the Admin CLI—Sets the debug level to **1** (basic debugging information only).
- From the interactive shell—Allows for more detailed debug information.

Use the Admin CLI to set the IPsec debug level to 1

To set the debug level to **1** by using the Admin CLI:

Command line

1. Log into the IX10 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 **action ipsec debug** command to **true**:

```
config> action ipsec debug 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.

This sets the IPsec debug level to **1**.

Use the interactive shell to set the IPsec debug level

By using the interactive shell to set the debug level, you can enable the IX10 device to write additional debug messages to the system log. The command accepts the following values to set the debug level:

- **-1** — (Default) No debug information is written. This is the equivalent of turning off debug messages for IPsec.

- **0** — Basic auditing logs, (for example, SA up/SA down).
- **1** — Generic control flow with errors. Select this for basic debugging information.
- **2** — More detailed debugging control flow.
- **3** — Includes RAW data dumps in hexadecimal format.
- **4** — Also includes sensitive material in dumps (for example, encryption keys).

To access the shell menu option, you must have shell access enabled. See Authentication groups for information about configuring authentication groups that include shell access.

Command line

1. Log into the IX10 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, execute the following command:

   ```
   # ipsec stroke loglevel ike debug_level
   #
   ```

   where *debug_level* is one of the following:

   - **-1** — (Default) No debug information is written. This is the equivalent of turning off debug messages for IPsec.
   - **0** — Basic auditing logs, (for example, SA up/SA down).
   - **1** — Generic control flow with errors. Select this for basic debugging information.
   - **2** — More detailed debugging control flow.
   - **3** — Includes RAW data dumps in hexadecimal format.
   - **4** — Also includes sensitive material in dumps (for example, encryption keys).
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 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 IX10 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 IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



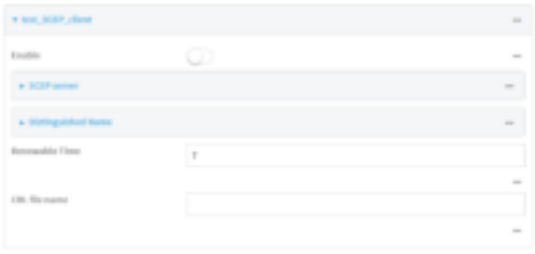
The **Configuration** window is displayed.



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

IX10 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 IX10 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. For **Password**, type the challenge password as configured on the SCEP server.
11. Click to expand **Distinguished Name**.



12. Type the value for each appropriate Distinguished Name attribute.
13. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX10 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. 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)>
```

7. 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)>
```

8. 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 IX10 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)>
```

9. (Optional) Set the filename of the Certificate Revocation List (CRL) from the CA.

   The CRL is stored on the IX10 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)>
```

10. Save the configuration and apply the change:

```
(config network scep_client scep_client_name)> 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.

Example: SCEP client configuration with Fortinet SCEP server

In this example configuration, we will configure the IX10 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 IX10 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 IX10 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 IX10 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**.

IX10 configuration

On the IX10 device:

WebUI

1. Log into the IX10 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 IX10 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 IX10 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.

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 IX10 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 IX10 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 IX10 device supports two mechanisms for configuring an OpenVPN server in TAP mode:

- OpenVPN managed—The IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 user authentication for more information about creating authentication groups and users.

WebUI

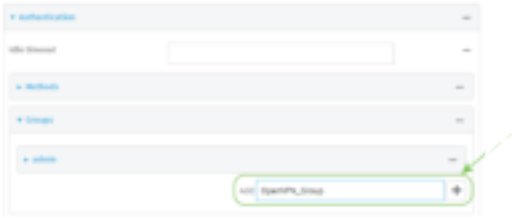
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



The **Configuration** window is displayed.



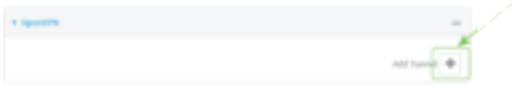
3. Add an OpenVPN authentication group:
   a. Click **Authentication** > **Groups**.
   b. For **Add Group**, type a name for the group (for example, **OpenVPN_Group**) and click ✚.



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.

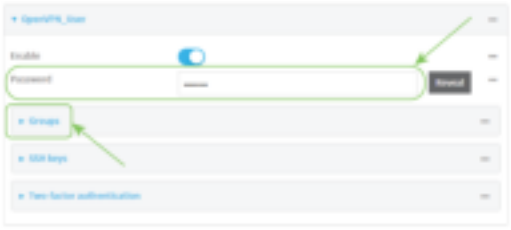
4. Add an OpenVPN authentication user:
   a. Click **Authentication** > **Users**.
   b. For **Add**, type a name for the user (for example, **OpenVPN_User**) and click ✚.



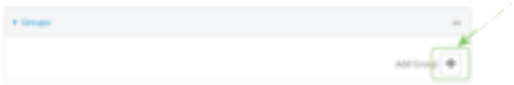
   c. Type a password for the user.

      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.

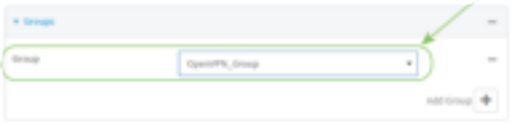
   d. Click to expand the **Groups** node.



   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 IX10 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

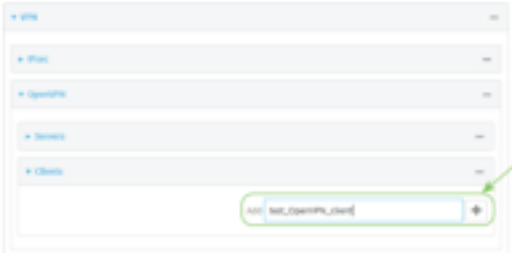
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



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 IX10 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

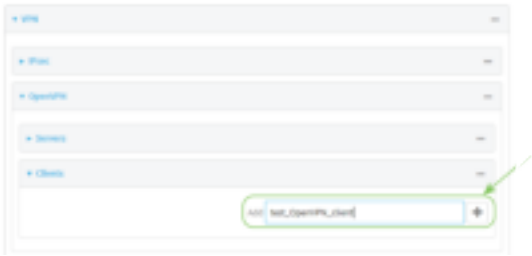
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

   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 IX10 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 IX10 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 IX10 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 IX10 device to regularly probe the OpenVPN connection:

WebUI

1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



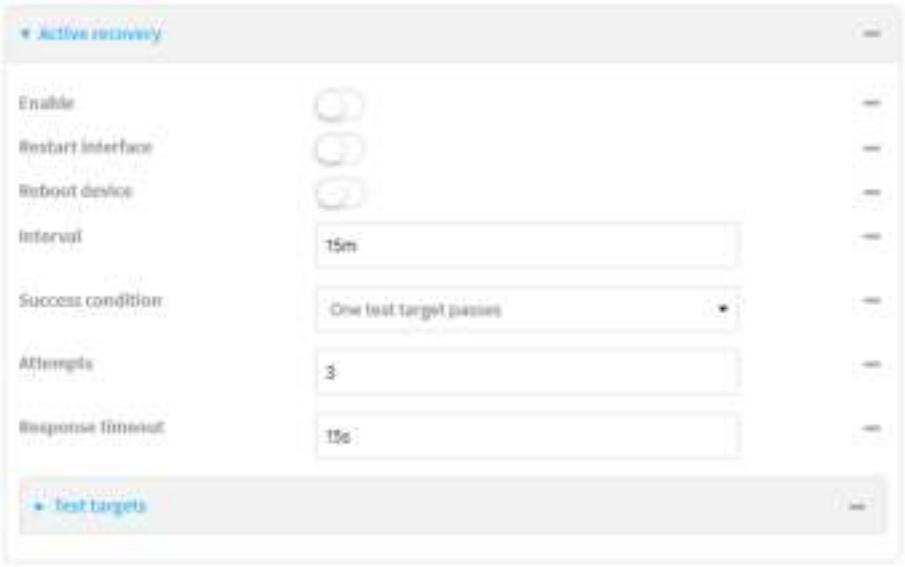
The **Configuration** window is displayed.



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**.

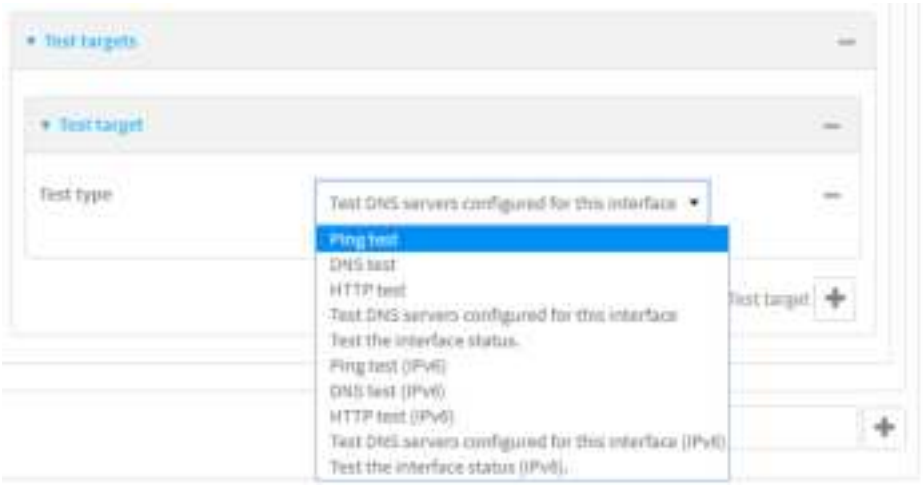
   The default is 15 minutes.
10. For **Success condition**, determine whether the interface should fail over based on the failure of one of the test targets, or all of the test targets.
11. For **Attempts**, type the number of probe attempts before the WAN is considered to have failed.
12. For **Response 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**:
        - **Ping test** or **Ping test (IPv6)**: 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** or **DNS test (IPv6)**: Tests connectivity by sending a DNS query to the specified **DNS server**.
        - **HTTP test HTTP test (IPv6)**: 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** or **Test DNS servers configured for this interface (IPv6)**: Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
        - **Test the interface status** or **Test the interface status IPv6**: 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 IX10 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)> connection_monitor 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)> connection_monitor 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)> connection_monitor reboot
enable
(config vpn openvpn client openvpn_client1)>
```

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

```
(config vpn openvpn client openvpn_client1)> connection_monitor 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)> connection_monitor 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)> connection_monitor 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)> connection_monitor 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)> connection_monitor 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)> connection_monitor interval
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 connection_monitor
target end
(config vpn openvpn client openvpn_client1 connection_monitor target
0)>
```

    b. Set the test type:

```
(config vpn openvpn client openvpn_client1 connection_monitor target
0)> test value
(config vpn openvpn client openvpn_client1 connection_monitor target
0)>
```

    where *value* is one of:

    - **ping** (IPv4) or **ping6** (IPv6): Tests connectivity by sending an ICMP echo request to a specified hostname or IP address.
      - Specify the hostname or IP address by using **ping_host** or **ping_host6**:

```
(config vpn openvpn client openvpn_client1 connection_monitor
target 0)> ping_host host
(config vpn openvpn client openvpn_client1 connection_monitor
target 0)>
```

      - (Optional) Set the size, in bytes, of the ping packet by using **ping_size** or **ping_size6**:

```
(config vpn openvpn client openvpn_client1 connection_monitor
target 0)> ping_size [num]
(config vpn openvpn client openvpn_client1 connection_monitor
target 0)>
```

    - **dns** (IPv4) or **dns6** (IPv6): 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 connection_monitor
target 0)> dns_server ip_address
(config vpn openvpn client openvpn_client1 connection_monitor
target 0)>
```

    - **dns_configured** (IPv4) or **dns_configured6** (IPv6): Tests connectivity by sending a DNS query to the DNS servers configured for this interface.
    - **http** (IPv4) or **http6** (IPv6): Tests connectivity by sending an HTTP or HTTPS GET request to the specified URL.
      - Specify the url. Allowed value uses the format **http[s]://*hostname*/[*path*]**.

```
(config vpn openvpn client openvpn_client1 connection_monitor
target 0)> http_url url
```

```
(config vpn openvpn client openvpn_client1 connection_monitor
target 0)>
```

- **interface_up** (IPv4) or **interface_up6** (IPv6): : 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 connection_monitor
    target 0)> interface_down_time value
    (config vpn openvpn client openvpn_client1 connection_monitor
    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 vpn openvpn client openvpn_client1 connection_monitor
    target 0)> interface_down_time 600s
    (config vpn openvpn client openvpn_client1 connection_monitor
    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 connection_monitor
    target 0)> interface_timeout value
    (config vpn openvpn client openvpn_client1 connection_monitor
    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 vpn openvpn client openvpn_client1 connection_monitor
    target 0)> interface_timeout 600s
    (config vpn openvpn client openvpn_client1 connection_monitor
    target 0)>
    ```

    The default is 60 seconds.

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.

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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device with a GRE tunnel through IPsec.



IX10-1 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 IX10-2 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 IX10-2, **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**.

IX10-2 configuration tasks

1. Create an IPsec tunnel named **ipsec_gre2** with:
   - The same pre-shared key as the **ipsec_gre1** tunnel on IX10-1.
   - **Remote endpoint** set to the public IP address of IX10-1.
   - 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 IX10-1, **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 IX10-1 device
Task one: Create an IPsec tunnel

WebUI

1. Log into the IX10 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 IX10-2 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 IX10 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 IX10-2 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

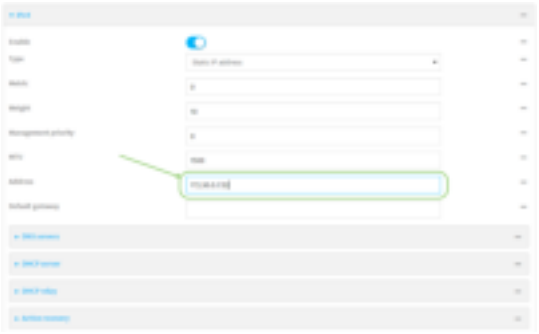
1. Click **Network** > **Interface**.
2. For **Add Interface**, type **ipsec_endpoint1** and click **+**.



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 IX10-2, **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 IX10-2, **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 ✚.

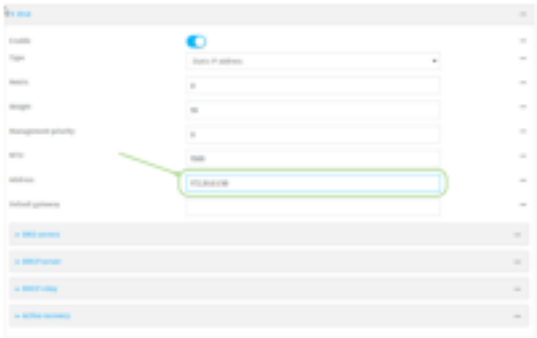


3. For **Zone**, select **Internal**.

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 IX10-2 device
Task one: Create an IPsec tunnel

WebUI

1. Log into the IX10 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.

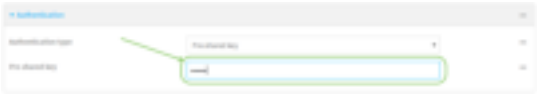


3. Click **VPN** > **IPsec** > **Tunnels**.
4. For **Add IPsec Tunnel**, type **ipsec_gre2** and click **✚**.

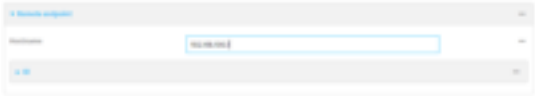


5. Click to expand **Authentication**.

6. For **Pre-shared key**, type the same pre-shared key that was configured for the IX10-1 (**testkey**).



7. Click to expand **Remote endpoint**.
8. For **Hostname**, type public IP address of the IX10-1 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 IX10 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 IX10-1 (**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 IX10-1 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

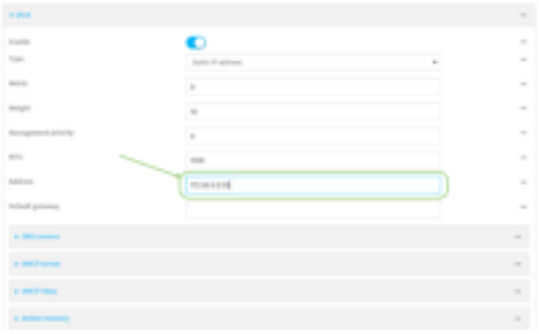
1. Click **Network** > **Interfaces**.
2. For **Add Interface**, type **ipsec_endpoint2** and click **+**.



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 IX10-1, **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 IX10-1, **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 ✚.

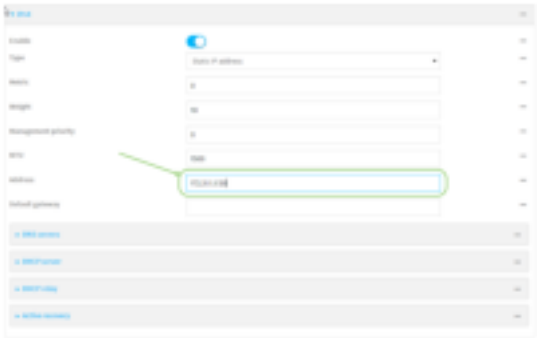


3. For **Zone**, select **Internal**.

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.
>
```

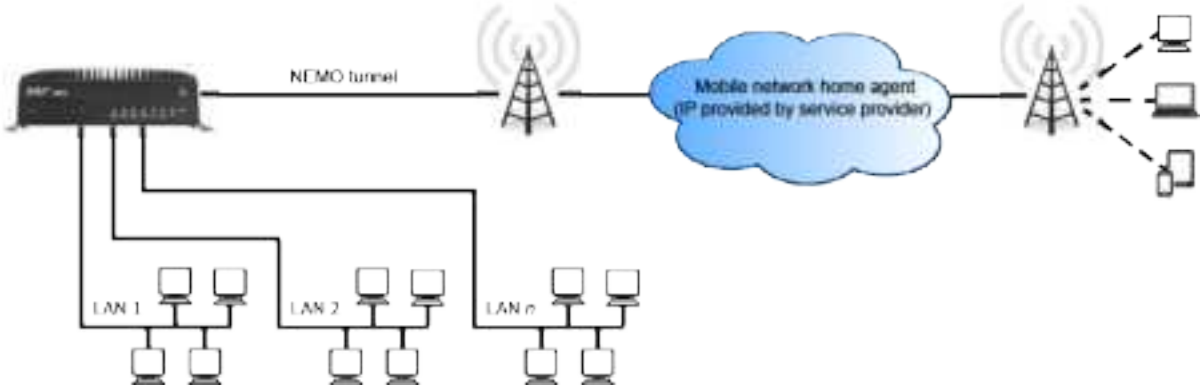
7. Type **exit** to exit the Admin CLI.

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

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 IX10 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 IX10 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 the firewall zone for the NEMO tunnel.
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 IX10 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 IX10 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 IX10 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:

```
(config vpn nemo nemo_example)> zone zone
(config vpn nemo nemo_example)>
```

    To view a list of available zones:

```
(config vpn nemo nemo_example)> zone ?

Zone: The firewall zone assigned to this network interface. This can be
used by
packet filtering rules and access control lists to restrict network
traffic on
this interface.
Format:
  any
  dynamic_routes
  edge
  external
  internal
  ipsec
  loopback
  setup
Current value:

(config vpn nemo nemo_example)> zone
```

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 IX10 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 IX10 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 tunnel, 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 296
Configure the web administration service 300
Configure SSH access 310
Use SSH with key authentication 316
Configure telnet access 319
Configure DNS 324
Simple Network Management Protocol (SNMP) 331
Location information 337
Modbus gateway 365
System time 382
Configure the system time 382
Network Time Protocol 385
Configure the device as an NTP server 386
Configure a multicast route 391
Enable service discovery (mDNS) 393
Use the iPerf service 397
Configure the ping responder service 402

Allow remote access for web administration and SSH

By default, only devices connected to the IX10's LAN have access to the device via web administration and SSH. To enable these services for access from remote devices:

- The IX10 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 IX10 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 IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

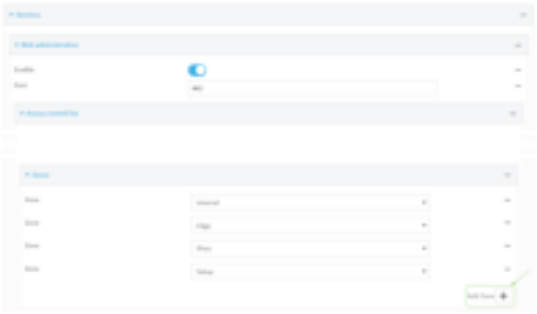


The **Configuration** window is displayed.



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 IX10 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

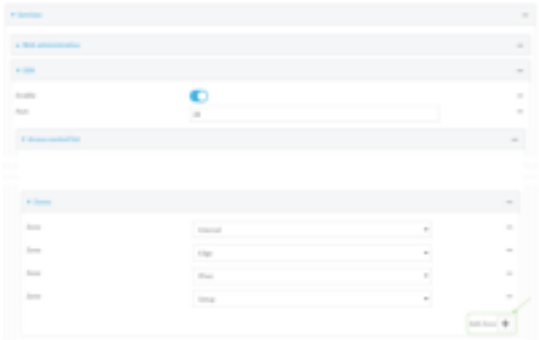
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



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 IX10 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 IX10 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 IX10'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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10'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.

See Set the idle timeout for IX10 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 IX10 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 IX10 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 IX10 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 IX10 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. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX10 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 IX10 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. 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.

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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device supports both SNMPv3 and SNMPv2c in read-only mode. Both are disabled by default. SNMPv1 is not supported.

SNMP Security

By default, the IX10 device automatically blocks SNMP packets from being received over WAN and LAN interfaces. As a result, if you want a IX10 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 IX10 device:

WebUI

1. Log into the IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device.
3. On the main menu, click **Status**. Under **Services**, click **SNMP**.



The **SNMP** page is displayed.



4. Click **Download**.

Location information

Your IX10 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 IX10 device to forward location messages, either from the IX10 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 338
Configure the device to use a user-defined static location 340
Configure the device to accept location messages from external sources 342
Forward location information to a remote host 346
Configure geofencing 352
Show location information 364

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 IX10 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 IX10 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 IX10 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 IX10 device to use a user-defined static location.

WebUI

1. Log into the IX10 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 IX10 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 IX10 device to accept NMEA and TAIP messages from external sources. For example, location-enabled devices connected to the IX10 device can forward their location information to the device, and then the IX10 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 IX10 device to forward location messages.

This procedure configures a UDP port on the IX10 device that will be used to listen for incoming messages.

Required configuration items

- The location server must be enabled.
- UDP port that the IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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.
- 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 IX10 device to forward location information:

WebUI

1. Log into the IX10 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.
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 IX10 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 IX10 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)>
```

9. (Optional) Set the text to prepend to the forwarded message. Two variables can be included in the prepended text:

- **%s**: Includes the IX10 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 IX10 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 IX10 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:

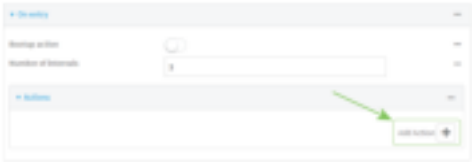


7. Define actions to be taken when the device's location triggers a geofence event:
   - To define actions that will be taken when the device enters the geofence, or is inside the geofence when it boots:
     a. Click to expand **On entry**.



     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**.



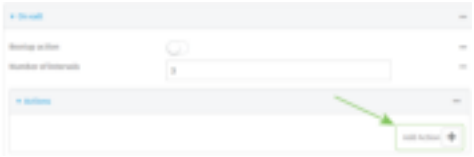
b. (Optional) Enable **Bootup action** to configure the device to perform the **On exit** actions if the device is inside the geofence when it boots.

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 IX10 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 IX10 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 IX10 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 : Tue, 15 June 2021 8:04:23 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 IX10 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 IX10 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 IX10 gateway allows for communication between buses and and networks that use the Modbus protocol.

This section contains the following topics:

Configure the Modbus gateway 366
Show Modbus gateway status and statistics 379

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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device can also be configured to use Network Time Protocol (NTP). In this configuration, the device serves as an NTP server, providing NTP services to downstream devices. See Network Time Protocol for more information about NTP server support.

Configure the system time

This procedure is optional.

The IX10 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, as well as configuring additional NTP servers.

Required Configuration Items

- The time zone for the IX10 device.
- At least one upstream NTP server for synchronization.

Additional Configuration Options

- Additional upstream NTP servers.

WebUI

1. Log into the IX10 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 IX10 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 IX10 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 IX10 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
```

   - 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.

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 IX10 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.

Show status and statistics of the NTP server

You can display status and statistics for active NTP servers

WebUI

1. Log into the IX10 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 IX10 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 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 IX10 device's NTP service.
- The time zone setting, if the default setting of UTC is not appropriate.

To configure the IX10 device's NTP service:

WebUI

1. Log into the IX10 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 IX10 device's NTP service by clicking **Enable**.
5. (Optional) Configure the access control list to limit downstream access to the IX10 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 IX10 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 IX10 device can use the NTP service.

6. (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.

7. (Optional) Configure the system time zone. The default is **UTC**.
   a. Click **System** > **Time**
   b. Select the **Timezone** for the location of your IX10 device.
8. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX10 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
```

   - 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. (Optional) Configure the access control list to limit downstream access to the IX10 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 IX10 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 IX10 device can use the NTP service.

6. (Optional) Set the timezone for the location of your IX10 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)>
```

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 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 IX10 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. Type the **Source port**. Ensure the port is not used by another protocol.
8. Select a **Source interface** where multicast packets will arrive.
9. Select a **Destination interface** that the IX10 device will use to send mutlicast packets.
10. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX10 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 port for the route. Ensure the port is not used by another protocol.

```
(config service multicast test)> port port
(config service multicast test)>
```

7. 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)>
```

8. Set the destination interface that the IX10 device will use to send mutlicast packets.

```
(config service multicast test)> interface interface
(config service multicast test)>
```

   a. Use the **?** to determine available interfaces:
   b. Set the interface. For example:

```
(config service multicast test)> interface /network/interface/eth1
(config service multicast test)>
```

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 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 IX10 device to use mDNS.

WebUI

1. Log into the IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device's iPerf3 server.

Required configuration items

- Enable the iPerf server on the IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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:

Configure applications to run automatically 408
Run a Python application at the shell prompt 414
Start an interactive Python session 416
Digidevice module 418
Use Python to access serial ports 444
Use the Paho MQTT python library 445
Use the local REST API to configure the IX10 device 448
Stop a script that is currently running 452
Show script information 452

Configure applications to run automatically

You can configure an 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 Python application.
- Enable the Python application to be run automatically.
- Select whether the application 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 application.
- The action to take if the Python application finishes. The actions that can be taken are:
  - None.
  - Restart the script.
  - Reboot the device.
- The arguments for the Python application.
- Whether to write the application output and errors to the system log.
- The memory available to be used by the application.
- Whether the script should run one time only.

Task one: Upload the application

WebUI

1. Log into the IX10 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 IX10 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 IX10 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 IX10 device.
   - *local-path* is the location on the IX10 device where the copied file will be placed.

   For example:

   To upload a Python application from a remote host with an IP address of 192.168.4.1 to the /etc/config/scripts directory on the IX10 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 Python applications 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 IX10 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 schedule script configuration window is displayed.



Scheduled 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 selected, 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. 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**.

11. **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.
12. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX10 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**|**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.

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 IX10 device:

WebUI

a. Log into the IX10 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 IX10 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 IX10 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 IX10 device.
- *local-path* is the location on the IX10 device where the copied file will be placed.

For example:

To upload a Python application from a remote host with an IP address of 192.168.4.1 to the /etc/config/scripts directory on the IX10 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 Python applications by using the **vi** command when logged in with shell access.

2. Log into the IX10 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
```

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 IX10 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()**.

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:

Use digidevice.cli to execute CLI commands 419
Use digidevice.datapoint to upload custom datapoints to Digi Remote Manager 420
Use digidevice.config for device configuration 422
Use Python to respond to Digi Remote Manager SCI requests 425
Use digidevice runtime to access the runtime database 433
Use Python to upload the device name to Digi Remote Manager 435
Use Python to access the device location data 437
Use Python to send and receive SMS messages 441

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 IX10 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 IX10
  Serial Number            : IX10-000065
  SKU                      : IX10
  Hostname                 : IX10
  MAC Address              : DF:DD:E2:AE:21:18

  Hardware Version         : 50001947-01 1P
  Firmware Version         : 21.5.56.106
  Alt. Firmware Version    : 21.5.56.106
  Alt. Firmware Build Date : Tue, 15 June 2021 8:04:23
  Bootloader Version       : 19.7.23.0-15f936e0ed

  Current Time             : Tue, 15 June 2021 8:04:23 +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 IX10 CLI commands

Get help executing a CLI command from Python by accessing help for **cli.execute**:

1. Log into the IX10 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 IX10 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()**.

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**:

1. Log into the IX10 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 **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 IX10 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:

```
>>> cfg = config.load()
>>> print(cfg)
```

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()
>>> 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 IX10 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 IX10 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 IX10 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 IX10 device to create a response

1. Log into the IX10 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 applications 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 IX10 WebUI as a user with full Admin access rights.

   ii. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

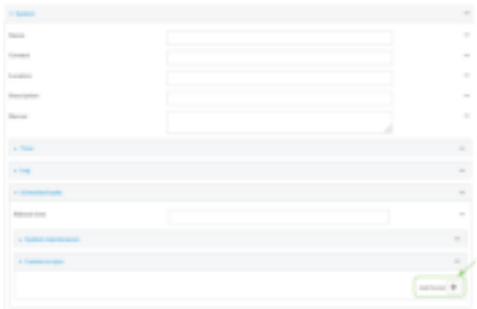
   

   The **Configuration** window is displayed.



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 IX10 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 IX10 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 IX10
          Serial Number            : IX10-000068
          Hostname                 : IX10
          MAC                      : 00:40:D0:13:35:36

          Hardware Version         : 50001959-01 A
          Firmware Version         : 21.5.56.106
          Bootloader Version       : 1
          Firmware Build Date      : Tue, 15 June 2021 8:04:23
          Schema Version           : 461

          Timezone                 : UTC
          Current Time             : Tue, 15 June 2021 8:04:23
          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 IX10
          Serial Number            : IX10-000023
          Hostname                 : IX10
          MAC                      : 00:40:D0:26:79:1C

          Hardware Version         : 50001959-01 A
          Firmware Version         : 21.5.56.106
          Bootloader Version       : 1
          Firmware Build Date      : Tue, 15 June 2021 8:04:23
          Schema Version           : 461

          Timezone                 : UTC
          Current Time             : Tue, 15 June 2021 8:04:23
          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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 access the device location data

The **location** submodule enables access to the location data for the IX10 device.

The module takes a snapshot of location data stored in the runt database. The location data snapshot can be subsequently updated by using the update method.

Determine if the device's location

1. Log into the IX10 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 **location** submodule:

```
>>> from digidevice import location
```

4. Use the **valid_fix** object to determine if the device has a valid fix:

```
>>> loc = location.Location()
>>> loc.valid_fix
True
>>>
```

5. Use the **position** object to return the device's position:

```
>>> loc.position
(44.926195299999998, -93.397084499999999, 292.39999399999999)
>>>
```

   The coordinates are returned in the following order:

   latitude, *longitude*, *altitude*

   altitude is in meters.

6. You can also return only one of the coordinate positions:
   - Use the **latitude** object to return the latitude:

```
>>> loc.latitude
44.926195299999998
>>>
```

   - Use the **longitude** object to return the longitude:

```
>>> loc.longitude
-93.397084499999999
>>>
```

   - Use the **altitude** object to return the altitude, in meters:

```
>>> loc.altitude
292.39999399999999
>>>
```

7. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Update the location data

The **location** submodule takes a snapshot of the current location and stores it in the runtime database. You can update this snapsot:

1. Log into the IX10 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 **location** submodule:

```
>>> from digidevice import location
```

4. Update the location object with the latest location data:

```
>>> loc = location.Location()
>>> loc.position
>>> (44.926195299999998, -93.397084499999999, 292.39999399999999)
>>> loc.update()
>>> loc.position
44.926231, -93.397923, 289.439229
>>>
```

5. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Output location data in json format

The **location** submodule takes a snapshot of the current location and stores it in the runtime database. You can update this snapsot

1. Log into the IX10 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 **json** submodule:

```
>>> import json
```

4. Import the **location** submodule:

```
>>> from digidevice import location
```

5. Print the location data in json format:

```
>>> geojson_data = location.Location().geojson
>>> print(json.dumps(geojson_data, indent=4))
{
    "type": "Feature",
    "geometry": {
        "type": "Point",
        "coordinates" [
            44.926195299999998,
            -93.397084499999999,
            273.20001200000002
        ]
    },
    "properties": {
        "direction": "None",
        "horizontal_velocity": "0.0",
        "latitude.deg_min_sec": "44* 54' 45.586\" N",
        "longitude.deg_min_sec": "93* 33' 52.334\" W",
        "num_satellites": "12",
        "quality": "Standard GNSS (2D/3D)",
        "selected_source_idx": "0",
        "source": "USB (/dev/ttyACM0)",
        "source_idx.0.altitude": "273.200012",
        "source_idx.0.direction": "None",
        "source_idx.0.horizontal_velocity": "0.195489",
        "source_idx.0.label": "usb",
        "source_idx.0.latitude": "44.902662",
        "source_idx.0.latitude.deg_min_sec": "44* 55' 45.065\" N",
        "source_idx.0.longitude": "-93.560648",
        "source_idx.0.longitude.deg_min_sec": "93* 16' 52.966\" W",
        "source_idx.0.num_satellites": "12",
        "source_idx.0.quality": "Standard GNSS (2D/3D)",
        "source_idx.0.utc_date_time": "Jun-15-2021 8:04:23",
        "source_idx.0.vertical_velocity": "0.0",
        "source_idx.1.label": "gnss",
        "source_idx.1.quality": "No Fix / Invalid",
        "state": "Enabled, signal",
        "utc_date_time": "Jun-15-2021 8:04:23",
        "vertical_velocity": "0.0"
    }
}
>>>
```

6. Use **Ctrl-D** to exit the Python session. You can also exit the session using **exit()** or **quit()**.

Help for the digidevice location module

Get help for the digidevice location module:

1. Log into the IX10 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 **location** submodule:

```
>>> from digidevice import location
>>>
```

4. Use the help command with **location**:

```
>>> help(location)
Help on module digidevice.location in digidevice:

NAME
digidevice.location - digidevice.location - API for accessing location
data
...
```

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 IX10 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 IX10 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 applications 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/python

# DIGI SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED
# TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
# PARTICULAR PURPOSE. THE SOFTWARE AND ACCOMPANYING DOCUMENTATION, IF ANY,
# PROVIDED HEREUNDER IS PROVIDED "AS IS" AND WITHOUT WARRANTY OF ANY KIND.
# DIGI HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES,
# ENHANCEMENTS, OR MODIFICATIONS.
#
# IN NO EVENT SHALL DIGI BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT,
# SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS,
# ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF
# DIGI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
"""
NOTE: This code allows SMS messages to be sent and received and should be
reviewed before implementing. If you allow SMS incoming messages to modify or
run commands on the device, all incoming messages should be encrypted and
validated prior to execution.
"""
import os
import threading
import sys
from digidevice.sms import Callback, send
COND = threading.Condition()

def sms_test_callback(sms, condtion):
    print(f"SMS message from {sms['from']} received")
    print(sms)
    condition.acquire()
    condition.notify()
    condition.release()

def send_sms(destination, msg):
    print("sending SMS message", msg)
    if len(destination) > 10:
        destination = "+1" + destination
        # NOTE: The number must include either the + prefix or leading zeros
(e.g, either +1 or 00).
    send(destination, msg)

if __name__ == '__main__':
    if len(sys.argv) > 1:
```

```
        dest = sys.argv[1]
    else:
        dest = '+15005550006'
        # NOTE: The number must include either the + prefix or leading zeros
(e.g, either +15005550006 or 0015005550006).
    my_callback = Callback(sms_test_callback, COND)
    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 IX10 device that are configured to be in Application mode. See Configure the serial port for information about configuring a serial port in Application mode.

To use Python to access serial ports:

1. Log into the IX10 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 IX10 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 some data from updates the device firmware, then publishes information about DHCP clients and system information to the MQTT server.

```
"""
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)
```

Use the local REST API to configure the IX10 device

Your IX10 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 IX10 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 IX10 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 add items to a list array

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 }
$
```

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 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 }
$
```

Stop a script that is currently running

You can stop a script that is currently running by using the system script stop *name* command.

Command line

1. Log into the IX10 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:

```
)> system script stop
0              script1
1              script2
>
```

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.

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 IX10 WebUI as a user with Admin access.
2. At the **Status** page, click **Scripts**.

   The **Scripts** page displays:



Command line

1. Log into the IX10 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            Script  Enabled  Status  Run time
 -----  ---------------  ------  -------  ------  --------
 0      mgmt_intf_fixup  #!/bin/sh -x
 send_mgmt_intf_update() {
   pri="$(runt get network.mgmt.log.default)"
   if [ "$pri" = 'pri' ]; then
     default_net="$(runt dump network.route.default | grep -m 1 -o
"interface_.*=" | cut -f2 -d'_' | tr -d '=')"
     if [ -n "$default_net" ]; then
       default_intf="$(runt get network.interface.${default_net}.device)"
          runt set network.mgmt.log.intf "$default_intf"
       fi
       log=$(runt log network.mgmt.log)
       accns_log network_mgmt "${log:+type=mgmt~}$log"
   fi
}
...
>
```

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.

User authentication

This chapter contains the following topics:

IX10 user authentication 455
User authentication methods 455
Authentication groups 463
Local users 473
Terminal Access Controller Access-Control System Plus (TACACS+) 486
Remote Authentication Dial-In User Service (RADIUS) 492
LDAP 498
Configure serial authentication 505
Disable shell access 507
Set the idle timeout for IX10 users 509
Example user configuration 512

IX10 user authentication

User authentication on the IX10 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 IX10.	■ **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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device by using the serial console.

Preconfigured authentication groups

The IX10 device has two preconfigured authentication groups:

- The **admin** group is configured by default to have full **Admin access** and **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.
- 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 465
Add an authentication group 467
Delete an authentication group 471

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 IX10 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.



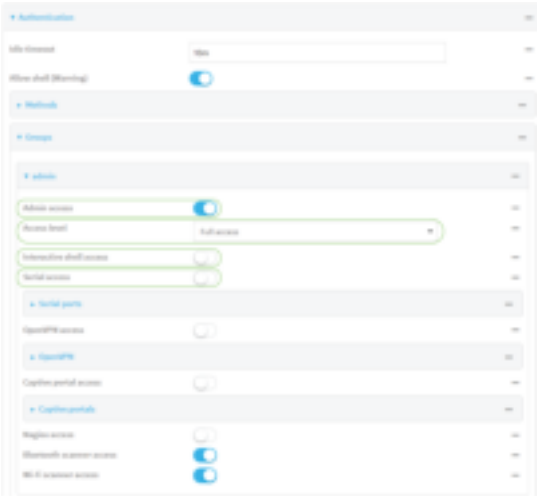
3. Click **Authentication** > **Groups**.
4. Click the authentication group to be changed, either **admin** or **serial**, to expand its configuration node.
5. Click the box next to the following options, as appropriate, to enable or disable access rights for each:
   - **Admin access**

     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 IX10 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 IX10 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 IX10 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

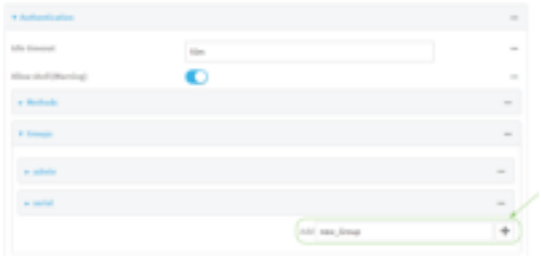
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



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 IX10 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 OpenVPN access. See for further information.
7. (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.
8. (Optional) Enable users that belong to this group to query the device for Nagios monitoring by checking the box next to **Nagios access**.
9. (Optional) Enable users that belong to this group to access the Bluetooth scanning service by checking the box next to **Bluetooth scanner access**.
10. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 474
Configure a local user 476
Delete a local user 483

Change a local user's password

To change a user's password:

WebUI

1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



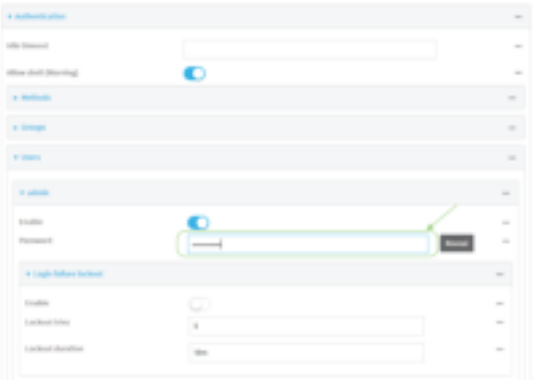
The **Configuration** window is displayed.



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 IX10 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

- 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

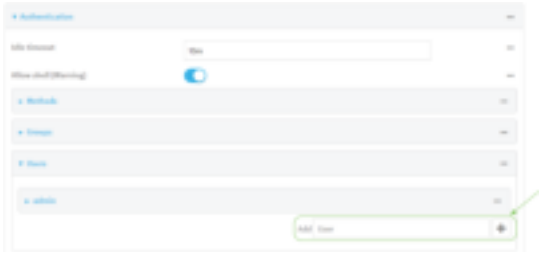
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



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. 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.
6. 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**.

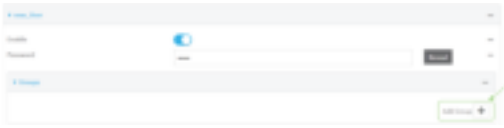
The minimum value is 1 second, and the maximum is 15 minutes. The default is 15 minutes.

7. 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.

8. (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 ✚.

9. (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. To display the QR code for the secret key, click **...** next to the field label and select **Show secret key QR code**.

iii. Copy the secret key, or scan or copy the QR code, for use with an application or mobile device to generate passcodes.

Note To copy the QR code, right-click the QR code and select your browser's save image functionality.

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.

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



Command line

1. Log into the IX10 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. 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)>
   ```

5. 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.

6. 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)>
   ```

7. (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)>
   ```

8. (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.

9. Save the configuration and apply the change:

```
(config auth user new 2fa scratch_code)> 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.

Delete a local user

To delete a user from your IX10:

WebUI

1. Log into the IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 487
TACACS+ server failover and fallback to local authentication 488
Configure your IX10 device to use a TACACS+ server 488

TACACS+ user configuration

When configured to use TACACS+ support, the IX10 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 IX10 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 IX10"
       pap = cleartext password1
       service = system {
         groupname = admin,serial
       }
}
user = user2 {
       name ="User2 for IX10"
       pap = cleartext password2
       service = system {
         groupname = serial
       }
}
```

   The **groupname** attribute is optional. If used, the value must correspond to authentication groups configured on your IX10. Alternatively, if the user is also configured as a local user on the IX10 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 IX10 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 IX10 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 IX10 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 IX10 device to use a TACACS+ server

This section describes how to configure a IX10 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 IX10 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.
- 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 IX10 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 IX10 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 IX10 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 IX10 configuration.
8. 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.
9. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX10 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 IX10 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 IX10 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 IX10 configuration.

```
(config)> auth tacacs+ service service-name
(config)>
```

6. 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.
7. 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)>
```

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.

Remote Authentication Dial-In User Service (RADIUS)

Your IX10 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 IX10 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 IX10 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 493
RADIUS server failover and fallback to local configuration 493
Configure your IX10 device to use a RADIUS server 494

RADIUS user configuration

When configured to use RADIUS support, the IX10 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 IX10 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 IX10. Alternatively, if the user is also configured as a local user on the IX10 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 IX10 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 IX10 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 IX10 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 IX10 device to use a RADIUS server

This section describes how to configure a IX10 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 IX10 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 IX10 device by using the WebUI, the default value is for NAS ID is **httpd**.
  - If you are access the IX10 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

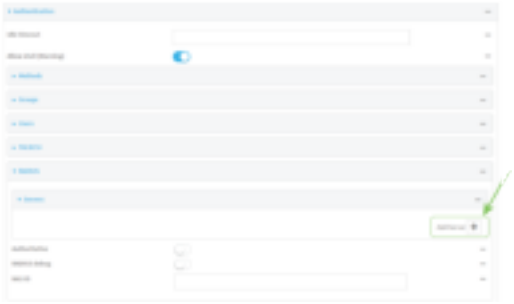
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



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 IX10 device by using the WebUI, the default value is for NAS ID is **httpd**.
- If you are accessing the IX10 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 IX10 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 IX10 device by using the WebUI, the default value is for NAS ID is **httpd**.
   - If you are accessing the IX10 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 IX10 device supports LDAP (Lightweight Directory Access Protocol), a protocol used for directory information services over an IP network. LDAP can be used with your IX10 device for centralized authentication and authorization management for users who connect to the device. With LDAP support, the IX10 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 IX10 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 499
LDAP server failover and fallback to local configuration 500
Configure your IX10 device to use an LDAP server 500

LDAP user configuration

When configured to use LDAP support, the IX10 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 IX10 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 IX10 device.
   - The **ou** attribute is optional. If used, the value must correspond to authentication groups configured on your IX10. Alternatively, if the user is also configured as a local user on the IX10 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 IX10 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 IX10 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 IX10 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 IX10 device to use an LDAP server

This section describes how to configure a IX10 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 IX10 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

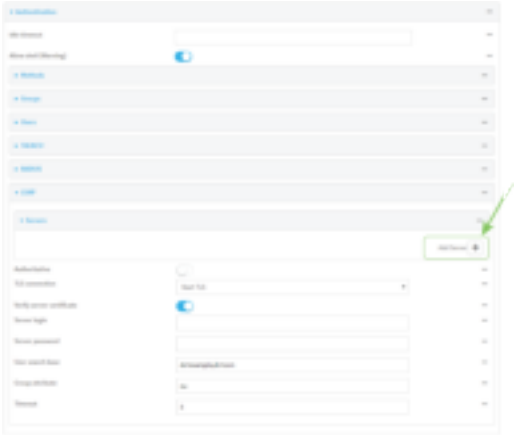
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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

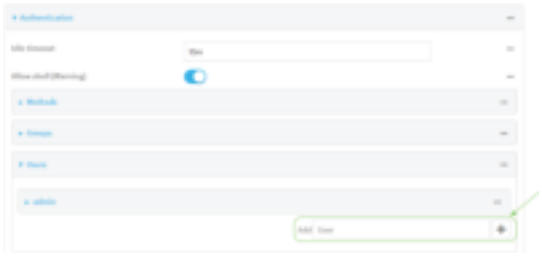
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device, **admin**, is identified in the **groupname** parameter.

   c. Save and close the **tac_plus.conf** file.
3. Log into the IX10 WebUI as a user with full Admin access rights.
4. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



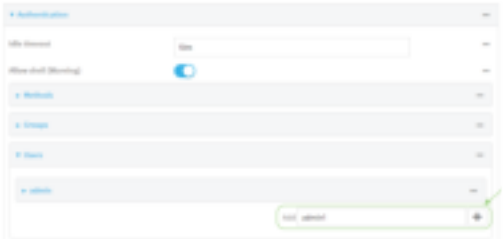
The **Configuration** window is displayed.



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**.

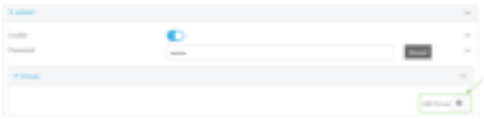


6. Create the local user:
   a. Click **Authentication** > **Users**.
   b. In **Add User:**, type **admin1** and click ✚.

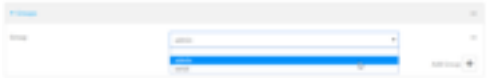


   c. For **password**, type **password1**.

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 IX10 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 IX10 device, **admin**, is identified in the **groupname** parameter.

   c. Save and close the **tac_plus.conf** file.
3. Log into the IX10 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 522
Port forwarding rules 527
Packet filtering 535
Configure custom firewall rules 543
Configure Quality of Service options 544

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 IX10 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 IX10.
- **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

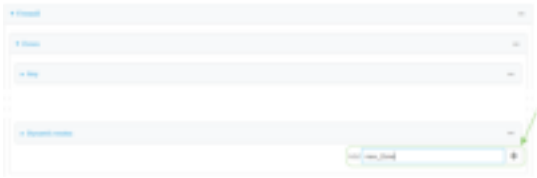
1. Log into the IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.



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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device on the binding's interface. By default, the IX10 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 IX10 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 IX10 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 IX10 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 IX10 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.

System administration

This chapter contains the following topics:

Review device status 556
Configure system information 557
Update system firmware 559
Update cellular module firmware 565
Reboot your IX10 device 568
Erase device configuration and reset to factory defaults 571
Configuration files 576
Schedule system maintenance tasks 581
Disable device encryption 588
Configure the speed of your Ethernet port 591

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 IX10 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 IX10 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 IX10
  Serial Number            : IX10-000065
  SKU                      : IX10
  Hostname                 : IX10
  MAC Address              : DF:DD:E2:AE:21:18

  Hardware Version         : 50001947-01 1P
  Firmware Version         : 21.5.56.106
  Alt. Firmware Version    : 21.5.56.106
  Alt. Firmware Build Date : Tue, 15 June 2021 8:04:23
  Bootloader Version       : 19.7.23.0-15f936e0ed

  Current Time             : Tue, 15 June 2021 8:04:23 +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 IX10 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 IX10
  Serial Number            : IX10-000065
  SKU                      : IX10
  Hostname                 : IX10
  MAC Address              : DF:DD:E2:AE:21:18

  Hardware Version         : 50001947-01 1P
  Firmware Version         : 21.5.56.106
  Alt. Firmware Version    : 21.5.56.106
  Alt. Firmware Build Date : Tue, 15 June 2021 8:04:23
  Bootloader Version       : 19.7.23.0-15f936e0ed
  Schema Version           : 715

  Timezone                 : UTC
  Current Time             : Tue, 15 June 2021 8:04:23 +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 IX10 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 IX10 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 IX10 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 IX10."
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 IX10 operating system firmware images consist of a single file with the following naming convention:

***platform-version*.bin**

For example, **IX10-21.5.56.106.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 IX10 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.

Update firmware over the air (OTA) from the Digi firmware server

WebUI

1. Log into the IX10 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 IX10 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 21.2.39.67
Checking for latest IX10 firmware...
Newest firmware version available to download is '21.5.56.106'
Device firmware update from '21.2.39.67' to '21.5.56.106' is needed
>
```

3. Use the **modem firmware ota list** command to list available firmware on the Digi firmware repository.

```
> system firmware ota list
21.2.39.67
21.5.56.106
>
```

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 '21.5.56.106'...
Downloaded firmware /tmp/cli_firmware.bin remaining
Applying firmware version '21.5.56.106'...
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 21.5.56.106
Downloading firmware version '21.5.56.106'...
Downloaded firmware /tmp/cli_firmware.bin remaining
Applying firmware version '21.5.56.106'...
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 IX10 operating system firmware from the Digi Support FTP site to your local machine.
2. Log into the IX10 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 IX10 operating system firmware from the Digi Support FTP site to your local machine.
2. Log into the IX10 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 IX10 device.

- *local-path* is the location on the IX10 device where the copied file will be placed.

For example:

```
> scp host 192.168.4.1 user admin remote /home/admin/bin/IX10-
21.5.56.106.bin local /etc/config/ to local
admin@192.168.4.1's password: adminpwd
IX10-21.5.56.106.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 IX10-
21.5.56.106.bin
-rw-r--r--    1 root     root          2580 May 16 16:44 accns.json
...
>
```

5. Update the firmware by entering the update firmware command, specifying the firmware file name:

```
> system firmware update file IX10-21.5.56.106.bin
36632K
netflash: got "/etc/config/IX10-21.5.56.106.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 IX10's command line as a user with Admin access and verify the running firmware version by entering the show system command.

```
> show system

 Hostname                : IX10
 FW Version              : 21.5.56.106
 MAC                     : 0040FF800120
 Model                   : Digi IX10
 Current Time            : Tue, 15 June 2021 8:04:23 +0000
 Uptime                  : 42 seconds (42s)

>
```

Dual boot behavior

By default, the IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device. Firmware should be uploaded to /opt/*MODEM_MODEL*/Custom_Firmware, for example, /opt/LM940/Custom_Firmware. Modem firmware can be downloaded from Digi at https://ftp1.digi.com/support/firmware/dal/carrier_firmware/. See Use the scp command for information about uploading files to the IX10 device.

1. Log into the IX10 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 IX10 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 IX10 device

You can reboot the IX10 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 IX10 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 IX10 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 IX10 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 IX10 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.

You can also reset the device to the default configuration without removing scripts, keys, and logfiles by using the **revert** command.

WebUI

1. Log into the IX10 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 IX10 by using the serial port or by using an Ethernet cable to connect the IX10 **ETH** port to your PC.
   b. Log into the IX10:

      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 IX10 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 IX10 by using the serial port or by using an Ethernet cable to connect the IX10 **ETH** port to your PC.
   b. Log into the IX10:

      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 IX10 by using the serial port or by using an Ethernet cable to connect the IX10 **ETH** port to your PC.
   b. Log into the IX10:

      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 IX10 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 IX10 device to use custom factory default settings

You can configure your IX10 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.

Required configuration items

- Custom factory default file

WebUI

1. Log into the IX10 WebUI as a user with Admin access.
2. Configure your IX10 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, into the **/opt** directory.

See Upload and download files for information about uploading a file to the device.

If you use th Web UI to upload the file, you will need to use the **mv** command at the Admin CLI to move the file to the **/opt** directory. For example:

```
> mv /etc/config/scripts/custom-default-config.bin /opt
>
```

Command line

1. Log into the IX10 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 /opt/custom-default-config.bin type archive
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.

Configuration files

The IX10 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 IX10 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 IX10 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 IX10 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 IX10 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.

1. Log into the IX10 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 IX10 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 IX10'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 IX10 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 IX10 device by using a backup from the device, or a backup from a similar device.

WebUI

1. Log into the IX10 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 IX10 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 IX10 device.
   - *local-path* is the location on the IX10 device where the copied file will be placed.

For example:

```
> scp host 192.168.4.1 user admin remote /home/admin/bin/backup-archive-
0040FF800120-21.5.56.106-19.23.42.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 IX10'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-0040FF800120-21.5.56.106-
19.23.42.bin
```

Schedule system maintenance tasks

You can configure tasks and custom scripts to be run during a specified maintenance window.

Required configuration items

- The time that the system maintenance tasks will start.
- The duration window during which the system maintenance tasks can run.
- The frequency (either daily or weekly) that the tasks will run.
- The tasks to be performed. Options are:
  - Modem firmware update.
  - Configuration check.

Additional configuration items

- Custom scripts that should be run as part of the configuration check.

WebUI

1. Log into the IX10 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. 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.

5. 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**.
6. For **Frequency**, select either **Daily** or **Weekly** for the frequency that the maintenance tasks should be run.
7. (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.
8. (Optional) Click to enable **Configuration check** to allow for the configuration to be updated, including by custom scripts, during the maintenance window.
9. (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**.

10. (Optional) Enable **Allow scheduled scripts to handle SMS** to allow scheduled scripts to handle SMS messages.
11. (Optional) To schedule custom scripts:
    a. Click **Custom scripts**.

    Note This feature does not provide syntax or error checking. Certain commands can render the device inoperable. Use with care. Scripts created here are also automatically entered in **Configuration** > **Applications**.

    b. For **Add Script**, click ✚.

    The schedule script configuration window is displayed.

    Scheduled scripts are enabled by default. To disable, click **Enable** to toggle off.
    c. (Optional) For **Label**, provide a label for the script.
    d. 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 selected, 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.

e. 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**).

f. Script logging options:

i. Click to enable **Log script output** to log the script's output to the system log.

ii. 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.

g. 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**}.

h. 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**.

i. **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.

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



Command line

1. Log into the IX10 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. Schedule system maintenance:
   a. 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 from HH:MM
      (config)>
      ```

      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.

   b. 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 length num
      (config)>
      ```

      where *num* is any whole number between **0** and **24**.

   c. Configure the frequency that the maintenance tasks should be run:

      ```
      (config)> system schedule maintenance frequency value
      (config)>
      ```

      where *value* is either **daily** or **weekly**. **Daily** is the default.

4. 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.

   ```
   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.

5. (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.

6. (Optional) Allow scheduled scripts to handle SMS messages:

```
(config)> system schedule sms_script_handling true
(config)>
```

7. (Optional) Schedule custom scripts:
   a. 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)>
```

   b. (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.

   c. 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.

d. 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**).

e. 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.

f. 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**}.

g. 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**.

h. **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)>
```

8. Allow for the configuration to be updated, including by custom scripts, during the maintenance window:

```
system schedule maintenance config_check value
(config)>
```

where *value* is either **true** or **false**. **yes** or **no**, and **1** or **0** are also allowed.

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.

Disable device encryption

You can disable the cryptography on your IX10 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 IX10 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.

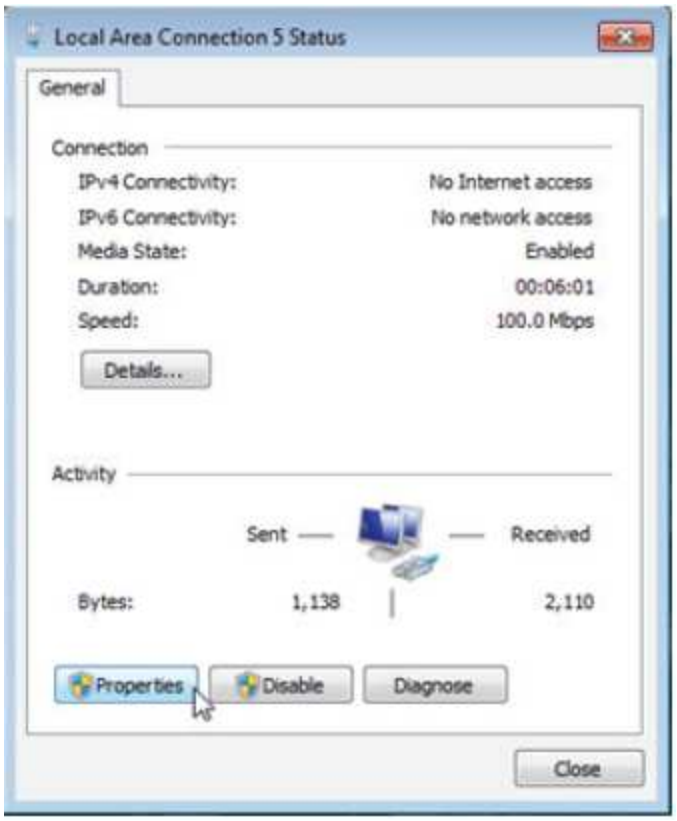
   Depending on your device configuration, you may be presented with an **Access selection menu**. Type **quit** to disconnect from the device.

Re-enable cryptography after it has been disabled.

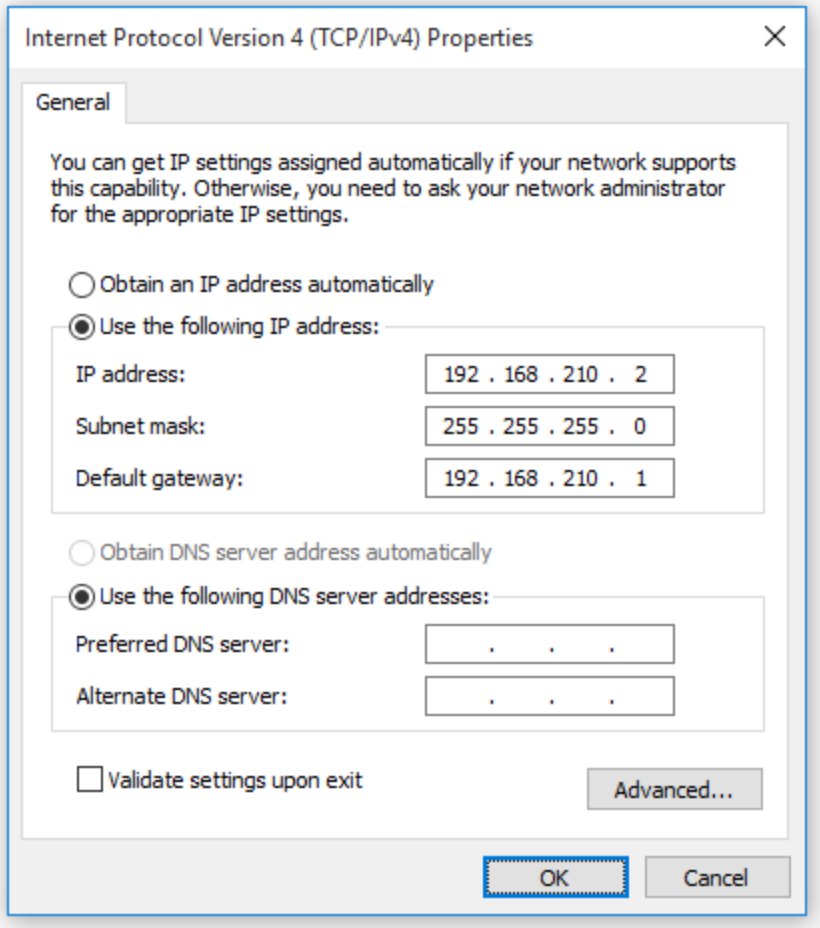
To re-enable cryptography:

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 IX10 device.
3. Open a telnet session and connect to the IX10 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 IX10 device's Ethernet port.

WebUI

1. Log into the IX10 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 IX10 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 595
Configure NetFlow Probe 602

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 IX10 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, it adds an estimated 50MB of data usage for the device by reporting the metrics to 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 608
Configure Digi Remote Manager 608
Collect device health data and set the sample interval 615
Log into Digi Remote Manager 618
Use Digi Remote Manager to view and manage your device 619
Add a device to Digi Remote Manager 620
View Digi Remote Manager connection status 620
Configure multiple devices using profiles 621
Learn more 621

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*.

Configure Digi Remote Manager

By default, your IX10 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 IX10 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 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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**.



Device health data upload is enabled by default. To disable, click to toggle off **Enable Device Health samples upload**.

4. For **Health sample interval**, select the interval between health sample uploads.
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. (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.
7. Click **Apply** to save the configuration and apply the change.



Command line

1. Log into the IX10 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.

Log into Digi Remote Manager

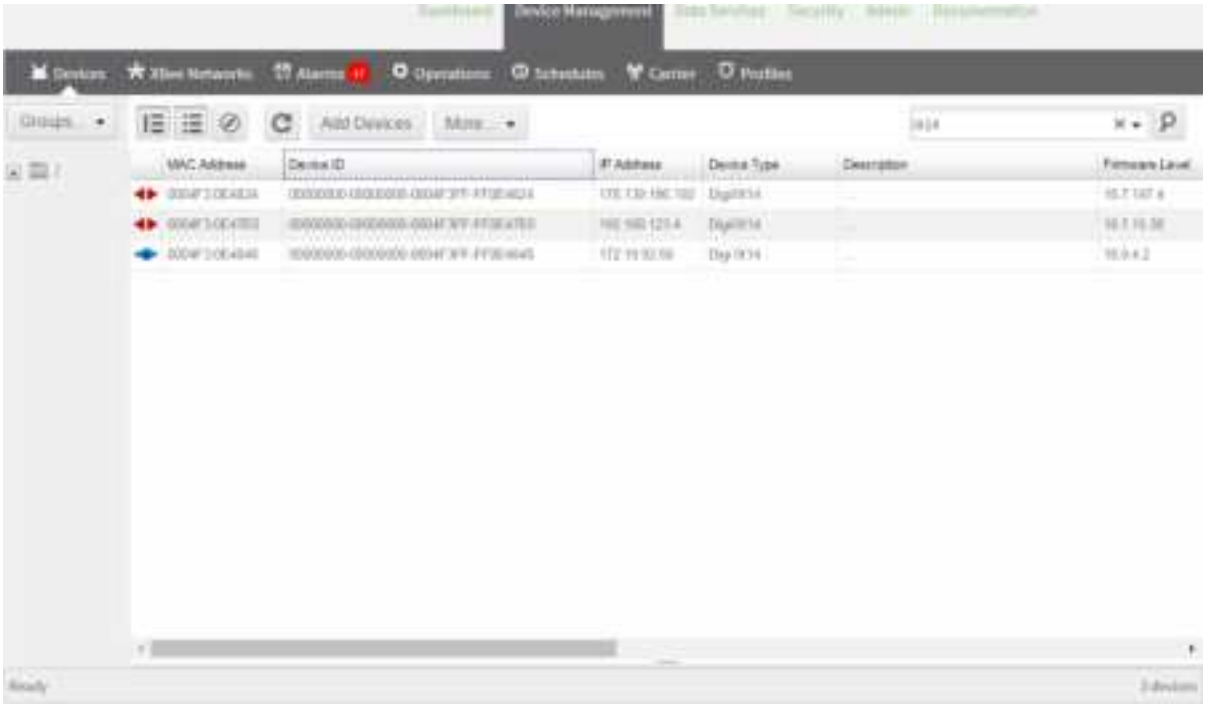
To start Digi Remote Manager

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.

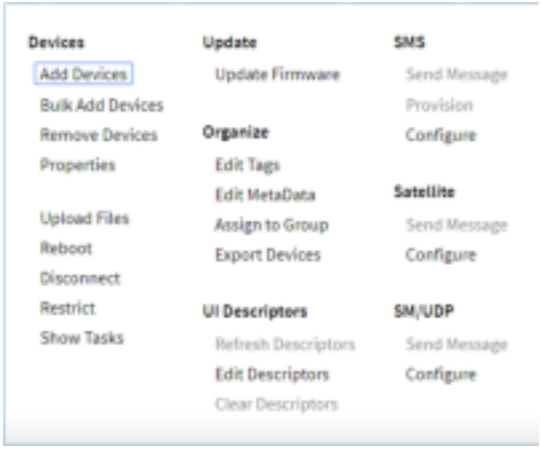
Use Digi Remote Manager to view and manage your device

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 IX10 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 IX10 WebUI as a user with Admin access.
2. The dashboard includes a Digi Remote Manager status pane:

Command line

1. Log into the IX10 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 cloud command 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 IX10 routers. Typically, if you want to provision multiple IX10 routers:

1. Using the IX10 local WebUI, configure one IX10 router to use as the model configuration for all subsequent IX10s you need to manage.
2. Register the configured IX10 device in your Digi Remote Manager account.
3. In Digi Remote Manager, create a profile based on the configured IX10.
4. Apply the profile to the IX10 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.

Learn more

- For information on using Digi Remote Manager to configure and manage IX10 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 IX10 local file system 623
Display directory contents 623
Create a directory 624
Display file contents 625
Copy a file or directory 625
Move or rename a file or directory 626
Delete a file or directory 627
Upload and download files 628

The IX10 local file system

The IX10 local file system has approximately TBD 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 device

To copy a file from a remote host to the IX10 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 IX10 device.
- *local-path* is the location on the IX10 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 IX10 device, issue the following command:

```
> scp host 192.168.4.1 user admin remote /home/admin/bin/IX10-21.5.56.106.bin
local /etc/config/scripts to local
admin@192.168.4.1's password: adminpwd
IX10-21.5.56.106.bin                    100%    36MB    11.1MB/s        00:03
>
```

Transfer a file from the IX10 device to a remote host

To copy a file from the IX10 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 IX10 device.

For example:

To copy a support report from the IX10 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-0040D0133536-21-06-15-
   8:04:23.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:D0:13:35:36-21-06-15-8:04:23.bin to remote
   admin@192.168.4.1's password: adminpwd
   support-report-0040D0133536-21-06-15-8:04:23.bin
   >
   ```

Upload and download files using SFTP

Transfer a file from a remote host to the IX10 device

This example uploads firmware from a remote host to the IX10 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 IX10-21.5.56.106
Uploading IX10-21.5.56.106 to IX10-21.5.56.106
IX10-21.5.56.106
      100%  24M  830.4KB/s   00:00
sftp> exit
$
```

Transfer a file from the IX10 device to a remote host

This example downloads a file named **test.py** from the IX10 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
$
```

Diagnostics

This chapter contains the following topics:

Generate a support report 633
View system and event logs 634
Configure syslog servers 638
Configure options for the event and system logs 641
Analyze network traffic 646
Use the ping command to troubleshoot network connections 664
Use the traceroute command to diagnose IP routing problems 664

Generate a support report

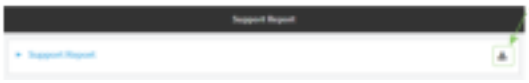
To generate and download a support report:

WebUI

1. Log into the IX10 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 IX10 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-0040D0133536-21-06-15-
8:04:23.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:D0:13:35:36-21-06-15-8:04:23.bin to remote
admin@192.168.4.1's password: adminpwd
support-report-0040D0133536-21-06-15-8:04:23.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

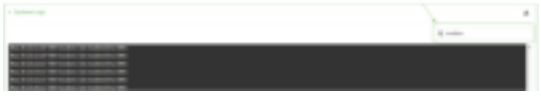
1. Log into the IX10 WebUI as a user with Admin access.
2. On the main menu, click **System** > **Logs**.



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 IX10 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 show log at the Admin CLI prompt:

```
> show log

Timestamp        Message
--------------  ------------------------------------------------------------
----
Nov 26 21:54:34  IX10 netifd: Interface 'interface_wan' is setting up now
Nov 26 21:54:35  IX10 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  IX10 netifd: Interface 'interface_wan' is setting up now
Nov 26 21:54:35  IX10 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 IX10 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 IX10 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 show event 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 647
Example filters for capturing data traffic 656
Capture packets from the command line 657
Stop capturing packets 658
Show captured traffic data 659
Save captured data traffic to a file 660
Download captured data to your PC 661
Clear captured data 662

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 IX10 WebUI as a user with full Admin access rights.
2. On the menu, click **System**. Under **Configuration**, click **Device Configuration**.

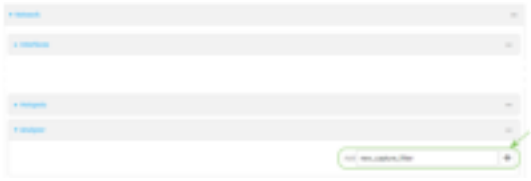
   

   The **Configuration** window is displayed.

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.

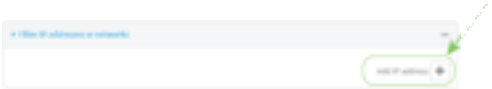


5. (Optional) Add a filter type:
   a. Click to expand **Filter**.



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 IX10 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 IX10 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 IX10 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 IX10 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 : Jun-15-2021 8:04:23.287682, 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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 IX10 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**.

Digi IX10 regulatory and safety statements

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**.

Federal Communication (FCC) Part 15 Class B

Radio Frequency Interference (RFI) (FCC 15.105)

The Digi IX10 has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and the receiver.
- Connect the equipment into an outlet that is on a circuit different from the receiver.
- Consult the dealer or an experienced radio/TV technician for help.

Labeling Requirements (FCC 15.19)

IX10 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 IX10 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

Wi-Fi frequency bands

Frequency bands	Maximum transmit power
13 overlapping channels at 22 MHz or 40 MHz wide spaced at 5 MHz Centered at 2.412 MHz to 2.472 MHz	651.784 mW
165 overlapping channels at 22 MHz or 40 MHz or 80 MHz wide spaced at 5 MHz Centered at 5180 MHz to 5825 MHz	351.295 mW

Innovation, Science, and Economic Development Canada (IC) certifications

This digital apparatus does not exceed the Class B limits for radio noise emissions from digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications.

Le present appareil numerique n’emet pas de bruits radioelectriques depassant les limites applicables aux appareils numeriques de la class B prescrites dans le Reglement sur le brouillage radioelectrique edicte par le ministere des Communications du Canada.

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.

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! RISK OF EXPLOSION IF BATTERY IS REPLACED BY INCORRECT BATTERY TYPE. DISPOSE OF USED BATTERIES ACCORDING TO THE INSTRUCTIONS.



ATTENTION! IL Y A RISQUE D'EXPLOSION SI LA BATTERIE EST REMPLACÉE PAR UNE BATTERIE DE TYPE INCORRECT. METTRE AU REBUT LES BATTERIES USAGÉES CONFORMÉMENT AUX INSTRUCTIONS



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 parts!
To avoid burns when handling device parts, wait at least one half hour after switching off the device before handling parts.

PRUDENCE! Pièces chaudes!
Doigts brûlés lors de la manipulation des pièces. Attendez une demi-heure après la mise hors tension avant de manipuler les pièces

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

DigiIX10 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 300 328 v1.8.1 ■ EN 301 893 v1.7.2 ■ EN 301 489 ■ FCC Part 15 Subpart B Class B
Safety compliance standards	EN 62368
E-UTRA CA, E-UTRA FDD, E-UTRA TDD, UMTS FDD	PTCRB
Cellular carriers	See the current list of carriers on the IX10 datasheet, available on the Digi IX10 Specifications page.

Certification category	Standards
Electrical safety compliance	The IX10 model 50002009-01 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.

Command line interface

This chapter contains the following topics:

Access the command line interface 674
Log in to the command line interface 674
Exit the command line interface 675
Execute a command from the web interface 675
Display help for commands and parameters 676
Auto-complete commands and parameters 678
Available commands 679
Use the scp command 680
Display status and statistics using the show command 681
Device configuration using the command line interface 682
Execute configuration commands at the root Admin CLI prompt 683
Configuration mode 685
Command line reference 697

Access the command line interface

You can access the IX10 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: Configure the 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 IX10 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**192.168.210.1** on the .
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 IX10 command line.

You will now be connected to the Admin CLI:

```
Connecting now, 'exit' to disconnect from 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 IX10 WebUI as a user with Admin access.
2. At the main menu, click **Terminal**. The device console appears.

   ```
   IX10 login:
   ```

3. Log into the IX10 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 IX10 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.
 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.

>
```

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.
 serial        Show serial statistics.
 system        Show system statistics.
 version       Show firmware version.


 > 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.
reboot	Reboots the IX10 device.
rm	Removes a file.
scp	Uses the secure copy protocol (SCP) to transfer files between the IX10 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.
system	Issues commands related to system functionality.
traceroute	Sends and tracks route packets to a destination host.
update	Updates the device firmware.

Note For commands that operate on the IX10'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.

Use the scp command

The **scp** command uses Secure Copy Protocol (SCP) to transfer files between the IX10 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 IX10 device from a remote host, or to the remote host from the IX10 device.
  - If the file is being copied to the IX10 device from a remote host:
    - The path and filename of the file on the remote host that will be copied to the IX10 device.
    - The location on the IX10 device where the file will be copied.
  - If the file is being copied to a remote host from the IX10 device:
    - The path and filename of the file on the IX10 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 IX10 device

To copy a file from a remote host to the IX10 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 IX10 device.
- *local-path* is the location on the IX10 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 IX10 device, issue the following command:

```
> scp host 192.168.4.1 user admin remote /home/admin/bin/IX10-21.5.56.106.bin
local /etc/config/scripts to local
admin@192.168.4.1's password: adminpwd
IX10-21.5.56.106.bin                    100%    36MB    11.1MB/s        00:03
>
```

Transfer a file from the IX10 device to a remote host

To copy a file from the IX10 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 IX10 device.

For example:

To copy a support report from the IX10 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-0040D0133536-21-06-15-
   8:04:23.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:D0:13:35:36-21-06-15-8:04:23.bin to remote
   admin@192.168.4.1's password: adminpwd
   support-report-0040D0133536-21-06-15-8:04:23.bin
   >
   ```

Display status and statistics using the show command

The IX10 **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 IX10
  Serial Number            : IX10-000065
  SKU                      : IX10
  Hostname                 : IX10
  MAC Address              : DF:DD:E2:AE:21:18

  Hardware Version         : 50001947-01 1P
  Firmware Version         : 21.5.56.106
  Alt. Firmware Version    : 21.5.56.106
  Alt. Firmware Build Date : Tue, 15 June 2021 8:04:23
  Bootloader Version       : 19.7.23.0-15f936e0ed

  Current Time             : Tue, 15 June 2021 8:04:23 +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 IX10 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.

Configuration actions	Description
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 IX10 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 IX10"
```

Example: Create a new user by using the command line

In this example, you will use the IX10 command line to create a new user, provide a password for the user, and assign the user to authentication groups.

1. Log into the IX10 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.

Command line reference

analyzer 698
cp 699
help 700
ls 701
mkdir 702
modem 703
modem puk status [imei STRING] [name STRING] 708
modem scan [imeiSTRING] [nameSTRING] 709
more 711
mv 712
ping 713
reboot 715
rm 716
scp 717
show 718
ssh 725
system 727
traceroute 731

analyzer

Analyzer commands.

analyzer clear name STRING

Clears the traffic captured by the analyzer.

Parameters

name

Name of the capture filter to use.

Syntax: STRING

analyzer save filename STRING name STRING

Saves the current captured traffic to a file.

Parameters

filename

The filename to save captured traffic to. The file will be saved to the device's /etc/config/analyzer directory.

Syntax: STRING

name

Name of the capture filter to use.

Syntax: STRING

analyzer start name STRING

Start a capture session of packets on this devices interfaces.

Parameters

name

Name of the capture filter to use.

Syntax: STRING

analyzer stop name STRING

Stops the traffic capture session.

Parameters

name

Name of the capture filter to use.

Syntax: STRING

cp

cp commands.

[force] SOURCE DESTINATION

Copy a file or directory.

Parameters

source

The source file or directory to copy.
Syntax: STRING

destination

The destination path to copy the source file or directory to.
Syntax: STRING

force

Do not ask to overwrite the destination file if it exists.
Syntax: BOOLEAN
Default: False
Optional: True

help

Show CLI editing and navigation commands.

Parameters

None

ls

Directory listing command.

ls [show-hidden] PATH

List a directory.

Parameters

path

List files and directories under this path.

Syntax: STRING

show-hidden

Show hidden files and directories. Hidden filenames begin with '.'.

Syntax: BOOLEAN

Default: False

Optional: True

mkdir

mkdir PATH

Create a directory. Parent directories are created as needed.

Parameters

path

The directory path to create.

Syntax: STRING

modem

Modem commands.

modem at [imei STRING] [name STRING] CMD

Send an AT command to the modem and display the response.

Parameters

cmd

The AT command string.

Syntax: STRING

imei

The IMEI of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

name

The configured name of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

modem at-interactive [imei STRING] [name STRING]

Start an AT command session on the modem's AT serial port.

Parameters

imei

The IMEI of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

name

The configured name of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

modem firmware

Commands for interacting with cellular modem firmware. See Update cellular module firmware for further information about using the modem firmware commands.

firmware check [imei *STRING*] [name *STRING*]

Inspect /opt/[MODEM_MODEL]/Custom_Firmware/ directory for new modem firmware file.

Parameters

imei
The IMEI of the modem to execute this CLI command on
Optional: True
Type: string

name
The configured name of the modem to execute this CLI command on
Optional: True
Ref: /network/modem
Type: string

firmware list [imei *STRING*] [name *STRING*]
List modem firmware files found in the /opt/[MODEM_MODEL]/ directory.

Parameters

imei
The IMEI of the modem to execute this CLI command on
Optional: True
Type: string

name
The configured name of the modem to execute this CLI command on
Optional: True
Ref: /network/modem
Type: string

firmware ota
Commands for performing FOTA (firmware-over-the-air) interactions with cellular modem.

ota check [imei STRING] [name STRING]
Query the Digi firmware server for the latest remote modem firmware version.

Parameters

imei
The IMEI of the modem to execute this CLI command on
Optional: True
Type: string

name
The configured name of the modem to execute this CLI command on
Optional: True
Ref: /network/modem
Type: string

ota list [imei STRING] [name STRING]

Query the Digi firmware server for a list of modem firmware versions.

Parameters

imei

The IMEI of the modem to execute this CLI command on

Optional: True

Type: string

name

The configured name of the modem to execute this CLI command on

Optional: True

Ref: /network/modem

Type: string

ota update [imei STRING] [name STRING] [version STRING]

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.

Parameters

imei

The IMEI of the modem to execute this CLI command on

Optional: True

Type: string

name

The configured name of the modem to execute this CLI command on

Optional: True

Ref: /network/modem

Type: string

version

Firmware version name

Optional: True

Type: string

firmware update [imei *STRING*] [name *STRING*] [version *STRING*]

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.

Parameters

imei

The IMEI of the modem to execute this CLI command on

Optional: True
Type: string

name
The configured name of the modem to execute this CLI command on
Optional: True
Ref: /network/modem
Type: string

version
Firmware version name
Optional: True
Type: string

modem pin

PIN commands.

pin change [imei *STRING*] [name *STRING*] OLD-PIN NEW-PIN
Change the SIM's PIN code. Warning: Attempting to use an incorrect PIN code may PUK lock the SIM.

Parameters

old-pin
The SIM's PIN code.
Syntax: STRING

new-pin
The PIN code to change to.
Syntax: STRING

imei
The IMEI of the modem to execute this CLI command on.
Syntax: STRING
Optional: True

name
The configured name of the modem to execute this CLI command on.
Syntax: STRING
Optional: True

pin disable [imei *STRING*] [name *STRING*] PIN
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.

Parameters

pin
The SIM's PIN code.

Syntax: STRING

imei

The IMEI of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

name

The configured name of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

pin enable [imei *STRING*] [name *STRING*] PIN

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.

Parameters

pin

The SIM's PIN code.

Syntax: STRING

imei

The IMEI of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

name

The configured name of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

pin status [imei *STRING*] [name *STRING*]

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

Parameters

imei

The IMEI of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

name

The configured name of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

pin unlock [imei *STRING*] [name *STRING*] PIN

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.

Parameters

pin

The SIM's PIN code.

Syntax: STRING

imei

The IMEI of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

name

The configured name of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

modem puk

PUK commands.

puk status [imei *STRING*] [name *STRING*]

Print the PUK status and the number of PUK unlock attempts remaining.

Parameters

modem puk status [*imei* STRING] [*name* STRING]

Print the PUK status and the number of PUK unlock attempts remaining.

imei

The IMEI of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

name

The configured name of the modem to execute this CLI command on.

Syntax: STRING

Optional: True

puk unlock [imei *STRING*] [name *STRING*] PUK NEW-PIN

Unlock the SIM with a PUK code from the SIM provider.

Parameters

puk
The SIM's PUK code.
Syntax: STRING

new-pin
The PIN code to change to.
Syntax: STRING

imei
The IMEI of the modem to execute this CLI command on.
Syntax: STRING
Optional: True

name
The configured name of the modem to execute this CLI command on.
Syntax: STRING
Optional: True

modem reset [imei STRING] [name STRING]

Reset the modem hardware (reboot it). This can be useful if the modem has stopped responding to the network or is behaving inconsistently.

Parameters

imei
The IMEI of the modem to execute this CLI command on.
Syntax: STRING
Optional: True

name
The configured name of the modem to execute this CLI command on.
Syntax: STRING
Optional: True

modem scan [*imei*STRING] [*name*STRING]

imei
The IMEI of the modem to execute this CLI command on.
Syntax: STRING
Optional: True

name
The configured name of the modem to execute this CLI command on.
Syntax: STRING

Optional: True

modem sim-slot [imei STRING] [name STRING] SLOT

Show or change the modem's active SIM slot. This applies only to modems with multiple SIM slots.

Parameters

slot
The SIM slot to change to.
Syntax: (1|2|show)

imei
The IMEI of the modem to execute this CLI command on.
Syntax: STRING
Optional: True

name
The configured name of the modem to execute this CLI command on.
Syntax: STRING
Optional: True

more

path
The file to view.
Syntax: STRING

mv

Move a file or directory.

mv [force] SOURCE DESTINATION

Parameters

source

The source file or directory to move.

Syntax: STRING

destination

The destination path to move the source file or directory to.

Syntax: STRING

force

Do not ask to overwrite the destination file if it exists.

Syntax: BOOLEAN

Default: False

Optional: True

ping

Ping a host using ICMP echo.

ping [broadcast|ipv6] [count INTEGER] [interface STRING] [size INTEGER] [source STRING] HOST

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.

Syntax: STRING

broadcast

Enable broadcast ping functionality

Syntax: BOOLEAN

Default: False

Optional: True

count

The number of ICMP ping requests to send before terminating.

Syntax: INT

Minimum: 1

Default: 100

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.

Syntax: STRING

Optional: True

ipv6

If a hostname is defined as the value of the 'host' parameter, use the hosts IPV6 address.

Syntax: BOOLEAN

Default: False

Optional: True

size

The number of bytes sent in the ICMP ping request.

Syntax: INT

Minimum: 0

Default: 56

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.

Syntax: STRING

Optional: True

reboot

Reboot the system.

Parameters

None

rm

Remove a file or directory.

rm [force] PATH

Parameters

path
The path to remove.
Syntax: STRING

force
Force the file to be removed without asking.
Syntax: BOOLEAN
Default: False
Optional: True

scp

Copy a file or directory over SSH.

scp host STRING local STRING [port INTEGER] remote STRING to STRING user STRING

Parameters

host
The name or address of the remote host.
Syntax: STRING

local
The file to copy to or from on the local device.
Syntax: STRING

port
The SSH port to use to connect to the remote host.
Syntax: INT
Maximum: 65535
Minimum: 1
Default: 22

remote
The file to copy to or from on the remote host.
Syntax: STRING

to
Copy the file from the local device to the remote host, or from the remote host to the local device.
Syntax: (remote|local)

user
The username to use when connecting to the remote host.
Syntax: STRING

show

Show instance status and statistics.

show analyzer name STRING

Show packets from a specified analyzer capture.

Parameters

name
Name of the capture filter to use.
Syntax: STRING

show arp [ipv4|ipv6|verbose]

Show ARP tables, if no IP version is specififed IPv4 IPV6 will be displayed.

Parameters

ipv4
Display IPv4 routes. If no IP version is specififed IPv4 and IPV6 will be displayed
Syntax: BOOLEAN
Default: False
Optional: True

ipv6
Display IPv6 routes. If no IP version is specififed IPv4 and IPV6 will be displayed
Syntax: BOOLEAN
Default: False
Optional: True

verbose
Display more information (less concise, more detail).
Syntax: BOOLEAN
Default: False
Optional: True

show cloud

Show Digi Remote Manager status and statistics.

Parameters
None

show config

Show changes made to default configuration.

Parameters

None

show dhcp-lease [all|verbose]

Show DHCP leases.

Parameters

all

Show all leases (active and inactive (not in etc/config/dhcp.*lease)).
Syntax: BOOLEAN
Default: False
Optional: True

verbose

Display more information (less concise, more detail).
Syntax: BOOLEAN
Default: False
Optional: True

show dns

Show DNS servers and associated domains.

show event [number INTEGER] [table STRING]

Show event list (high level).

Parameters

number

Number of lines to retrieve from log.
Syntax: INT
Minimum: 1
Default: 20

table

Type of event log to be displayed (status, error, info).
Syntax: (status|error|info)
Optional: True

show hotspot [ip STRING] [name STRING]

Show hotspot statistics.

Parameters

ip

IP address of a specific client, to limit the status display to only this client.

Syntax: STRING
Optional: True

name
The configured instance name of the hotspot.
Syntax: STRING
Optional: True

show ipsec [all] [tunnel STRING]

Show IPsec status statistics.

Parameters

all
Display all tunnels including disabled tunnels.
Syntax: BOOLEAN
Default: False
Optional: True

tunnel
Display more details and config data for a specific IPsec tunnel.
Syntax: STRING
Optional: True

verbose
Display status of one or all tunnels in plain text.
Syntax: BOOLEAN
Default: False
Optional: True

show location [geofence]

Show location information.

Parameters

geofence
Shows the status of any configured geofences.

show log [filter STRING] [number INTEGER]

Show system log (low level).

Parameters

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'.

Syntax: (critical|warning|debug|info)
Optional: True

number
Number of lines to retrieve from log.
Syntax: INT
Minimum: 1
Default: 20

show manufacture [verbose]

Show manufacturer information.

Parameters

verbose
Display more information (less concise, more detail).
Syntax: BOOLEAN
Default: False
Optional: True

show modbus-gateway [verbose]

Show Modbus gateway status and statistics.

verbose
Display more information.
Syntax: BOOLEAN
Default: False
Optional: True

show modem [verbose] [imei STRING] [name STRING]

Show modem status and statistics.

Parameters

imei
The IMEI of the modem to execute this CLI command on.
Syntax: STRING
Optional: True

name
The configured name of the modem to execute this CLI command on.
Syntax: STRING
Optional: True

verbose
Display more information (less concise, more detail).

Syntax: BOOLEAN
Default: False
Optional: True

show nemo [name STRING]

Show NEMO status and statistics.

Parameters

name
The name of a specific NEMO instance.

show network [all|verbose] [interface STRING]

Show network interface status and statistics.

Parameters

all
Display all interfaces including disabled interfaces.
Syntax: BOOLEAN
Default: False
Optional: True

interface
Display more details and config data for a specific network interface.
Syntax: STRING
Optional: True

verbose
Display more information (less concise, more detail).
Syntax: BOOLEAN
Default: False
Optional: True

show ntp

Show NTP status and statistics.

show openvpn

Show OpenVPN status and statistics.

openvpn client [*all*] [name *STRING*]
Show OpenVPN client status statistics.

Parameters

all
Display all clients including disabled clients.

Syntax: BOOLEAN
Default: False
Optional: True

name
Display more details and config data for a specific OpenVPN client.
Syntax: STRING
Optional: True

openvpn server [*all*] [name *STRING*]
Show OpenVPN server status and statistics.

Parameters

all
Display all servers including disabled servers.
Syntax: BOOLEAN
Default: False
Optional: True

name
Display more details and config data for a specific OpenVPN server.
Syntax: STRING
Optional: True

show route [ipv4|ipv6|verbose]

Show IP routing information.

Parameters

ipv4
Display IPv4 routes.
Syntax: BOOLEAN
Default: False
Optional: True

ipv6
Display IPv6 routes.
Syntax: BOOLEAN
Default: False
Optional: True

verbose
Display more information (less concise, more detail).
Syntax: BOOLEAN
Default: False

Optional: True

show scripts

Show scheduled system scripts

Parameters

None

show serial PORT

Show serial status and statistics.

Parameters

port

Display more details and config data for a specific serial port.
Syntax: STRING
Optional: True

show system [verbose]

Show system status and statistics.

Parameters

verbose

Display more information (disk usage, etc)
Syntax: BOOLEAN
Default: False
Optional: True

show usb

Show USB information.

Parameters

None

show version [verbose]

Show firmware version.

Parameters

verbose

Display more information (build date)
Syntax: BOOLEAN
Default: False
Optional: True

show vrrp [all|verbose] [name STRING]

Show VRRP status and statistics.

Parameters

all

Display all VRRP instances including disabled instances.

Syntax: {True|False}

Type: boolean

name

Display more details and configuration data for a specific VRRP instance.

Optional: True

Type: string

verbose

Display all VRRP status and statistics including disabled instances.

Syntax: {True|False}

Type: boolean

show web-filter

Show web filter status and statistics.

Parameters

None

ssh

Use SSH protocol to log into a remote server.

ssh [command STRING] host STRING [port INTEGER] user STRING

Parameters

command

The command that will be automatically executed once the SSH session to the remote host is established.

Optional: True

Type: string

host

The hostname or IP address of the remote host

Syntax: {*hostname*|*IPv4_address*|*IPv6_address*}

Type: string

port

The SSH port to use to connect to the remote host.

Default: 22
Maximum: 65535
Minimum: 1
Syntax: {*Integer*}
Type: integer

user
The username to use when connecting to the remote host.
Type: string

system

System commands.

system backup [passphrase STRING] type STRING PATH

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.

Parameters

passphrase

Encrypt the archive with a passphrase.

Syntax: STRING

Optional: True

Depends on: **type** equals 'archive'

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.

Syntax: (cli-config|archive)

Default: archive

path

The file path to save the backup to.

Syntax: STRING

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.

Parameters

None

system duplicate-firmware

Duplicate the running firmware to the alternate partition so that the device will always boot the same firmware version.

Parameters

None

system factory-erase

Erase the device to restore to factory defaults. All configuration and automatically generated keys will be erased.

Parameters

None

system firmware

System firmware commands.

system firmware update file STRING

Update the current firmware image. Upon reboot the new firmware will be run.

Parameters

file

Firmware filename and path.

Type: string

system firmware ota

Commands for performing FOTA (firmware-over-the-air) interactions.

system firmware ota check

Query the Digi firmware server for the latest device firmware version.

system firmware ota list

Query the Digi firmware server for a list of device firmware versions.

system firmware ota update [version STRING]

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.

Parameters

version

Firmware version name.

Syntax: STRING

Optional: True

system restore [passphrase STRING] PATH

Restore the device's configuration from a backup archive or CLI commands file.

Parameters

path

The path to the backup file.

Syntax: STRING

passphrase

Decrypt the archive with a passphrase.

Syntax: STRING
Optional: True

system script stop SCRIPT

Stop an active running script. Scripts scheduled to run again will still run again (disable a script to prevent it from running again).

Parameters

script
Script to stop.
Syntax: STRING

system serial clear PORT

Clears the serial log.

Parameters

port
Serial port.
Type: string

system serial save PORT FILENAME

Saves the current serial log to a file.

Parameters

port
Serial port.
Type: string

filename
The filename to save the serial log. The file will be saved to the device's /etc/config/serial directory.
Type: string

system serial show PORT

Displays the serial log on the screen.

Parameters

port
Serial port.
Type: string

system serial start [size INTEGER] PORT

Start logging data on a serial port.

Parameters

size
Maximum size of serial log.
Default: 65536
Syntax: {*Integer*}
Type: integer

port
Serial port.
Type: string

system serial stop PORT

Start logging data on a serial port.

Parameters

port
Serial port.
Type: string

system support-report PATH

Save a support report to a file and include with support requests.

Parameters

path
The file path to save the support report to.
Syntax: STRING

traceroute

Print the route packets trace to network host.

traceroute [bypass|debug|dontfragment|icmp|ipv6|nomap] [first_ttl INTEGER] [gateway STRING] [interface STRING] [max_ttl INTEGER] [nqueries INTEGER] [packetlen INTEGER] [pausemsecs INTEGER] [port INTEGER] [src_addr STRING] [tos INTEGER] [waittime INTEGER] HOST

Parameters

bypass

Bypass the normal routing tables and send directly to a host on an attached network.

Syntax: BOOLEAN

Default: False

Optional: True

debug

Enable socket level debugging.

Syntax: BOOLEAN

Default: False

Optional: True

dontfragment

Do not fragment probe packets.

Syntax: BOOLEAN

Default: False

Optional: True

first_ttl

Specifies with what TTL to start.

Syntax: INT

Minimum: 1

Default: 1

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

Syntax: STRING

Optional: True

icmp

Use ICMP ECHO for probes.

Syntax: BOOLEAN

Default: False

Optional: True

interface

Specifies the interface through which traceroute should send packets. By default, the interface is selected according to the routing table.

Syntax: STRING

Optional: True

ipv6

If a hostname is defined as the value of the 'host' parameter, use the hosts IPV6 address.

Syntax: BOOLEAN

Default: False

Optional: True

max_ttl

Specifies the maximum number of hops (max time-to-live value) traceroute will probe.

Syntax: INT

Minimum: 1

Default: 30

nomap

Do not try to map IP addresses to host names when displaying them.

Syntax: BOOLEAN

Default: False

Optional: True

nqueries

Sets the number of probe packets per hop. A value of -1 indicated

Syntax: INT

Minimum: 1

Default: 3

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.

Syntax: INT

Minimum: -1

Default: -1

pausemsecs

Minimal time interval between probes

Syntax: INT

Minimum: 0

Default: 0

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.

Syntax: INT

Minimum: -1

Default: -1

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.

Syntax: STRING

Optional: True

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.

Syntax: INT

Minimum: -1

Default: -1

waittime

Determines how long to wait for a response to a probe.

Syntax: INT

Minimum: 1

Default: 5

host

The host that we wish to trace the route packets for.

Syntax: STRING