Workspace ONE Notifications Service Guide

[V4, October, 2024]

- Overview
- Authentication
  - Source of Authentication Makes a Difference
  - Creating a service client
  - Getting an access token
- Creating a new notification using Notification API
  - Create a notification for a user
  - Create a notification for multiple users (Synchronous)
  - Create a notification for multiple users (Asynchronous)
  - Create a notification for a user on a specific device
- Creating a new notification using Admin Console Notification Builder
  - Notification Setting
    - Enable Notification
    - Display Push Notification Description
    - New App Notifications
    - Default Notification Image
  - Choose an audience
  - Choose a priority level
  - Choose a template
  - Action Button Configuration
  - Add an attachment
  - Persistent Notification
- Guidelines for creating user friendly notifications
  - No-action announcement or alert
  - Single action notification
  - Multiple action notification
  - Notifications with a custom icon
  - Priority notifications
  - Notification with links
  - Notification with Banner Image
  - Notification with Banner Video
  - Notification with Attachments
- View a notification

- Update a notification
- Delete a notification
- Additional Information
    - Getting the user id of a specific user
    - Getting the device id of a specific user
- Matching users

Overview

The Workspace ONE Notifications Service is a robust, flexible cloud-hosted service designed to generate and serve actionable, real-time notifications in compliance with the Hero card specifications. It enables a wide variety of powerful use cases to provide an engaging user experience within the Workspace ONE Intelligent Hub. Whether it's alerting employees about new apps, open benefits enrollment, or general organization-wide announcements, you have complete control over the design, layout and content of every message, who will receive it and when.

After 20.04 release, end-user will see "For You" tab replacing the Notification tab with a brandnew modernized page layout across all platforms. Once a new notification is created, it will be rendered under the "For You" tab. You can create a notification using either the Notification API or in the Hub Services admin console Notification Builder.

Depending on which priority or notification type that you choose, the notification you created will be grouped under the following three sections:

- Priority
- Actionable
- Informational

Authentication

Source of Authentication Makes a Difference

When the source of authentication for Intelligent Hub is set to Workspace ONE UEM, you do not need to configured Workspace ONE Access to create custom notifications.

- Users receive notifications on their iOS, Android, and macOS devices.
- Users receive New App notifications for apps used on their devices.

When the source of authentication is set to Workspace ONE Access, you have additional notification features.

- Users receive notifications on their iOS, Android, and macOS devices and in the Hub portal from a browse.
- Users receive new app notifications about web and virtual apps.

To see which service is the source for authentication in the Intelligent Hub app, in the UEM console go to the Devices > Device Settings > Devices & Users > General > Enrollment > Authentication tab.

Creating a service client

You need to create a service client on Workspace ONE Access for your app to get the access token that you need to create notifications. This is a one-time setup. In the Workspace ONE Access admin UI, go to Settings Click on the OAuth 2.0 Management menu on the left-hand side and click Add Client * Select Service Client Token as the Access Type * Enter a Client ID * Click Add

Check the 'Admin' scope, and ensure it is assigned the 'Super Admin' role.

The **client secret** is generated and is displayed on the UI.

Getting an access token

You will need to acquire a new access token whenever the current token expires. By default, an access token is valid for 6 hours but that can be configured to a different value too. To acquire an access token, we use the following API:

POST /SAAS/auth/oauthtoken
Host: acme.uss.workspacevmwareidentity.com (Note* this is just an example, customers should invoke the
APIs for the host which matches customer's tenant url)
Body: grant_type=client_credentials
Authorization: Basic <token>
Content-Type: application/x-www-form-urlencoded

Here <token> is obtained by **base64** encoding of client_id:client_secret from the previous step.

For example: if, client id = test-notification and client secret = QdW9baluUl9zUiadOjXuzu8F8fyE1Qs Then the **base 64** encoding of testnotification:dQdW9baluUl9zUiadOjXuzu8F8fyE1Qs gives dGVzdC1ub3RpZmljYXRpb246ZFFkVzliYWx1VWw5elVpYWRPalh1enU4RjhmeUUxUXPCoA==

More information can be found here:
(https://docs.omnissa.com/bundle/workspace-one-access-administration-guide/page/AddOAuth20ServiceClientTokensinWorkspaceONEAccess.html)

Creating a new notification using Notification API

The notification service API can be used to create new notifications for a particular user on a specific device and/or tenant.

To get the user ID of a particular user, see here.

Create a notification for a user

To create a new notification for a user with user ID = {**targetUserId**}

POST: /ws1notifications/api/v1/users/{targetUserId}/notifications
Host: acme.vmwareidentity.com (Note* this is just an example, customers should invoke the APIs for the host which matches customer's tenant url)
Content-Type: application/json
Authorization: Bearer <Access-Token> **Body**

See Guidelines for creating user friendly notifications

Response

HTTP/1.1 201 Created
Content-Type: application/json


```
{
   "created_at": "2018-04-02T08:03:07.71Z",
   "updated_at": "2018-04-02T08:03:07.71Z",
   "id": "2cad8515-c6db-446a-b543-390bac2e67fc",
   "tenant_id": "greenbox-ui",
   "user_id": "7cd85d9a-c742-43ef-90dc-2d48d3b85539",
   "creator_id": "70d95845-4e5f-4567-b515-e0333a5b3ce2",
   "device_id": null,
   "read_at": null,
   "last_action_id": null,
   "notification_card": {
      "header": {
         "title": "This is a title"
      },
      "body": {
         "description": "This is a description"
      }
   }
}
```

Create a notification for multiple users (Synchronous)

This is an API endpoint to create a new notification for multiple users. This endpoint synchronously creates notifications for up to 100 users. To create notifications for more than 100 users, we use the asynchronous endpoint.

For example, to create a notification for user1 with userID = "abc123" and user2 with userID = "xyz456", we use

POST: /ws1notifications/api/v1/distributed_notifications
Host: acme.vmwareidentity.com (Note* this is just an example, customers should invoke the APIs for the host which matches customer's tenant url) Content-Type: application/json
Authorization: Bearer <Access-Token>

Body

```
{
  "notification_card": {
   "header":{
      "title":"This is a title"
   },
   "body":{
      "description":"This is a description"
   }
},
  "user_ids": [
    "abc123",
    "xyz456"
  ]
}
```

Response

HTTP/1.1 200 OK
Content-Type: application/json

```
{
    "abc123": {
        "status_code": "200",
        "notification_id": "023a5977-cf8e-4bdd-b120-ded901ec34a8"
    },
    "xyz456": {
        "status_code": "200",
        "notification_id": "105cfeac-9e4f-4219-b6ba-a72ede3e52d9"     } }
```

Create a notification for multiple users (Asynchronous)

This is an API endpoint to create a new notification for multiple users. This endpoint asynchronously creates notifications for a very large number of users.

For example, to create a notification for user1 with userID = "abc123" and user2 with userID = "xyz456", we use

POST: /ws1notifications/api/v1/distributed_notifications_async
Host: acme.vmwareidentity.com (Note* this is just an example, customers should invoke the APIs for the host which matches customer's tenant url) Content-Type:
application/json
Authorization: Bearer <Access-Token>

Body

```
{
  "notification_card": {
   "header":{
      "title":"This is a title"
   },
   "body":{
      "description":"This is a description"
   }
},
  "user_ids": [
    "abc123",
    "xyz456"
  ]
}
```

Response

HTTP/1.1 200 OK

Create a notification for a user on a specific device

To create a new notification for a user with user ID = {**targetUserId**} and device ID = {**targetDeviceId**}

To get a specific device ID of a particular user, see here.

POST
/ws1notifications/api/v1/users/{targetUserId}/devices/{targetDeviceId}/notifi cations
Host: acme.vmwareidentity.com (Note* this is just an example, customers should invoke the APIs for the host which matches customer's tenant url)
Content-Type: application/json

Authorization: Bearer <Access-Token>
Body

See Guidelines for creating user friendly notifications

Response

HTTP/1.1 201 Created
Content-Type: application/json

Creating a New Notification using Admin Console Notification Builder

Intelligent Hub customers can also use Notification Builder which is a code free solution to create and send push notifications. The custom Notification Builder guides you through the steps to create a notification, select the target audience, and create an actionable response. When you click **Create** in the wizard, the notification is sent to the target audience. Each notification excluding attachment and icon cannot exceed 399 KB. Notification record will be cleared every 90 days.

Notification Setting

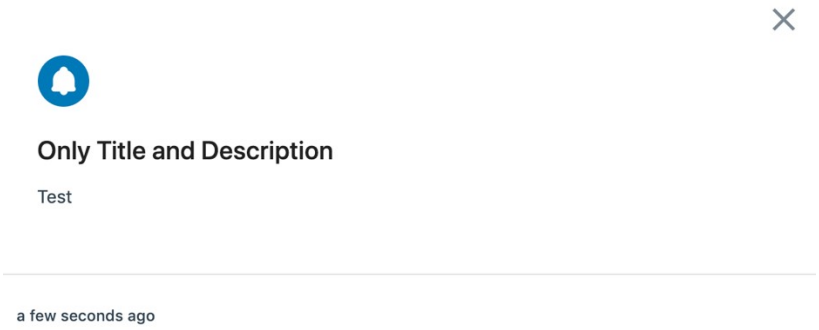
Once you are in Intelligent Hub Admin Console, go to Notifications -> Global Settings

Enable Notifications
Once this toggle is turned on, end users will start seeing the "For You" tab in Intelligent Hub app.

Display Push Notification Description
Once this toggle is turned on, the Intelligent Hub app will display push notification with a description about the notification on user's devices. Only the push notification title is displayed when this is disabled.

New App Notifications
Once this toggle is turned on, the Intelligent Hub app will start generating a weekly digest of new apps sent to employees. The first digest will be sent out within one week after enabling.

Default Notification Image
You can add a default notification icon to enhance your branding experience for your employees. Once you upload an image, this image will be used as the default for all custom notifications. Upload a JPG, PNG, GIF, or SVG file that is 40px wide by 40px high for best results. File size cannot exceed 30KB. You can change the icon for individual notification when you create them. If default notification image is not uploaded, Intelligent Hub will use a bell icon for each notification. See example below:



Once you have enabled Notifications, you can click CREATE CUSTOM NOTIFICATION to launch Notification Builder and start creating a notification.

Choose an audience

You can target custom notifications to specific audiences, or you can send the notification to all employees in your organization. You can assign the following audience types.

Audience Type	Description
All Employees	Notifications can be sent to all employees listed as users in the Workspace ONE Access service.
All Devices	Notifications can be sent to all devices configured in your Workspace ONE UEM environment, regardless of organization groups.
Organisation Group	Organization groups (OG) in the Workspace ONE UEM console are created to group individual organizations in your corporate structure, geographical location, business unit, or department. You can select a specific organization group in Workspace ONE UEM in which to send the notification.
Smart Group	Smart groups are customizable groups within Workspace ONE UEM that determine which platforms, devices, and users receive an assigned application, book, compliance policy, device profile, or provision. You can select a smart group in Workspace ONE UEM in which to send the notification.
Platform	You can select a specific platform type to send a notification. The platforms are iOS, Android, macOS, and Windows.

Audience Type	Description

Users Group	Groups in the Workspace ONE Access service are imported from your Active Directory or are created as local groups in the Workspace ONE Access console. You can select a group in Workspace ONE Access in which to send a notification.

Choose a priority level

There are three levels of priority.

Standard

Informational notifications such as approvals, maintenance notices and organizational updates that do not require a timely response or no response at all. If you choose Informational template, then this notification will be grouped under "For You" tab Informational section. If you choose Actionable template, then this notification will be grouped under "For You" tab Actionable section.

High Priority

Time sensitive notifications such as password expiration updates, required security training and legal notices that require attention and a timely response. Once you choose High Priority, regardless which template you choose, this notification will be grouped under "For You" tab Priority section.

Urgent

Extremely urgent notifications that proactively alert users about important updates such as severe weather alerts, emergencies and compromised devices that require immediate attention and/or response. Once you choose Urgent priority level, this notification will serve as a screen blocker. It will require users to act on the notification before they can continue using Intelligent Hub.

Choose a template

There are two templates you can choose.

Actionable

Actionable notifications that require employees to take action and appear separately from informational cards.

Informational

Informational notifications that do not require employees to take any action and appear separately from actionable cards.

Action Button Configuration

At the most, three actions can be presented on the notification card. When you create an action, a button appears with the text you configured. One or two actions is the ideal user experience. Any more than three can degrade the user experience as the buttons are in a horizontal line on the card, and the card can be too narrow to display the action buttons correctly.

Three action types can be selected.

- **Open In**. When you select the Open In action, you then enter the URL that opens when the button is clicked.
- **API**. When you select API, you then select the method to use to interact with the data being addressed in the notification.

API	Task Description
GET	Use GET to retrieve data from a specified resource. The data is not modified. Enter the URL of the resource that supplies the data.
POST	Use POST to send data to a specified URL to insert data in a resource. Enter the URL where the data is sent.
PUT	Use PUT to send data to a specified URL to update a resource. Use PUT when user can only send the data one time.
PATCH	Use PATCH to allow users to modify their data on the resource.
DELETE	Use Delete to let the user delete data from the specified URL.

- **API with Parameters**. When you select API with Parameters, in addition to selecting the method and API endpoint, you can add specific parameters and a value for the parameters.

For more information about parameters, go to the Workspace ONE Notification service API page. The Notifications Service reference guide is available from the Documentation tab.

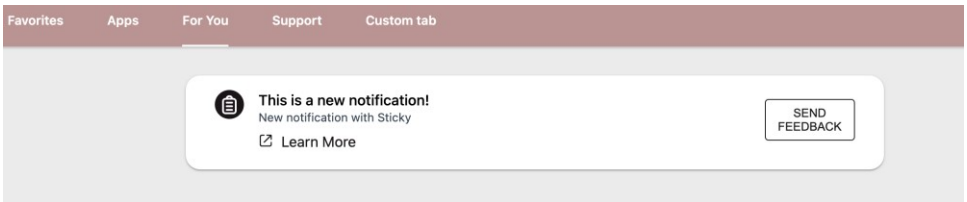
Add an attachment

You can add up to 10 attachments to notification that you create. The individual file size cannot exceed 1 MB.

Persistent Notification

A persistent notification is a sticky notification card that will persist on top of the For You page in the Intelligent Hub for a defined duration of time. Once the duration of time defined by you expires, this sticky notification card will be moved to the History page automatically. To generate a persistent notification, you need to set the expiration time within the "sticky" object when generating a notification using the Notifications API. See the json example below for how to set the sticky object and include expiration date and time in the notification.

```
{
        "header": {
           "title": "sasaaa dadasdasdassd This is new notification! dasda",
           "subtitle": ["learn more"],
           "links": {
                           "subtitle": ["https://google.com"]
           }
        },
        "body": {
           "description": "New notification with Sticky "
        },
        "sticky": {
                "until": "2020-02-18T11:27:52.667Z"
        },
        "actions": [
        {
           "id": "0e49552a-500e-48b5-b48f-d0c7d9681b19",
           "label": "Send Feedback",
           "completed_label": "Feedback Submitted",
           "allow_repeated": false,
           "type": "GET",
           "url": {
              "href": "https://google.com"
           },
           "action_key": "OPEN_IN"
        }
      ]
  }
```



Guidelines for creating user friendly notifications

The notification card (the body of the create notification request) should be in accordance with the Hero card specifications aas well as some conventions that are specific to Intelligent Hub as described below.

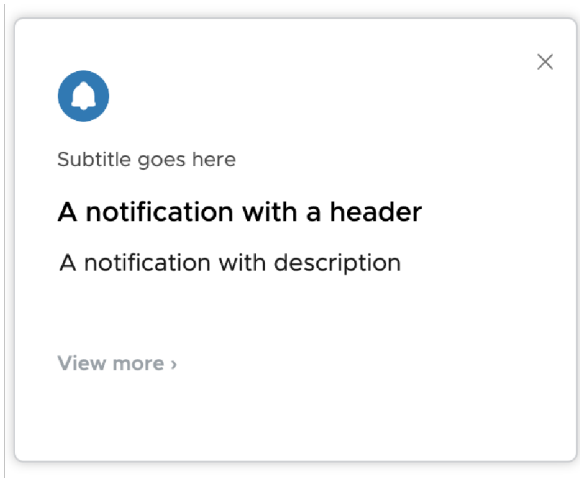
Here are some guidelines for creating effective notifications that enable a great user experience on Intelligent Hub!

Both title and description of the following attributes are **required** on the cards:

1. **header.title** 2. **header.subtitle** 3. **body.description**

This is a valid (albeit not very useful) notification.

```
{
  "header":{
    "title":"A notification with a header"
  },
"body":{
    "description":"A notification with description."
},
}
```



1. Write short titles that catch the user's attention. The description message should be clear and concise.
2. At most three actions should be present on the notification card. One or two actions is the ideal user experience. Any more than three and the user experience degrades as the individual buttons will not be wide enough to accomodate even medium length action labels.
3. Keep action labels short to avoid truncation.
4. If we want the user to be able to repeat the action, we set "allow_repeated" : true else "allow_repeated" : false if we want the user to be able to perform that action only once.
5. The action buttons are rendered as either primary or secondary actions. We can set "primary": true if we want the action to be rendered as a primary action or "primary": false if we want it to be a secondary action as shown:

Shown below are notifications illustrating a few common use cases:

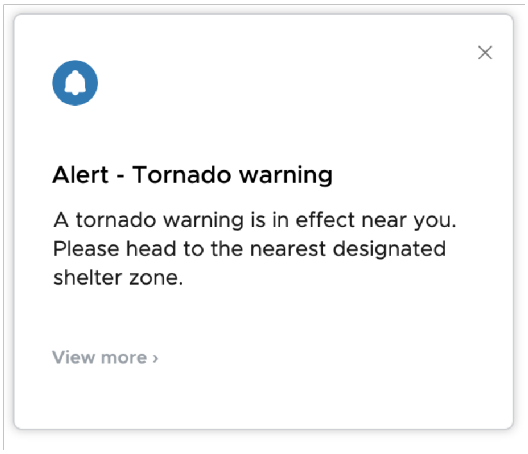
No-action announcement or alert

To create an unactionable announcement, notice or alert, we structure the notification body as follows:

```
{
  "header":{
    "title":"Alert - Tornado warning"
  },
```

```
  "body":{
    "description":"A tornado warning is in effect near you. Please head to the nearest designated shelter zone."
    }
}
```

which looks like

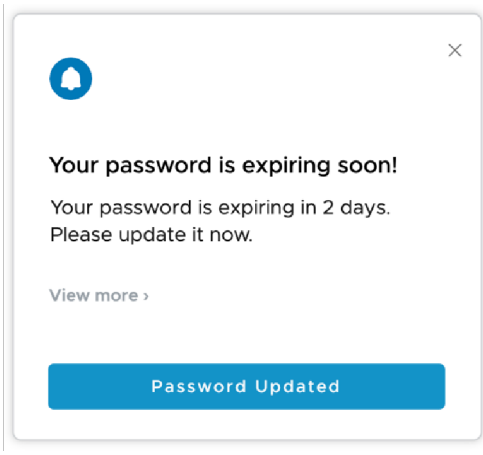


Single action notification

To create a notification with a single action such as a prompt for the user to change their password, we structure the body as follows:

```
{
  "header":{
    "title":"Your password is expiring soon!"
  },
  "body":{
    "description":"Your password is expiring in 2 days. Please update it now."    },
"actions":[{
      "id":"a8406d97-2f46-45a8-9e3d-f68dac0cdf18",
      "label":"Update password",
      "completed_label": "Password updated",
      "type":"POST",
      "primary": true,
      "allow_repeated": false,
      "url":{
        "href":"https://password-change-url.vmwareidentity.com"
      },
      "action_key":"OPEN_IN"
    }]
}
```

which looks like



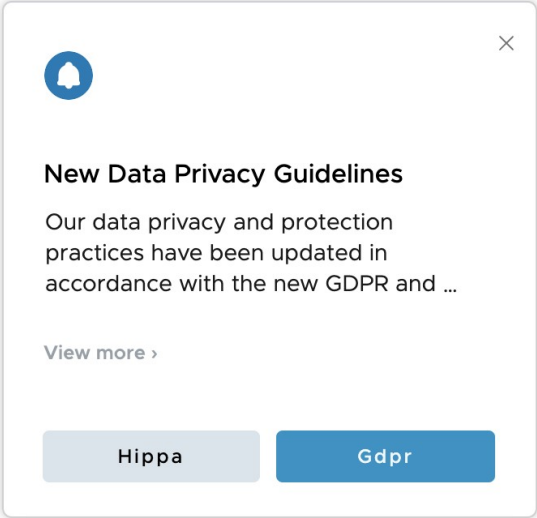
Multiple action notification

To create a notification with multiple actions, we structure the body as follows:


```
{
  "header":{
    "title":"New Data Privacy Guidelines"
  },
  "body":{
    "description":"Our data privacy and protection practices have been updated in accordance with the new GDPR and HIPPA guidelines. To read more, see below."
  },
  "actions":[{
      "id":"a8406d97-2f46-45a8-9e3d-f68dac0cdf18",
      "label":"GDPR",
      "type":"POST",
      "primary": true,
      "allow_repeated": false,
      "completed_label": "Done",
      "url":{
        "href":"https://gdpr.com"
      },
      "action_key":"OPEN_IN"
    },{
      "id":"b8406d97-2f46-45a8-9e3d-f68dac0cdf18",
      "label":"HIPPA ",
      "type":"POST",
      "primary": false,
      "allow_repeated": false,
      "completed_label": "Done",
      "url":{
        "href":"https://hippa.com"
```

```
        },
        "action_key":"OPEN_IN"
      }]
}
```

which looks like

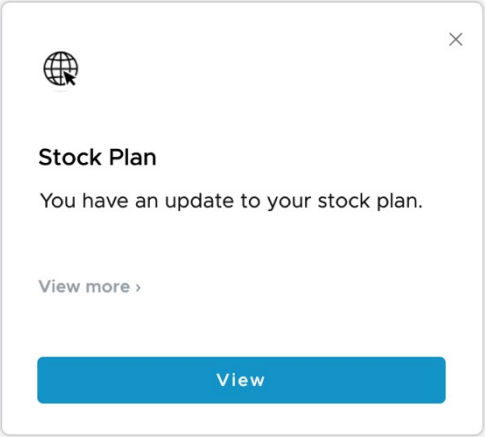


Notification with a custom icon

To create a notification with a custom image as the icon, we add an 'image' property to the card.

```
{
   "header":{
      "title":"Stock Plan"
   },
   "body":{
      "description":"You have an update to your stock plan."     },

   "image":{
      "href":" https://image.shutterstock.com/image-vector/click-go-website-internet-line-600w-516477817.jpg "
   },
   "actions":[{
        "id":"a8406d97-2f46-45a8-9e3d-f68dac0cdf11",
        "label":"View",
        "type":"POST",
        "primary": true,
        "allow_repeated": true,
        "completed_label": "View",
        "url":{
```

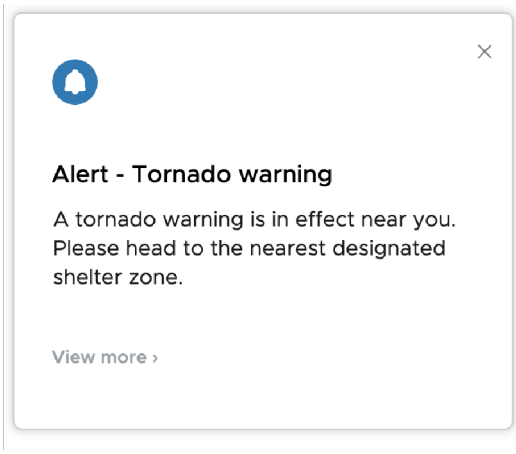
```
        "href":"https://stock-plan.acme.com"
      },
      "action_key":"OPEN_IN"
    }]
}
```



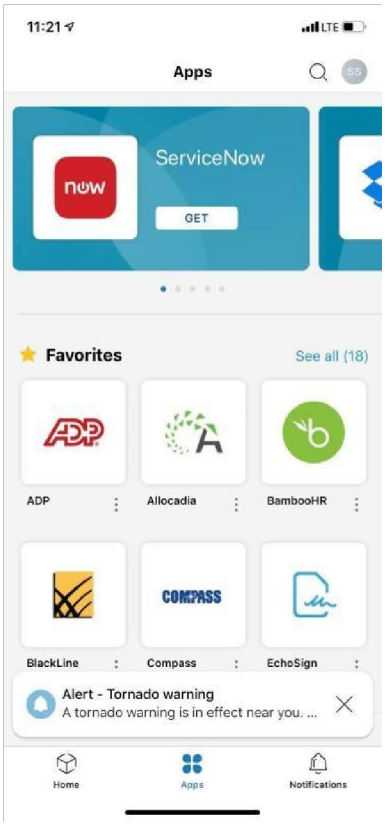
Priority notifications

To mark a notification as high priority, we add an 'importance' property in the body of the card and give it a value of 1 as follows:

```
{
  "header":{
    "title":"Alert - Tornado warning"    },

     "importance" : 1,

  "body":{
    "description":"A tornado warning is in effect near you. Please head to the nearest designated
shelter zone."

    }
}
```



By marking a card as priority, the user sees it in the priority section of the notifications feed with a highlighted border and the notification count does not reduce until the priority card is marked as read or acted on. In addition, if the user is on a different view of Intelligent Hub and a priority notification is received, they are alerted in the form of a message near the bottom of the screen which brings the user's attention to the incoming priority notification.

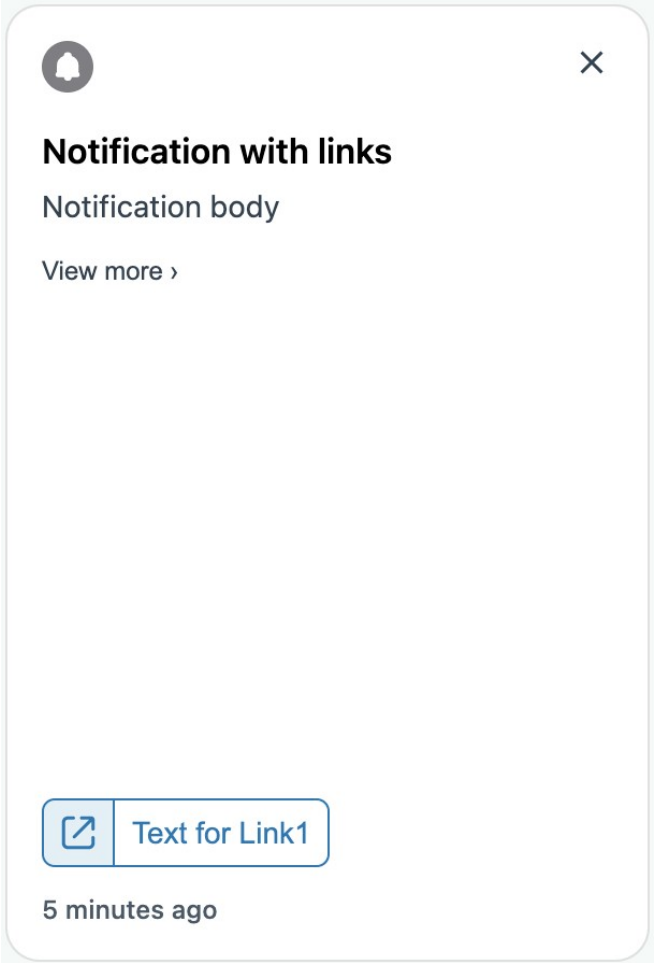


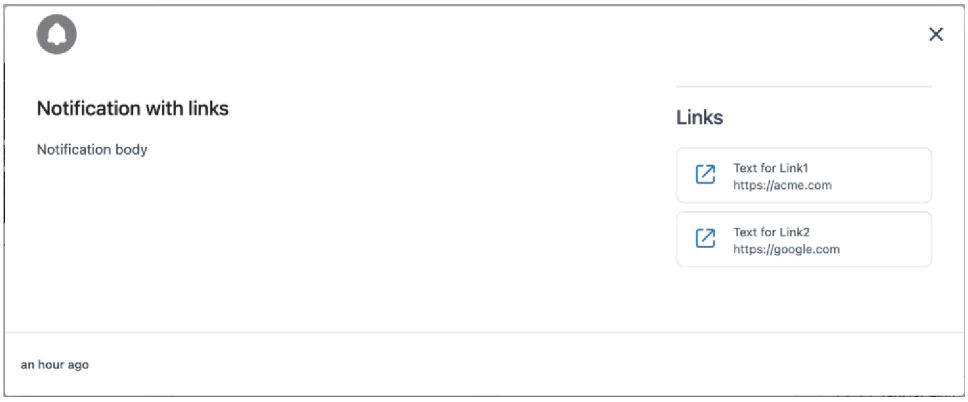
Notifications with Links

To create a notification with a custom links in the notification body, we add a 'links' property to the card with the URL and the text as follows:

{
"header": {
"title": "Notification with links"},
"body": {
"description": "Notification body"},
"links": [
{
"href": "https://acme.com",
"text": "Text for Link1"
},
{
"href": "https://google.com",
"text": "Text for Link2"
}
]
} which looks

like





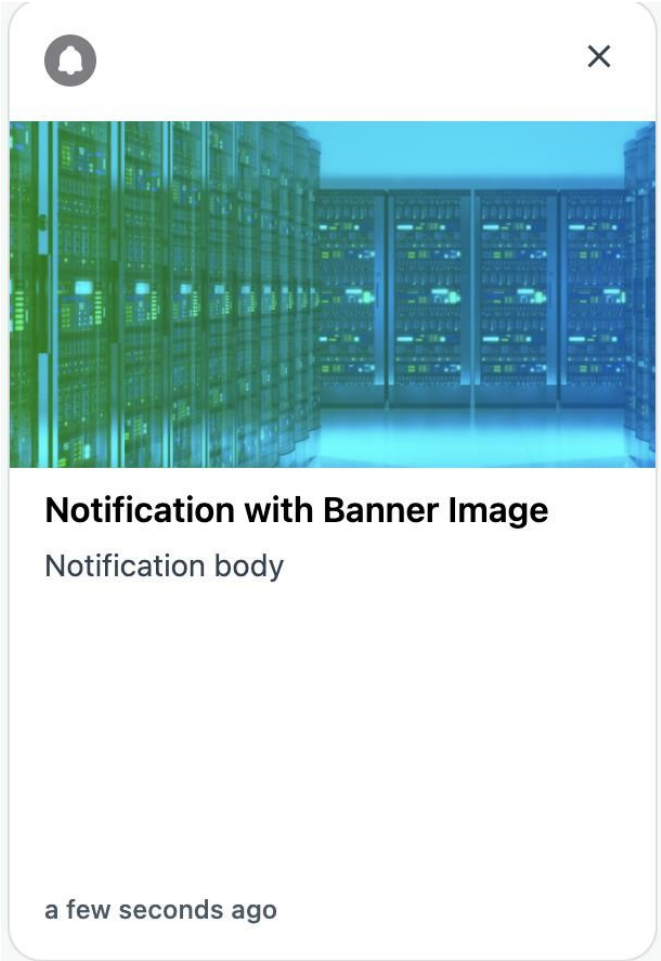
You can add up to 10 link objects to a notification. Short card of the notification shows only the first link added. To see the other links, check the long card of the notification.

Notifications with Banner Image

To create a notification with a banner image in the notification body, we add `banner' property which can contain banner items for the card with the type as `image' and other details about the image as follows

```
{
   "header": {
     "title": "Notification with Banner image"},
   "body": {"description": "Notification body"},
   "banner": {
      "items": [{
           "type": "image",
           "href": "https://mti.com/wp-content/uploads/2017/08/Omnissa-banner.jpg",
           "title": "Omnissa logo",
           "description": "description about the banner image"}
      ]
    }
}
```

which looks like

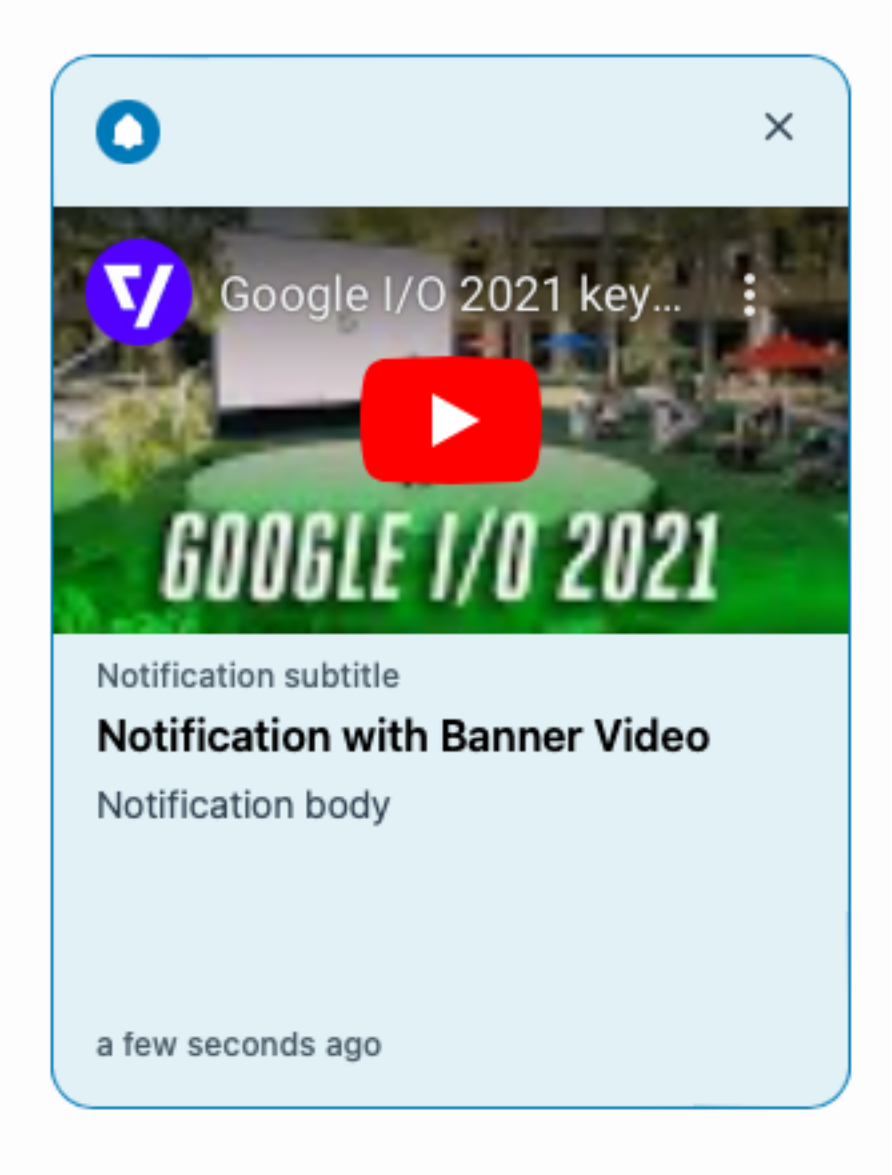


Notifications with Banner Videos

To create a notification with a banner video in the notification body, we add 'banner' property which can contain banner items for the card with the type as 'video' and other details about the video as follows:

```
{
  "header": {
    "title": "Notification with Banner Video"},
  "body": {"description": "Notification body"},
  "banner": {
    "items": [{
        "type": "video",
        "href": " https://www.youtube.com/embed/_xLgXIhebxA ",
        "title": "UI Community Team",
        "description": "description about the banner video"}
```

```
    ]
  }
} which looks
like
```



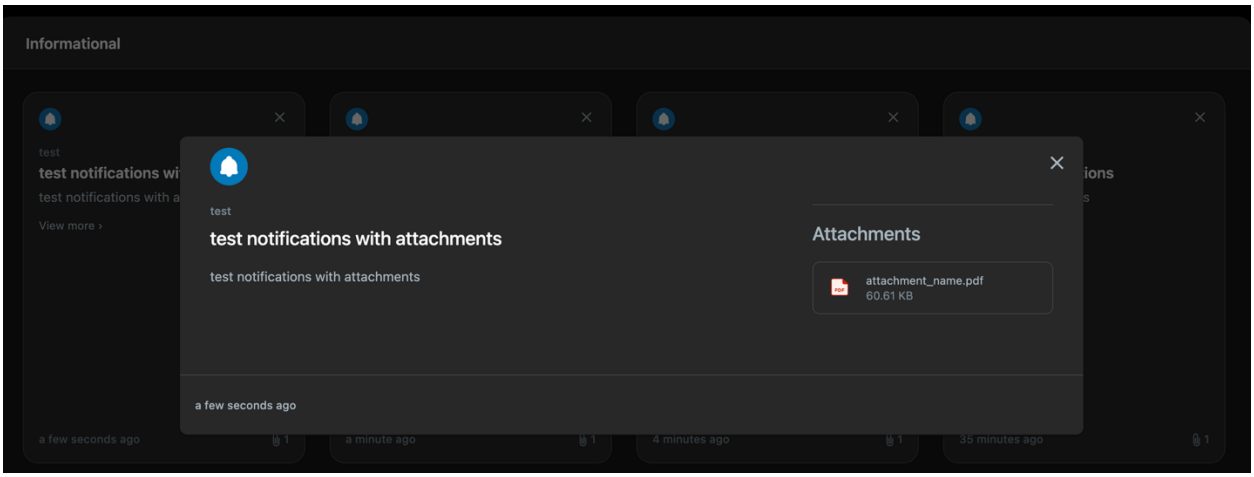
Notifications with Attachments

To create a notification with attachments in the notification body, we structure the body as follows:

```
{
  "header": {
```

```
    "title": "test notifications with attachments",
    "subtitle": [
      "test"
    ]
  },
  "body": {
    "description": "test notifications with attachments",
    "fields": [
      {
        "type": "SECTION",
        "title": "Attachments",
        "items": [
          {
            "type": "ATTACHMENT_URL",
            "title": "Click here to download the attachment",
            "attachment_name": "attachment_name.pdf",
            "attachment_content_length": 62063,
            "attachment_content_type": "application/pdf",
            "attachment_url": "https://host.com:443/assets/id/attachment.pdf",
            "attachment_method": "GET"
          }
        ]
      }
    ]
  }
}
```

which looks like



View a notification

To view the details of a notification with ID = {**notification-id**}

GET /ws1notifications/api/v1/notifications/{notification-id}
Host: acme.vmwareidentity.com
Authorization: Bearer <Access-Token>

Response

HTTP/1.1 200 OK
Content-Type: application/json


```
{
  "created_at": "2018-07-13T07:25:26.854Z",
  "updated_at": "2018-07-13T07:25:26.854Z",
  "id": "bd6d0c3d-5152-489b-b0b0-11168133f634",
  "tenant_id": "acme",
  "user_id": "51e01723-9a10-416e-82f3-e151cabc3770",
  "creator_id": "69561a9e-6968-4e1d-8611-6622d0a4ae92",
  "device_id": null,
  "read_at": null,
  "last_action_id": null,
  "notification_card": {
    "header": {
      "title": "This is a title"
    },
    "body": {
      "description": "This is a description"
    }
  }
```

}

Update a notification

The Notifications API supports updating a notification with new content.

To update a notification with ID = {**notification-id**}

POST /ws1notifications/api/v1/notifications/{notification-id}
Host: acme.vmwareidentity.com
Content-Type: application/json
Authorization: Bearer <Access-Token>
Body:

```
{
  "header":{
    "title":"This is the updated title"
  },
  "body":{
    "description":"This is the updated description"    }
}
```

Response:

HTTP/1.1 201 Created
Content-Type: application/json
Below is the response to update a notification with notification ID = 2cad8515-c6db446ab543-390bac2e67fc


```
{
   "created_at": "2018-04-02T08:03:07.71Z",
   "updated_at": "2018-04-02T08:11:08.165Z",
   "id": "2cad8515-c6db-446a-b543-390bac2e67fc",
   "tenant_id": " acme ",
   "user_id": "7cd85d9a-c742-43ef-90dc-2d48d3b85539",
   "creator_id": "70d95845-4e5f-4567-b515-e0333a5b3ce2",
   "device_id": null,
   "read_at": null,
   "last_action_id": null,
   "notification_card": {
      "header": {
         "title": "This is the updated title"
      },
      "body": {
         "description": "This is the updated description"
      }
```

```
  }
}
```


Delete a notification

The Notifications API supports deleting a notification. To delete a notification with ID = {**notification-id**}

DELETE /ws1notifications/api/v1/notifications/{notification-id}
Host: acme.vmwareidentity.com
Authorization: Bearer <Access-Token> **Response:**

HTTP/1.1 204 No Content

Additional Information

Getting the user id of a specific user

To get the user ID of a user with username = {**userName**}

GET
/SAAS/jersey/manager/api/scim/Users?filter=userName%20eq%20%22{userName}%22 Host:
acme.vmwareidentity.com
Authorization: Bearer <Access-Token> Accept:
application/json

Response

HTTP/1.1 200 OK
Content-Type: application/json


```
{
  "totalResults": 1,
  "itemsPerPage": 1,
  "startIndex": 1,
  "schemas": [
    "urn:scim:schemas:core:1.0",
    "urn:scim:schemas:extension:workspace:1.0",
    "urn:scim:schemas:extension:enterprise:1.0",
    "urn:scim:schemas:extension:workspace:mfa:1.0",
    "urn:scim:schemas:extension:workspace:tenant:greenbox-ui:1.0"  ],
  "Resources": [
    {
      "active": true,
      "userName": "user-name",
      "id": "51e01723-9a10-416e-82f3-e151as342bc3770",
      "meta": {
```

"created": "2018-07-10T18:17:55.335Z", "lastModified": "2018-07-10T18:19:21.995Z", "location": "https://acme.vmwareidentity.com/SAAS/jersey/manager/api/scim/Users/51e01723-9a10-416e-82f3-e151as342bc3770",
"version": "W/\"1531246761995\""
},
"name": {
"givenName": "First Name",
"familyName": "Last Name"
},
"emails": [
{
"value": "useremail@domain.com"
}
],
"groups": [
{
"value": "337f36b2-7550-49a5-a1d0-121da939d94b",
"type": "direct",
"display": "ALL USERS"
}
],
"roles": [
{
"value": "04f574c0-06ac-4063-85a7-2ad0635b5c6d",
"display": "User"
}
],
"urn:scim:schemas:extension:workspace:1.0": {
"internalUserType": "LOCAL",
"userStatus": "1",
"domain": "System Domain",
"userStoreUuid": "487faa2c-e9d1-4939-bc70-26f79faf7f29"
}
}
]
}

Getting the device id of a specific user

This Workspace ONE UEM API can be used to get the device IDs of a user with username = **userName**

The auth token is the base64 encoding of admin-username:admin-password

For example if the Workspace ONE UEM admin username for the tenant is 'admin' and the admin
password is 'password', then the auth Token is the base64 encoding of admin:password which is YWRtaW46cGFzc3dvcmQ=

We also need an admin API key of the Workspace ONE UEM tenant which can be found in the Workspace ONE UEM console under Settings > System > Advanced > API

GET /api/mdm/devices/search?user=userName
Host: <uem-tenant-url>
Authorization: Basic <Token> Accept: application/json aw-tenant-code: <tenantadmin-api-key>

Similary using this API endpoint we can also get all the device IDs of a certain device **model** (iPhone, Galaxy) or **platform** (iOS, Android)

To search by model or platfrom, we use '?model=iPhone' or '?platform=Android' instead of the '?user=userName' query string parameter at the end of the url.

Response

HTTP/1.1 200 OK
Content-Type: application/json


```
{
  "Devices": [
    {
      "EasIds": {
        "EasId": [
          "Text value"
        ]
      },
      "Udid": "027BE1C5AEC05C378C61C44103E9D3FCB2EC354D",
      "SerialNumber": "R51G844T90R",
      "MacAddress": "485A3F880798",
      "Imei": "356766060039613",
      "EasId": "6Q93UFOQ7H0K39JPMFPTEMJQ3K",
      "AssetNumber": "827BE1C5AEC05C378C61C44103E9D3FCB2EC354D",
      "DeviceFriendlyName": "users iPhone iOS 10.3.2 ",
      "LocationGroupId": {
        "Id": {
          "Value": 1
        },
        "Uuid": "5741a4a4-ece3-4f18-b0b0-28a3b2017b2e",
        "Name": "Text value"
```

},
"LocationGroupName": "locationgroup1",
"UserId": {},
"UserName": "userName",
"UserEmailAddress": "userName@omnissa.com",
"Ownership": "C",
"PlatformId": {},
"Platform": "Apple",
"ModelId": {},
"Model": "iPhone",
"OperatingSystem": "10.3.2",
"PhoneNumber": "+14045550100",
"LastSeen": "2018-07-16T11:41:11.3254194-04:00",
"EnrollmentStatus": "Enrolled",
"ComplianceStatus": "Compliant",
"CompromisedStatus": true,
"LastEnrolledOn": "2018-07-16T11:41:11.3254194-04:00",
"LastComplianceCheckOn": "2018-07-16T11:41:11.3254194-04:00",
"LastCompromisedCheckOn": "2018-07-16T11:41:11.3254194-04:00",
"ComplianceSummary": {
"DeviceCompliance": [
{
"CompliantStatus": true,
"PolicyName": "application list compliance policy",
"PolicyDetail": "compliance policy for device compromised status including application list contains rule",
"LastComplianceCheck": "2018-07-16T11:41:11.3364487-04:00",
"NextComplianceCheck": "2018-07-16T11:41:11.3364487-04:00",
"ActionTaken": [
{
"ActionType": 0
}
],
"Id": {
"Value": 0
},
"Uuid": "74f8db15-ecca-4782-afaf-3b590907b7a7"
}
]
},
"IsSupervised": true,
"DeviceMCC": {
"SIMMCC": "404",

```
      "CurrentMCC": "310"
    },
    "IsRemoteManagementEnabled": "abcd",
    "DataEncryptionYN": "Y",
    "AcLineStatus": 1,
    "VirtualMemory": 2,
    "OEMInfo": "abcd",
    "DeviceCapacity": 1,
    "AvailableDeviceCapacity": 1,
    "LastSystemSampleTime": "2018-07-16T11:41:11.3434692-04:00",
    "IsDeviceDNDEnabled": true,
    "IsDeviceLocatorEnabled": true,
    "IsCloudBackupEnabled": true,
    "IsActivationLockEnabled": true,
    "IsNetworkTethered": true,
    "BatteryLevel": "abcd",
    "IsRoaming": true,
    "LastNetworkLANSampleTime": "2018-07-16T11:41:11.344471-04:00",
"LastBluetoothSampleTime": "2018-07-16T11:41:11.344471-04:00",
    "SystemIntegrityProtectionEnabled": true,
    "ProcessorArchitecture": 5,
    "UserApprovedEnrollment": true,
    "EnrolledViaDEP": true,
    "TotalPhysicalMemory": 3,
    "AvailablePhysicalMemory": 4,
    "Id": {
      "Value": 0
    },
    "Uuid": "fa1451be-9b6e-4e15-85fe-b2e884cf556a"
  }
 ],
 "Page": 1,
 "PageSize": 2,
 "Total": 3
}
```


Matching users

Customer can use SCIM API (https://developer.omnissa.com/ws1-access-apis/api-reference) and filter based on username, email address or any other parameter to match user within other systems. Workspace ONE solution mandates userName attribute and it would be advisable for customers to filter based on username. In addition to that, email address can be used as long as email address is also a synced attribute via Workspace ONE Access.