



# Platform API Guide

---

Version: 2020.3.0

# Copyright AppViewX, Inc.

**Copyright © 2020 AppViewX, Inc. All Rights Reserved.**

This document may not be copied, disclosed, transferred, or modified without the prior written consent of AppViewX, Inc. While all content is believed to be correct at the time of publication, it is provided as general-purpose information. The content is subject to change without notice and is provided “as is” and with no expressed or implied warranties whatsoever, including, but not limited to, a warranty for accuracy made by AppViewX. The software described in this document is provided under written license only, contains valuable trade secrets and proprietary information, and is protected by the copyright laws of the United States and other countries. Unauthorized use of software or its documentation can result in civil damages and criminal prosecution.

## **Trademarks**

The trademarks, logos, and service marks displayed in this manual are the property of AppViewX or other third parties. Users are not permitted to use these marks without the prior written consent of AppViewX or such third party which may own the mark.

## **External Reference Links**

This product includes software developed by the CentOS Project ([www.centos.org](http://www.centos.org)).

This product includes software developed by Red Hat, Inc. ([www.redhat.com](http://www.redhat.com)).

This product includes software developed by VMware, Inc. ([www.vmware.com](http://www.vmware.com)).

All other trademarks mentioned in this document are the property of their respective owners.

## **Contact Information**

AppViewX, Inc.

222 Broadway, FL 19

New York, NY 10038

Email: [info@appviewx.com](mailto:info@appviewx.com)

Web: [www.appviewx.com](http://www.appviewx.com)

# Contents

Preface.....	5
Revision History.....	5
About the Documentation.....	5
Audience.....	5
Text Conventions.....	5
<b>Chapter 1. Overview.....</b>	<b>6</b>
Restful HTTPS Requests.....	6
Description of Server Responses.....	7
URI Scheme.....	7
Types of Accounts in AppViewX.....	7
<b>Chapter 2. Accessing API with User Account.....</b>	<b>8</b>
Retrieve session ID using login API.....	8
Using Session ID for further API calls.....	10
<b>Chapter 3. Accessing API with Service Account.....</b>	<b>12</b>
Retrieve Access Token using get-service-token API.....	12
Using Access Token in the header for further API calls.....	14
<b>Chapter 4. Commons.....</b>	<b>16</b>
Add a Resource .....	16
Add Service Account .....	18
Search for a Resource .....	20
Get Resources for User Group.....	22
Delete a Resource .....	24
Search for a Role .....	26
Add a Role .....	28
Get Role Information .....	30
Delete a Role.....	32
Clone a Role .....	33

Share Dashboard to User Group .....	35
Add a User Group.....	37
Associate Permissions to Role .....	38
Search User Group.....	40
Delete a User Group .....	42
Associate User Group with Resources .....	44
Add a User .....	45
Associate Roles to a User Group .....	47
Associate User Groups to a User .....	49
Search for a user.....	51
Assign User group to Service Account.....	54
Update Resource Description.....	55
Before you begin.....	56
Request Structure.....	56
Response Structure.....	57
Status Codes.....	58
Sample Request/Response.....	58
Update User Group Description.....	59
Before you begin.....	59
Request Structure.....	59
Response Structure.....	60
Status Codes.....	61
Sample Request/Response.....	61
<b>Chapter 5. Glossary.....</b>	<b>62</b>

# Preface

## Revision History

Revision	Description	Date
1.0	Initial Release of AppViewX_v2020.3.0 Platform	September 2020

## About the Documentation

The AppViewX Platform is a modular, low-code software application that enables the automation and orchestration of network infrastructure using an intuitive, context-aware, visual workflow. Leveraging a vast library of pre-built tasks and workflows, the platform enables Ops teams to quickly and easily translate business requirements into automation workflows that improve agility, enforce compliance, eliminate errors, and reduce cost.

## Audience

This guide is intended for CISO, PKI Security, and Application Teams.

## Text Conventions

The following text conventions are used in this document:

Convention	Description
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, or terms defined in the text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>codeblock</code>	Indicates commands with a paragraph, URLs, codes in examples, text that appears on the screen, or text that you enter.

# Chapter 1: Overview

The AppViewX Platform is a module that lets you enable general configuration settings such as authentication, authorization, and integration of external services like log forwarding, HSM integration, SMTP configuration, and so on. These general configuration settings are applicable to all AppViewX subsystems such as ADC, CERT+, Security+, Visual Workflow, and so on.

An API enables users to programmatically transfer data in and out of the AppViewX subsystems. With access to RESTful AppViewX APIs, you can leverage the raw potential of AppViewX. It provides a powerful way to channel the data into native business applications. This document comprises of module-wise APIs used in the AppViewX Platform.

- [Restful HTTPS Requests](#)
- [Description of Server Responses](#)
- [URI Scheme](#)
- [Types of Accounts in AppViewX](#)

## Restful HTTPS Requests

### HTTP GET

GET requests, retrieve resource representation/information only and not to modify it.

### HTTP POST

POST APIs create new subordinate resources. For example, a file is subordinate to a directory containing it or a row is subordinate to a database table. In terms of REST, POST methods are used to create a new resource into the collection of resources.

### HTTP PUT

PUT APIs are used to update existing resources (if a resource does not exist then API may decide whether to create a new resource or not).

### HTTP DELETE

DELETE APIs are used to delete resources (identified by the Request-URI).

## Description of Server Responses

1. **200** OK - The request was successful (some API calls may return 201 instead).
2. **400** Bad Request - The request is not understood or required parameters are missing.
3. **401** Unauthorized - Authentication failed or the user doesn't have permissions for the requested operation.
4. **403** Forbidden - Access denied.
5. **404** Not Found - Resource not found.
6. **429** Too many requests - The number of requests to the service has crossed the threshold.
7. **504** Gateway timeout - The given request has exceeded the expected time.
8. **503** Service unavailable - The client cannot communicate with the service.

This guide explains how to use the AppViewX API to perform different functions. It includes descriptions, pointers to code samples, and request/response formats.

## URI Scheme

- **Host** : {url}
- **BasePath** : /avxapi
- **Schemes** : HTTPS
- **URL** : https://{url}/avxapi

## Types of Accounts in AppViewX

There are two types of accounts in AppViewX:

- **User Accounts**: These are used by actual users.
- **Service Accounts**: These are used by system services such as web servers, automation tools, and so on.

AppViewX recommends using a Service Account for accessing APIs from automation tools. Service Accounts are supported with OAuth standard for a more secure and standard way of accessing APIs.



**Note:** AppViewX supports both User Account and Service Account for accessing APIs.

## Chapter 2: Accessing API with User Account

For accessing APIs with a user account, you need to get the session ID by providing a username and password in the login API. This session ID can then be used for accessing other APIs.



**Note:** You can also use the username and password in all API calls instead of the sessionId. However, this is not recommended.

- [Retrieve session ID using login API](#)
- [Using Session ID for further API calls](#)

### Retrieve session ID using login API

**URL:** http://<Hostname>:31443/avxapi/login?gwsouce=external

**Method:** POST

**Response Format:** JSON

**Requires authentication:** Yes

**Request timeout period:** 15 minutes

#### Parameters

Param Type	Name	Description	Field Type
Header	username**	Login username	String
Header	Password**	Password for the username	String
Header	Content-Type**	Specifies the nature of the data in the payload (The value of the param should be "application/json")	String
Query	gwsouce**	Source from which the request is triggered (for example, web, external)	String

\*\* - Mandatory value

#### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	Login failed. Invalid credentials.
200	OK	Login successful
401	Unauthorized	Authentication failed. Invalid Credentials.

### Sample Request/Response

#### Use Case

Log in to the application with a username and password.

#### Sample Request

URL: http://<Hostname>:31443/avxapi/login?gwsources=external

#### Headers

username: admin

password: AppViewX@123

Content-Type: application/json

grant\_type=client\_credentials

#### Request Payload

```
{}
```

#### Sample Response

```
{
  "response":
  {
    "status": "SUCCESS",
    "appStatusCode": null,
    "statusDescription": null,
    "sessionId": "ce7f1a14-2bf9-4e4a-89a8-bc780a255813",
    "availableLoginAttemptCount": 10
  },
  "message": null,
  "appStatusCode": null,
  "tags": null,
}
```

```
"headers": null
}
```

## Using Session ID for further API calls

The sessionId retrieved using the login API can be used in the header for making further API calls.

**Use Case:** Add a role using API with sessionId

**URL:** <http://<Hostname>:31443/avxapi/role?gwsource=external>

**Type:** POST

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId**	Use sessionId retrieved from login API	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installation	String
Query	gwsource**	Source from which the request is triggered (for example, web, external)	String
Payload	name **	Name of the role to be added	String
Payload	description	Description of the role to be added	String

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Mes
400	Bad Request	Role name cannot be left empty or blank.
201	Created	Role added successfully
209	Conflict	Role Name Already Exists

### Sample Request

**URL:** http://<Hostname>:31443/avxapi/role?gwsouce=external

#### Headers

sessionId: ce7f1a14-2bf9-4e4a-89a8-bc780a255813

#### Request Payload

```
{
  "payload":{
    "name":"role_1",
    "description":"Adding a new role"
  }
}
```

#### Sample Response

```
{
  "response": "Role added successfully",
  "message": "Role added successfully",
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Chapter 3: Accessing API with Service Account

For accessing APIs with a service account, you need to get the Access Token by providing Client ID and Client Secret in get-service-token API. This Access Token can then be used for accessing other APIs.



**Note:** Access Token Validity is 30 minutes by default and it can be configured in **Settings > Authentication > OAuth Settings**.

- Retrieve Access Token using [get-service-token API](#)
- Using Access Token in the header for further API calls

### Retrieve Access Token using get-service-token API

**URL:** `http://<Hostname>:31443/avxapi/acctmgmt-get-service-token?gwsource=external`

**Method:** POST

**Response Format:** JSON

**Requires authentication:** Yes

**Request timeout period:** 15 minutes

#### Parameters

Param Type	Name	Description	Field Type
Header	Authorization**	Please form a string in this format <Client ID>:<Client Secret> and do base64 encoding. Then prepend a key 'Basic' before the encoded value. Final value should be "Basic <EncodedValue>"	String
Header	Content-Type**	Specifies the nature of the data in the payload (The value of the param should be "application/json")	String
Payload	grant_type**	Payload Type should be "Form". The value of the param should be Client_Credentials".	Text

\*\* - Mandatory value

#### Possible Response Message and Code



## Using Access Token in the header for further API calls

The access token retrieved using the get-service-token API can be used in the header for making further API calls.

**Use Case:** Add a role using API with Access Token

**URL:** <http://<Hostname>:31443/avxapi/role?gwsource=external>

**Type:** POST

### Parameters

Param Type	Name	Description	Field Type
Header	Token**	Use token retrieved from login API.	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installation.	String
Query	gwsource**	Source from which the request is triggered (for example, web, external).	String
Payload	name **	Name of the role to be added.	String
Payload	description	Description of the role to be added.	String

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad Request	Role name cannot be left empty or blank.
201	Created	Role added successfully
209	Conflict	Role Name Already Exists

### Sample Request

**URL:** <http://<Hostname>:31443/avxapi/role?gwsource=external>

Headers



## Chapter 4: Commons

- Add a Resource
- Add Service Account
- Search for a Resource
- Get Resources for User Group
- Delete a Resource
- Search for a Role
- Add a Role
- Get Role Information
- Delete a Role
- Clone a Role
- Share Dashboard to User Group
- Add a User Group
- Associate Permissions to Role
- Search User Group
- Delete a User Group
- Associate User Group with Resources
- Add a User
- Associate Roles to a User Group
- Associate User Groups to a User
- Search for a user
- Assign User group to Service Account
- Update Resource Description
- Update User Group Description

### Add a Resource

This API is used to create a new resource. A resource is a logical entity to group one or more ACL managed entities.

**URL:** */resource*

**Type:** POST

## Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login	String
Header	Username	Name of the user	String
Header	Password	Password of the user	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations	String
Query	gwsource**	Source from which the request is triggered (for example, web, external)	String
Payload	name**	Name of the resource to create. Name cannot be duplicated.	String
Payload	description	Description of the resource	String

\*\* - Mandatory value

## Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	Mandatory field 'name' is missing
201	Created	Resource added successfully
409	Conflict	Resource with the given name already exists
407	Proxy Authentication Required	The Username and Password field is mandatory.

## Sample Request/Response

### Use Case

Add a new resource with the name resource\_1.

### Request URL

<https://appviewxapi.com/avxapi/resource?gwkey=f000ca01&gwsource=external>

## Request Payload

```
{
  "payload": {
    "name": "resource_1",
    "description": "This is a sample resource."
  }
}
```

## Response

```
{
  "response": "Resource added successfully",
  "message": "Resource added successfully",
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Add Service Account

This API is used to create a new service account.

**URL:** [avxapi/acctmgmt-add-service-account?gwsource=web&gwkey=f000ca](https://avxapi/acctmgmt-add-service-account?gwsource=web&gwkey=f000ca)

**Method** - Post

**Response Format:** JSON

**Requires authentication:** Yes

**Request timeout period:**15 minutes

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login	string
Query	email	Email Address	string
Payload	loginName**	Unique names used as the service account name for authentication. Duplication is not allowed.	string

Param Type	Name	Description	Field Type
Payload	description	Description of the service account	string
Header	Content-Type	Specifies the nature of the data in the payload (The value of the param should be "application/json")	string

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	Mandatory field 'name' is missing.
201	Created	Resource added successfully.
409	Conflict	Resource with the given name already exists.
407	Proxy Authentication Required	The Username and Password field is mandatory.

### Sample Request/Response

#### Use Case

To add a Service account.

#### Request URL

avxapi/acctmgmt-add-service-account?gwsouce=web&gwkey=f000ca

#### Request Payload

```
{
  "payload": {
    "loginName": "internal90",
    "email": "internaldemo188@appviewx.com",
    "description": "description"
  }
}
```

#### Response

```

{
  "response": null,
  "message": null,
  "appStatusCode": null,
  "tags": null,
  "headers": null
}

```

## Search for a Resource

This API is used to search for a resource based on input filter condition. A resource is a logical entity to group one or more ACL managed entities. The response could be one or more resource(s) matching the input.

**URL:** */resource-search*

**Type:** POST

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations	String
Query	Token	access token received after login	String
Query	gwsouce**	Source from which the request is triggered (for example, web, external)	String
Payload	filterValue	Input filter string value to filter the resources.	String
Payload	sortColumn	Column by which the output has to be sorted	String
Payload	startIndex	Start index to show in output in case if there are more than one matching records	Integer
Payload	maxSize	The number of records to return in a response. maxSize is starting from the start index	Integer

Param Type	Name	Description	Field Type
Payload	sortOrder	The order to sort. Possible values are asc and desc	String

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
200	OK	Matching results found for the given input.
404	OK	No matching records found.
400	Bad request	Invalid sort Column

### Sample Request/Response

#### Use Case

Search a resource with the name Role\_admin.

#### Request URL

<https://appviewxapi.com/avxapi/resource-search?gwkey=f000ca01&gwsource=external>

#### Request Payload


```
{
  "payload": {
    "startIndex": 1,
    "maxSize": 3,
    "sortOrder": "asc",
    "sortColumn": "name",
    "filterValue": " resource_1"
  }
}

Response
{
  "response": {
    "data": [
      {
```

```

"name": "Role_admin",
"description": "Super access will have the permissions to access all the resources in AppViewX.",
"state": "A",
"_id": " Role_admin ",
"_keywords": null
}
],
"totalRecords": 1,
"obtainedRecords": 1,
"obtainedRecordRange": {
"start": 1,
"end": 1
}
},
"message": "Matching results found for the given input.",
"appStatusCode": null,
"tags": null,
"headers": null
}

```

 **Note:** The state of a resource indicates if the resource is active or not. 'A' stands for active.

## Get Resources for User Group

This API is used to fetch all the resources for a given user group.

**URL:** */usergroup-getResources*

**Type:** POST

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String

Param Type	Name	Description	Field Type
Query	gwsource**	Source from which the request is triggered (for example, web, external).	String
Payload	Name **	Name of the user group.	String
Payload	startIndex	Start index to show in output in case if there are more than one matching records.	Integer
Payload	maxSize	The number of records to return in a response. maxSize is starting from the start index	Integer

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
200	OK	Matching results found for the given input.
404	Not Found	No matching records found.
400	Bad request	Usergroup name cannot be empty.

### Sample Request/Response

#### Use Case

Get all resources under the user group usergroup\_1.

#### Request URL

<https://appviewxapi.com/avxapi/usergroup-getResources?gwkey=f000ca01&gwsource=external>

#### Request Payload

```
{
  "payload": {
    "startIndex": 1,
    "maxSize": 0,
    "name": "usergroup_1"
  }
}
```

## Response

```
{
  "response":{
    "data":[
      {"name": " resource_0", "description": "Resource for usergroup_0", "state": "A",...},
      {"name": "resource_1", "description": " Resource for usergroup_1", "state": "A",...},
      {"name": " resource_2", "description": "Resource for usergroup_1", "state": "A", ,...},
      {"name": " resource_3", "description": "Resource for usergroup_1", "state": "A",...}
    ],
    "totalRecords": 4,
    "obtainedRecords": 4,
    "obtainedRecordRange":{
      "start": 1,
      "end": 4
    }
  },
  "message": "Matching results found for the given input.",
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```



**Note:** The state of a resource indicates if the resource is active or not. 'A' stands for active.

## Delete a Resource

This API is used to delete a resource.

**URL:** */resource*

**Type:** DELETE

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String
Query	gwsouce**	Source from which the request is triggered (for example, web, external)	String
Query	resourceName**	Name of the resource to be deleted.	String

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	ResourceName cannot be empty.
200	OK	Resource deleted successfully.
404	Not Found	Resource Name does not exist.
403	Forbidden	Current user(s) aclIdentifier/resource cannot be deleted.

### Sample Request/Response

#### Use Case

Delete a resource with name resource\_1.

#### Request URL

[https://appviewxapi.com/avxapi/resource?  
resourceName=resource\\_1&gwkey=f000ca01;gwsouce=external](https://appviewxapi.com/avxapi/resource?resourceName=resource_1&gwkey=f000ca01;gwsouce=external)

#### Request Payload

NA

#### Response

```
{
  "response": "Resource deleted successfully",
  "message": "Resource deleted successfully",
```

```

"appStatusCode": null,
"tags": null,
"headers": null
}

```

## Search for a Role

This API is used to search for a role based on the input filter condition. A role is a logical entity to group one or more Access Control Functions. The response could be one or more resource(s) matching the input.

**URL:** */role-search*

**Type:** POST

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String
Query	gwsource**	Source from which the request is triggered (for example, web, external).	String
Payload	filterValue	Input filter value string to filter the role.	String
Payload	sortColumn	Column by which the output has to be sorted.	String
Payload	startIndex	Start index to show in output in case if there are more than one matching records.	Integer
Payload	maxSize	The number of records to return in a response. maxSize is starting from the start index.	Integer
Payload	sortOrder	The order to sort. Possible values are asc and desc.	String

\*\* - *Mandatory value*

### Possible Response Message and Code

HTTP Code	Description	Response Message
200	OK	Matching results found for the given input.
404	OK	No matching records found.
400	Bad request	Invalid sortColumn.

### Sample Request/Response

#### Use Case

Search a role with the name Role\_admin.

#### Request URL

<https://appviewxapi.com/avxapi/role-search?gwkey=f000ca01&gwsource=external>

#### Request Payload

```
{
  "payload": {
    "startIndex": 1,
    "maxSize": 2,
    "sortColumn": "name",
    "sortOrder": "asc",
    "filterValue": "admin"
  }
}
```

#### Response

```
{
  "response": {
    "data": [
      {
        "name": "admin",
        "description": "admin",
        "permissions": [
          "certificate:connectorActions:secureConnector",
          "certificate:settings:casettings:custom_ca",
          "certificate:settings:appsettings:view",
          "adc:dashboard:deviceheatmap:modifysettings",

```

```
"certificate:client:adminaccess",
"adc:deviceGroup:delete"
],
"state": "A",
"accessControlObjects": null,
"_id": "admin",
"_keywords": null
}
],
"totalRecords": 1,
"obtainedRecords": 1,
"obtainedRecordRange": {
"start": 1,
"end": 1
}
},
"message": "Matching results found for the given input.",
"appStatusCode": null,
"tags": null,
"headers": null
}
```



**Note:** The state of a role indicates if the resource is active or not. 'A' stands for active.

## Add a Role

This API is used to add a new role to the system.

**URL:** */role*

**Type:** POST

**Parameters**

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations	String
Query	gwsouce**	Source from which the request is triggered (for example, web, external)	String
Payload	name**	Name of the role.	String
Payload	description	Description of the role	String

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	Role name cannot be left empty or blank.
201	Created	Role added successfully
409	Conflict	Role Name Already Exists.

### Sample Request/Response

#### Use Case

Add a role with a valid role name.

#### Request URL

<https://appviewxapi.com/avxapi/role?gwkey=f000ca01&gwsouce=external>

#### Request Payload

```
{
  "payload": {
    "name": "role_1",
    "description": "Adding a new role"
  }
}
```

## Response

```
{
  "response": "Role added successfully",
  "message": "Role added successfully",
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Get Role Information

This API is used to retrieve the role information.

**URL:** /role

**Type:** GET

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login	String
Query	gwkey	Tenant Key. Needed only in case of multitenant installations	String
Query	gwsouce**	Source from which the request is triggered (for example, web, external)	String
Query	roleName **	Name of the role.	String

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
200	OK	Role Information Retrieved Successfully
400	Bad Request	Role name cannot be empty

HTTP Code	Description	Response Message
404	Not Found	Role name is not found

### Sample Request/Response

### Use Case

Get the information about the role **role\_1**.

### Request URL

[https://appviewxapi.com/avxapi/role?roleName=role\\_1&gwkey=f000ca01;gwsouce=external](https://appviewxapi.com/avxapi/role?roleName=role_1&gwkey=f000ca01;gwsouce=external)

### Request Payload

NA

### Response

```
{
  "response": {
    "name": "role_1",
    "description": "role_1 details",
    "permissions": [
      "permission_1"
    ],
    "state": "A",
    "_id": "log",
    "_keywords": [
      "role used to test role functions.",
      "log",
      "Enabled"
    ],
    "message": "Role Information Retrieved Successfully",
    "appStatusCode": null,
    "tags": null
  }
}
```

## Delete a Role

This API is used to delete a role.

**URL:** `/role`

**TYPE:** DELETE

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String
Query	gwsouce**	Source from which the request is triggered (for example, web, external).	String
Query	roleName**	Name of the role to delete.	String

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	RoleName value is missing.
200	OK	Role deleted successfully
404	Not Found	RoleName does not exist
409	Forbidden	Role(s) which have active user(s) cannot be deleted.

### Sample Request/Response

#### Use Case

Delete a role with name role\_1.

#### Request URL

[https://appviewxapi.com/avxapi/role?roleName=role\\_1&gwkey=f000ca01;gwsouce=external](https://appviewxapi.com/avxapi/role?roleName=role_1&gwkey=f000ca01;gwsouce=external)

## Request Payload

NA

## Response

```
{
  "response": "Role deleted successfully",
  "message": "Role deleted successfully",
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Clone a Role

This API is used to clone a role.

**URL:** *role-clone*

**Type:** POST

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String
Query	gwsouce**	Source from which the request is triggered (for example, web, external).	String
Payload	existingName **	Name of the role to be cloned.	String
Payload	newName**	Name of the cloned new role.	String
Payload	description**	Description of the cloned role.	String

\*\* - *Mandatory value*

## Possible Response Message and Code

HTTP Code	Description	Response Message
200	Ok	Clone role successful.
404	Not Found	Role Name to be cloned not found.
400	Bad Request	The payload cannot be empty or null.
400	Bad Request	Source Role Name can not be empty for clone action.
400	Bad Request	Role name cannot be left empty or blank.
400	Bad Request	Role name cannot be less than 2 characters.
400	Bad Request	Role name cannot exceed 64 characters.
400	Bad Request	Invalid Description.
400	Bad Request	Source role and new role names cannot be empty.
409	Conflict	ClonedRole Name already exists for the given RoleName.

## Sample Request/Response

### Use Case

Clone a role role\_1 to create a new cloned role named clonerole\_1.

### Request URL

<https://appviewxapi.com/avxapi/role-clone?gwkey=f000ca01;gwsource=external>

### Request Payload

```
{
  "payload": {
    "existingName": "role_1",
    "newName": "clonerole_1",
    "description": "clone test role."
  }
}
```

### Response

```

{
  "response": "Clone role successful",
  "message": "Clone role successful",
  "appStatusCode": null,
  "tags": null,
  "headers": null
}

```

## Share Dashboard to User Group

This API is used to share an existing dashboard to one or more existing usergroups. The sharing can be restricted to read-only or as a read-write.

**URL:** `/dashboard-share-usergroupdetails-update`

**Type:** PUT

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login	String
Query	Gwkey	Tenant Key. Needed only in case of multitenant installations	String
Query	gwsource**	Source from which the request is triggered (for example, web, external)	String
Payload	readUserGroup	Name of the user group that need to have only read access to the dashboard.	String
Payload	readWriteUserGroup	Name of the user group that need to have read-write access to the dashboard.	String
Query	dashboardName**	Name of the dashboard to be shared	String

\*\* - *Mandatory value*

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	Dashboard name is mandatory.
404	Not Found	Dashboard not found.
403	Forbidden	Read-write permission not available for the given dashboard.

### Sample Request/Response

#### Use Case

Share a dashboard named Dashboard1 in read mode to the usergroup and in read/write mode to the usergroup.

#### Request URL

<https://appviewxapi.com/avxapi/dashboard-share-usergroupdetails-update?name=Dashboard1&gwkey=f000ca01&gwsource=WEB>

#### Request Payload

```
{
  "payload": {
    "readUserGroups": [
      "UG1"
    ],
    "readWriteUserGroups": ["UG2"]
  }
}
```

#### Response

```
{
  "response": "Success",
  "message": null,
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Add a User Group

This API is used to add a new user group to the system.

**URL:** `/usergroup`

**Type:** POST

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String
Query	gwsource**	Source from which the request is triggered (for example, web, external).	String
Payload	name**	Name of the user group.	String
Payload	description	Description of the user group.	String

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	User group name cannot be null, empty or whitespace alone
201	Created	User group added successfully
409	Conflict	User group with the given name already exists

### Sample Request/Response

#### Use Case

Add a user group with a valid name.

#### Request URL

<https://appviewxapi.com/avxapi/usergroup?gwkey=f000ca01&gwsource=external>

## Request Payload

```
{
  "payload": {
    "name": "usergroup_1",
    "description": "Adding user group"
  }
}
```

## Response

```
{
  "response": " Usergroup added successfully",
  "message": " Usergroup added successfully",
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Associate Permissions to Role

This API is used to assign/un-assign ACF permissions to a role.

**URL:** */role-updatePermission*

**Type:** PUT

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	Gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String
Query	gwsource**	Source from which the request is triggered (for example, web, external).	String
Payload	roleName**	Name of the role.	String

Param Type	Name	Description	Field Type
Payload	assignPermissionList	ACF permissions to be assigned to the role. Refer to the list of possible permissions list for details.	Array
Payload	unassignPermissionList	ACF permissions to be unassigned from the role. Refer to the list of possible permissions list for details.	Array

\*\* - Mandatory value



**Note:** Even if the assignPermissionList and unassignPermissionList contain invalid ACF permission text that cannot be processed, the API would still go-ahead to assign/un-assign any functions in wither list that can be processed.

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	Role name cannot be left empty or blank.
202	Accepted	ACF assignment initiated for the given role.
400	Bad request	Permission list cannot be empty.

### Sample Request/Response

#### Use Case

Assign and unassign few ACF permissions to/from the role "role\_1".

#### Request URL

<https://appviewxapi.com/avxapi/role-updatePermission?gwkey=f000ca01&gwsorce=external>

#### Request Payload

```
{
  "payload": {
    "roleName": "role_1",
    "assignPermissionList": [
      "general:accounts:resource:clone",
      "general:accounts:resource:delete"
    ]
  }
}
```

```

],
"unassignPermissionList": [
"general:accounts:resource:delete"
]
}
}

```

**Response**

```

{
"response": "ACF assignment initiated for the given role",
"message": "ACF assignment initiated for the given role",
"appStatusCode": null,
"tags": null,
"headers": null
}

```

## Search User Group

This API is used to search / get the details about a user group. The user group can be filtered by the given input.

**URL:** */usergroup-search*

**Type:** POST

**Parameters**

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String
Query	gwsouce**	Source from which the request is triggered (Eg: web, external).	String
Payload	filterValue	Input filter value string to filter the role.	String
Payload	sortColumn	Column by which the output has to be sorted.	String

Param Type	Name	Description	Field Type
Payload	startIndex	Start index to show in output in case if there are more than one matching records.	Integer
Payload	maxSize	The number of records to return in a response. maxSize is starting from the start index.	Integer
Payload	sortOrder	Order of sorting. Possible values are asc and desc.	String

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
200	Accepted	Matching results found for the given input.
404	Bad request	No matching records found.

### Sample Request/Response

#### Use Case

Get the details about a user group with the name usergroup\_1.

#### Request URL

<https://appviewxapi.com/avxapi/usergroup-search?gwkey=f000ca01&gwsouce=external>

#### Request Payload

```
{
  "payload": {
    "startIndex": 1,
    "maxSize": 1,
    "sortColumn": "name",
    "sortOrder": "asc",
    "filterValue": "admin usergroup"
  }
}
```

#### Response

```

{
  "response":{
    "data":[
      {
        "name": "super access",
        "description": "Super access will have the permissions to access all the resources in AppViewX.",
        "state": "A",
        "_id": "super access",
        "_keywords":[
          "super access",
          "Super access will have the permissions to access all the resources in AppViewX.",
          "Enabled"
        ]
      }
    ],
    "totalRecords": 1,
    "obtainedRecords": 1,
    "obtainedRecordRange":{
      "start": 1,
      "end": 1
    },
    "message": "Matching results found for the given input.",
    "appStatusCode": null,
    "tags": null,
    "headers": null
  }
}

```

## Delete a User Group

This API is used to delete a user group.

**URL:** */userGroup*

**Type:** DELETE

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login	String
Query	gwkey	Tenant Key. Needed only in case of multitenant installations	String
Query	gwsource**	Source from which the request is triggered (for example, web, external)	String
Payload	userGroupName **	Name of the user group to be deleted	String

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	Invalid Request.
200	OK	UserGroup deleted successfully.
404	Not Found	UserGroup does not exist.
403	Forbidden	Current user(s) usergroup(s) cannot be deleted.

### Sample Request/Response

#### Use Case

Add a user group with the name group\_1.

#### Request URL

https://appviewxapi.com/avxapi/userGroup?userGroupName=group\_1&gwkey=f000ca01;gwsource=external

#### Request Payload

NA

#### Response

```
{
  "response": "UserGroup deleted successfully",
}
```

```

"message": "UserGroup deleted successfully",
"appStatusCode": null,
"tags": null,
"headers": null
}

```

## Associate User Group with Resources

This API is used to associate one or more resources to a user group.

**URL:** `/usergroup-updateResource`

**Type:** POST

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login	String
Query	Gwkey	Tenant Key. Needed only in case of multitenant installations	String
Query	gwsouce**	Source from which the request is triggered (For example, web, external)	String
Payload	filterValue**	Value to filter the name	String
Payload	assignPermissionList	ACF permissions to be assigned to the role	Array
Payload	unassignPermissionList	ACF permissions to be unassigned from the role	Array

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	User group not found.
200	Accepted	Successfully associated user group - resource.

## Sample Request/Response

### Use Case

Get the details about a user group with the name usergroup\_1.

### Request URL

<https://appviewxapi.com/avxapi/usergroup-updateResource?gwkey=f000ca01&gwsouce=external>

### Request Payload

```
{
  "payload": {
    "name": "testgroup",
    "assignList": [
      "resource_2"
    ],
    "unassignList": [
      "resource_1"
    ]
  }
}
```

### Response

```
{
  "response": "Updating usergroup with resource(s) successful",
  "message": "Updating usergroup with resource(s) successful",
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Add a User

This API is used to create a new user.

**URL:** */user*

**Type:** POST

## Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login	String
Query	gwkey	Tenant Key. Needed only in case of multitenant installations	String
Query	gwsource**	Source from which the request is triggered (Eg: web, external)	String
Payload	loginName**	Unique names used as the username for authentication. Duplication is not allowed	String
Payload	password**	A password that can be used to authenticate. Note: Not relevant if 'isExternalAuth' is set to Y	String
Payload	confirmPassword**	Reconfirm the given password  Note: Not relevant if 'isExternalAuth' is set to Y	String
Payload	isExternalAuth**	Specify if the user is from an external authentication system like LDAP, Radius, etc. Possible values are 'Y' for yes and 'N' for No	String
Payload	comment	Comment for the user	String
Payload	preferredContactMode**	Preferred contact mode. Possible values are 'E' for email.	String
Payload	email**	Email address	String

\*\* - Mandatory value

## Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	User password cannot be null or empty.
201	Created	User added successfully
409	Conflict	User name already exist.

## Sample Request/Response

### Use Case

Add a new internal user with login name **appviewx**.

### Request URL

<https://appviewxapi.com/avxapi/user?gwkey=f000ca01&gwsouce=external>

### Request Payload

```
{
  "payload": {
    "loginName": "appviewx",
    "password": "QWRtaW5AMTlz",
    "confirmPassword": "QWRtaW5AMTlz",
    "isExternalAuth": "N",
    "comment": "This is a test user",
    "preferredContactMode": "E",
    "email": "appviewx@app.com"
  }
}
```

### Response

```
{
  "response": "User added successfully",
  "message": "User added successfully",
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Associate Roles to a User Group

This API is used to associate or disassociate one or more roles to a user group.


**URL:** */usergroup-updateRoles*

**Type:** PUT

### Parameters

Param	TypeName	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String
Query	gwsource**	Source from which the request is triggered (for example, web, external).	String
Payload	assignRoles	An array of role names to assign.	Array
Payload	unassignRoles	An array of role names to unassign.	Array
Payload	name**	Name of the user-group for consideration.	string

\*\* - Mandatory value



**Note:** Even if the assignRoles and unassignRoles contain invalid role names that cannot be processed, the API would still go-ahead to assign/unassign any role in the list that can be successfully processed.

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	Usergroup Name cannot be null or empty.
200	OK	Updating user group with role(s) successful TBD – Change text, multiple c

### Sample Request/Response

#### Use Case

role\_1, role\_2, and role\_3 are existing in the application. role\_1 and role\_2 have to be assigned to the user group *appviewx\_usergroup* and role\_3 has to be unassigned.

#### Request URL

<https://appviewxapi.com/avxapi/platform-assign-roles-to-usergroup?gwkey=f000ca01&gwsource=external>

#### Request Payload

```
{
  "payload": {
    "userGroupName": "UG1", "assignedRoles": ["R1"], "unassignedRoles": []
  }
}
```

## Response

```
{
  "response": "success",
  "message": null,
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Associate User Groups to a User

This API is used to associate or disassociate one or more user groups to a user.

**URL:** */user-updateUsergroups*

**Type:** PUT

### Parameters

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String
Query	gwsouce**	Source from which the request is triggered (for example, web, external).	String
Payload	assignUserGroups	An array of user group names to assign.	Array
Payload	unAssignUserGroups	An array of user group names to unassign.	Array
Payload	loginName**	Login name of the user.	String

\*\* - Mandatory value



**Note:** Even if the assignUserGroups and unAssignUserGroups contain invalid user group names that cannot be processed, the API would still go-ahead to assign/un-assign any user group in the list that can be successfully processed.

**Possible Response Message and Code**

HTTP Code	Description	Response Message
400	Bad request	Usergroup name cannot be null or empty.
200	OK	Updating user group with a role(s) successful TBD – Change text, multiple cases.

**Sample Request/Response**

**Use Case**

"usergroup\_1, usergroup\_2, and usergroup\_3 are existing in the application. usergroup\_1 and usergroup\_2 have to be assigned to the user with loginname user\_1 and usergroup\_3 has to be unassigned"

**Request URL**

<https://appviewxapi.com/avxapi/user-updateUsergroups?gwkey=f000ca01&gwsorce=external>

**Request Payload**

```
{
  "payload": {
    "loginName": "appviewx",
    "assignUserGroups": [
      "usergroup_1",
      "usergroup_2"
    ],
    "unassignUserGroups": [
      "usergroup_3"
    ]
  }
}
```

```
}
}
```

**Response**

```
{
  "response": "Successfully modified user-user group association",
  "message": " Successfully modified user-user group association ",
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Search for a user

This API is used to search for a user using filter conditions.

**URL:** */user-search*

**Type:** POST

**Parameters**

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String
Query	gwsource**	Source from which the request is triggered (for example, web, external).	String
Payload	filterValue	Input filter value string to filter the role.	String
Payload	sortColumn	Column by which the output has to be sorted.	String
Payload	startIndex	Start index to show in output in case if there are more than one matching records.	Integer
Payload	maxSize	The number of records to return in a response. maxSize is starting from the start index	Integer

Param Type	Name	Description	Field Type
Payload	sortOrder	Order of sorting. Possible values are asc and desc	String

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
404	Not Found	No matching records found.
200	OK	Matching results found for the given input.

### Sample Request/Response

#### Use Case

Search for use with name appviewx\_user1.

#### Request URL

<https://appviewxapi.com/avxapi/user-search?gwkey=f000ca01&gwsouce=external>

#### Request Payload

```
{
  "payload": {
    "startIndex": 1,
    "maxSize": 2,
    "sortColumn": "loginName",
    "sortOrder": "asc",
    "filterValue": "appviewx_user1"
  }
}
```

#### Response

```
{
  "response": {
    "data": [
      {
        "loginName": "appviewx_user1",

```

```
"firstName": "Appviewx",
"lastName": "user1",
"comment": "Appviewx login user",
"email": "",
"mobile": "",
"preferredContactMode": "",
"isExternalAuth": "N",
"state": "A",
"roles": null,
"swImages": null,
"authenticationMode": "Internal",
"available": "Active",
"lastLogin": "Online",
"sessionIds": null,
"aclIdentifiers": null,
"tenantId": null,
"availableIncorrectAttempt": 10,
"lastLoginTimestamp": 1585669200198,
"adUserGroups": null,
"sso": false,
"date": "2020-04-13 12:05:09",
"_id": "5d258291d7f495bb31691fc4",
"userGroups": null,
"_keywords": null,
"isSSO": false
}
],
"totalRecords": 1,
"obtainedRecords": 1,
"obtainedRecordRange": {
"start": 1,
"end": 1
}
},
"message": "Matching results found for the given input.",
"appStatusCode": null,
"tags": null,
```

```
"headers": null
}
```

## Assign User group to Service Account

This API is used to create a new resource. A resource is a logical entity to group one or more ACL managed entities.

**URL:** avxapi/platform-assign-usergroup-to-service-account?gwkey=f000ca01&gwsourc=web

**Method** - Post

**Response Format:** JSON

**Requires authentication:** Yes

**Request timeout period:**15 minutes

Param Type	Name	Description	Field Type
Header	sessionId** (or) Token**	Use either Session Id or Token received after login.	String
Query	assignedUserGroup	Mention the array of user groups to be assigned.	String
Payload	loginName**	Unique names used as the service account name for authentication. Duplication is not allowed.	String
Payload	unassignedUserGroup	Mention the array of user groups to be unassigned.	String

\*\* - Mandatory value

### Possible Response Message and Code

HTTP Code	Description	Response Message
400	Bad request	Mandatory field 'name' is missing.
201	Created	Resource added successfully.
409	Conflict	Resource with the given name already exists.
407	Proxy Authentication Required	The Username and Password field is mandatory.

## Sample Request/Response

### Use Case

To assign/unassign user groups to service account.

### Request URL

avxapi/platform-assign-usergroup-to-service-account?gwkey=f000ca01&gwsouce=web

### Request Payload

```
{
  "payload": {
    "loginName": "internal90",
    "assignedUserGroup": [
      "admin usergroup"
    ],
    "unassignedUserGroup": [
    ]
  }
}
```

### Response

```
{
  "response": "success",
  "message": null,
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Update Resource Description

This API is used to update existing resource descriptions.

## Before you begin

- Session ID is obtained from the login API.
- Ensure that the session ID is valid and has not expired.
- Make sure you have valid login credentials for accessing the system.

## Request Structure

<b>Endpoint:</b>	/platform-update-aclidentifier
<b>Type:</b>	PUT
<b>Sample URL:</b>	https://<Tenant_name/Host_name>:<portno>/avxapi/platform-update-aclidentifier? gwkey=f000ca01&gwsorce=external
<b>Headers:</b>	
<b>Content-Type:</b>	application/json

**Table 1. Input Parameters**

Name	Description
sessionID <i>Header</i>	(Mandatory) Use session ID retrieved from login API is required if username and password are not provided.  <b>Example:</b> "sessionId": "ce7f1a14-2bf9-4e4a-89a8-bc780a255813"
username <i>Header</i>	(Mandatory) AppViewX login username is required if sessionId is not provided.  <b>Example:</b> "admin"
Password <i>Header</i>	(Mandatory) AppViewX login password is required if sessionId is not provided..  <b>Example:</b> "AppViewX@123"
Content-Type <i>Header</i>	(Mandatory) The parameter should be set to <code>application/json</code> to specify the nature of the data in the payload.  <b>Example:</b> "application/json"
gwkey	(Optional) Tenant Key. Needed only in the case of multi-tenant installations which is a string value.

**Table 1. Input Parameters (continued)**

Name	Description
<i>Query</i>	
gwsource	(Mandatory) Source from which the request is triggered.
<i>Query</i>	The values can be: <ul style="list-style-type: none"> <li>• web</li> <li>• external</li> </ul>
name	(Mandatory) Name of the resource whose description needs to be updated.
<i>Payload</i>	<b>Example:</b> "myResource"
<i>String</i>	
description	(Mandatory) Description that needs to be updated.
<i>Payload</i>	<b>Example:</b> "Sample resource for test"
<i>String</i>	

## Response Structure

- **Status Code:** 200 OK
- **Message:** null.
- **Headers:**
  - **Content-Type:** application/json

**Table 2. Response Parameters**

Name	Description
response	Contains the response attributes for search of role successfully.
message	Success message or failure description in case of error.
appStatusCode	Application specific status code for the response. Will be non-null for failure response.
tags	More info in case of failure response.

## Status Codes

HTTP Code	appStatusCode	Response Message
200 OK	null	null
400 Bad request	VALIDATION_ERROR_0006	Field(s) does not comply with the validation criteria. Please recheck the input payload:: [description::VALIDATION_ERROR_0006:: Input does not match the pattern configured]  Field(s) does not comply with the validation criteria. Please recheck the input payload:: [name::VALIDATION_ERROR_0006:: Input does not match the pattern configured].
404 Not found	RBAC_RE_005	Resource(s) not found.
407 Proxy Authentication Required	AVX_GW_011	Session validation failed, reason - Session information is missing.

## Sample Request/Response

### Use Case

To update the existing resource descriptions.

### Sample Request

```
https://<Tenant_name/Host_name>:<portno>/avxapi/platform-update-acidentifier?gwkey=f000ca01&gwsorce=external
```

### Request Payload

```
{name: "myResource", description: "Sample resource for test"}
```

### Sample Response

```
{
  "response": null,
  "message": null,
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Update User Group Description

This API is used to update the existing user group descriptions.

### Before you begin

- Session ID is obtained from the login API.
- Ensure that the session ID is valid and has not expired.
- Make sure you have valid login credentials for accessing the system.

### Request Structure

<b>Endpoint:</b>	/acctmgmt-update-usergroup
<b>Type:</b>	PUT
<b>Sample URL:</b>	https://<Tenant_name/Host_name>:<portno>/avxapi/acctmgmt-update-usergroup? gwkey=f000ca01&gwsorce=external
<b>Headers:</b>	
<b>Content-Type:</b>	application/json

**Table 3. Input Parameters**

Name	Description
sessionID <i>Header</i>	(Mandatory) Use session ID retrieved from login API is required if username and password are not provided.  <b>Example:</b> "sessionId": "ce7f1a14-2bf9-4e4a-89a8-bc780a255813"
username <i>Header</i>	(Mandatory) AppViewX login username is required if sessionId is not provided.  <b>Example:</b> "admin"
Password <i>Header</i>	(Mandatory) AppViewX login password is required if sessionId is not provided..  <b>Example:</b> "AppViewX@123"
Content-Type <i>Header</i>	(Mandatory) The parameter should be set to <code>application/json</code> to specify the nature of the data in the payload.

**Table 3. Input Parameters (continued)**

Name	Description
	<b>Example:</b> "application/json"
gwkey <i>Query</i>	(Optional) Tenant Key. Needed only in the case of multi-tenant installations which is a string value.
gwsource <i>Query</i>	(Mandatory) Source from which the request is triggered.  The values can be: <ul style="list-style-type: none"> <li>• web</li> <li>• external</li> </ul>
name  <i>Payload</i> <i>String</i>	(Mandatory) Name of the resource whose description needs to be updated.  <b>Example:</b> "myGroup"
description  <i>Payload</i> <i>String</i>	(Mandatory) Description that needs to be updated.  <b>Example:</b> "Sample usergroup for test"

## Response Structure

- **Status Code:** 200 OK
- **Message:** null.
- **Headers:**
  - **Content-Type:** application/json

**Table 4. Response Parameters**

Name	Description
response	Contains the response attributes for search of role successfully.
message	Success message or failure description in case of error.

**Table 4. Response Parameters (continued)**

Name	Description
appStatusCode	Application specific status code for the response. Will be non-null for failure response.
tags	More info in case of failure response.

### Status Codes

HTTP Code	appStatusCode	Response Message
200 OK	null	null
400 Bad request	ACCT_USERGROUP_001	Invalid usergroup name.
400 Bad request	ACCT_USERGROUP_025	Invalid description.
407 Proxy Authentication Required	AVX_GW_011	Session validation failed, reason - Session information is missing.

### Sample Request/Response

#### Use Case

To update the existing user group descriptions.

#### Sample Request

```
https://<Tenant_name/Host_name>:<portno>/avxapi/acctmgmt-update-usergroup?gwkey=f000ca01&gwsouce=external
```

#### Request Payload

```
{name: "myGroup", description: "Sample usergroup for test"}
```

#### Sample Response

```
{
  "response": null,
  "message": null,
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

## Chapter 5: Glossary

This section describes common terms used in this guide.

Term/Abbreviation	Expansion
ACL	Access Control List
API	Application Programming Interface
HSM	Hardware Security Module
HTTPS	Hypertext Transfer Protocol Secure
SMTP	Simple Mail Transfer Protocol
URI	Uniform Resource Indicator