# Table of Contents

**Preface** ........................................................................................................................................... 4  
About Qualys ...................................................................................................................................... 4  
Contact Qualys Support .................................................................................................................. 4  

**Get Started** .................................................................................................................................. 5  
Qualys API Framework ..................................................................................................................... 5  
Qualys API Gateway URL .................................................................................................................. 6  
Introduction to GAV/CSAM API Paradigm ....................................................................................... 7  
API Rate Limits ................................................................................................................................. 9  
User Scoping for APIs ....................................................................................................................... 11  

**Assets Host Data APIs** .............................................................................................................. 12  
Count of Assets ............................................................................................................................... 12  
Get Host details of specific asset ..................................................................................................... 15  
Get Host Details of All Assets .......................................................................................................... 24  
Import Business Information Metadata ......................................................................................... 37  

**Import Business Information Metadata** .................................................................................. 41  
Import Business App Metadata ....................................................................................................... 41  
Import Asset Business Metadata .................................................................................................... 43  

**Appendix** ..................................................................................................................................... 47  
Error Messages ................................................................................................................................. 47  
Supported Operators ....................................................................................................................... 47
Preface

This user guide is intended for application developers who will use the Qualys Global AssetView (GAV)/CyberSecurity Asset Management (CSAM) API v2. It is recommended to use v2 APIs.
For GAV, we are still supporting the v1 APIs. Refer API v1 User Guide to use v1 APIs.

About Qualys

Qualys, Inc. (NASDAQ: QLYS) is a pioneer and leading provider of cloud-based security and compliance solutions. The Qualys Cloud Platform and its integrated apps help businesses simplify security operations and lower the cost of compliance by delivering critical security intelligence on demand and automating the full spectrum of auditing, compliance and protection for IT systems and web applications.

Founded in 1999, Qualys has established strategic partnerships with leading managed service providers and consulting organizations including Accenture, BT, Cognizant Technology Solutions, Deutsche Telekom, Fujitsu, HCL, HP Enterprise, IBM, Infosys, NTT, Optiv, SecureWorks, Tata Communications, Verizon and Wipro. The company is also a founding member of the Cloud Security Alliance (CSA). For more information, please visit www.qualys.com.

Contact Qualys Support

Qualys is committed to providing you with the most thorough support. Through online documentation, telephone help, and direct email support, Qualys ensures that your questions will be answered in the fastest time possible. We support you 7 days a week, 24 hours a day. Access support information at www.qualys.com/support/.
Get Started

Qualys API Framework - Learn the basics about making API requests. The base URL depends on the platform where your Qualys account is located.

Introduction to GAV/CSAM API Paradigm - Get tips on using the Curl command-line tool to make API requests. Every API request must authenticate using a JSON Web Token (JWT) obtained from the Qualys Authentication API.

Get API Notifications
Subscribe to our API Notifications RSS Feeds for announcements and latest news.

Qualys API Framework
The Qualys GAV/CSAM API uses the following framework.

Request URL
The URL for making API requests respects the following structure:
https://<baseurl>/<module>/<object>/<object_id>/<operation>
where the components are described below.

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;baseurl&gt;</td>
<td>The Qualys API server URL that you should use for API requests depends on the platform where your account is located. The base URL for Qualys US Platform 1 is: <a href="https://gateway.qg1.apps.qualys.com">https://gateway.qg1.apps.qualys.com</a></td>
</tr>
<tr>
<td>&lt;module&gt;</td>
<td>The API module. For the GAV/CSAM, the module is: “am”.</td>
</tr>
<tr>
<td>&lt;object&gt;</td>
<td>The module specific object.</td>
</tr>
<tr>
<td>&lt;object_id&gt;</td>
<td>(Optional) The module specific object ID, if appropriate.</td>
</tr>
<tr>
<td>&lt;operation&gt;</td>
<td>The request operation, such as count.</td>
</tr>
</tbody>
</table>
Qualys API Gateway URL

The Qualys API URL you should use for API requests depends on the Qualys platform where your account is located.

Click here to identify your Qualys platform and get the API URL

This documentation uses the API gateway URL for Qualys US Platform 1 (https://gateway.qg1.apps.qualys.com) in sample API requests. If you’re on another platform, please replace this URL with the appropriate gateway URL for your account.
Introduction to GAV/CSAM API Paradigm

Authentication

You must authenticate to the Qualys Cloud Platform using Qualys account credentials (user name and password) and get the JSON Web Token (JWT) before you can start using the GAV/CSAM APIs. Use the Qualys Authentication API to get the JWT.

For example,

```bash
```

where gateway.qg1.apps.qualys.com is the base URL to the Qualys API server where your account is located.

- **username** and **password** are the credentials of the user account for which you want to fetch GAV/CSAM data
- **token** should be true
- **Content-Type** should be "application/x-www-form-urlencoded"

The Authentication API returns a JSON Web Token (JWT) which you can use for authentication during GAV/CSAM calls. The **token expires in 4 hours**. You must regenerate the token to continue using the GAV/CSAM API.

Using Curl

**Curl** is a multi-platform command-line tool used to transfer data using multiple protocols. This tool is supported on many systems, including Windows, Unix, Linux and Mac. In this document Curl is used in the examples to build Qualys API requests using the HTTP over SSL (https) protocol, which is required.

Want to learn more? Visit [https://curl.haxx.se/](https://curl.haxx.se/)

The following Curl options are used according to different situations:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-X &quot;POST&quot;</code></td>
<td>The POST method is required for all GAV/CSAM API requests.</td>
</tr>
<tr>
<td><code>-H &quot;Authorization: Bearer &lt;token&gt;&quot;</code></td>
<td>This option is used to provide a custom HTTP request header parameter for authentication. Provide the JSON Web Token (JWT) received from Qualys authentication API in the following format: Authorization: Bearer &lt;token&gt; For information about Qualys authentication API, see <a href="#">Authentication</a>.</td>
</tr>
</tbody>
</table>

The sample below shows a typical Curl request using options mentioned above and how they interact with each other.

```bash
```
Limit your results

Use the optional “fields” parameter for any API request to limit the amount of information returned in the results. Simply specify the fields you want to include or exclude in the output, and all other information will be filtered out (excluded). Multiple fields are comma separated.

Sample limit results

Use this request to get a list of all asset hosts with information for only the operatingSystem and hardware fields:


Note:

The response would still include all the fields, but other than the included fields, the value returned for all other fields would be null.

You can include the following fields to limit your results:

- address
- lastLocation
- agent
- lastLoggedOnUser
- agentId
- netbiosName
- assetName
- networkInterface
- biosAssetTag
- openPort
- biosDescription
- operatingSystem
- biosSerialNumber
- processor
- cloudProvider
- provider
- container
- sensor
- cpuCount
- service
- dnsName
- software
- hardware
- tag
- hostId
- timeZone
- inventory
- totalMemory
- isContainerHost
- userAccount
- lastBoot
- volume
- criticality
- businessApps
- businessInformation
- assignedLocation
**API Rate Limits**

The Qualys API enforces limits on the API calls a customer can make based on their subscription settings. The limits apply to the use of all Qualys APIs except “auth” API (JWT Token Generation API). Default API control settings are provided by the service. Note these settings may be customized per subscription by Qualys Support.

The rate count and period are calculated dynamically each time an API call is received. The rate period represents a rolling window when API calls are counted.

**API Controls Definition**

**X-RateLimit-Remaining**: This indicates the total API calls remaining in current rate limit window.

**X-RateLimit-ToWait-Sec**: This time indicates the wait time for the rate limit to be reset. The customer has to wait for that time to execute next API calls.

**X-RateLimit-Window-Sec**: This value indicates the total time window assigned for the APIs to be executed.

**X-RateLimit-Limit**: This indicates the max number of API calls that can be executed in that particular rate limit window.

**Sample Request**


**Note**: Provide "-i" in the curl request as shown in the example returns the response headers which includes the rate limit related parameters.

After executing a curl request, check the following parameters in response headers to check the rate-limit status:

- X-RateLimit-Remaining: 0
- X-RateLimit-ToWait-Sec: 300
- X-RateLimit-Window-Sec: 3600
- X-RateLimit-Limit: 300

**Example**: A subscription for Standard API Service has the default API control settings. Consider that the API rate limit set for a customer is 300 API calls for a time window of 3600 seconds. If 300 API calls are received in a 5 minute period and none are blocked by any API limiting rules, then you need to wait 55 minutes before making the next call to the API. During the wait period API calls will be blocked by the rate limiting rule.
Sample HTTP Response Headers

Sample 1: Normal API call (API call not blocked)

Server: nginx/1.19.1
Date: Fri, 16 Apr 2021 12:29:52 GMT
Content-Type: application/json
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Accept-Encoding
X-RateLimit-Remaining: 4
X-RateLimit-Window-Sec: 100
X-RateLimit-Limit: 5
Vary: Accept-Encoding
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
X-Content-Type-Options: nosniff
X-Frame-Options: DENY
X-XSS-Protection: 1 ; mode=block
Referrer-Policy: no-referrer

{"count":580,"responseCode":"SUCCESS","responseMessage":"Valid API Access"}

Sample 2: API Call Blocked - Rate Limit exceeded

Server: nginx/1.19.1
Date: Fri, 16 Apr 2021 12:28:53 GMT
Content-Length: 0
Connection: keep-alive
X-RateLimit-Remaining: 0
X-RateLimit-ToWait-Sec: 33
X-RateLimit-Window-Sec: 100
X-RateLimit-Limit: 5
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
X-Content-Type-Options: nosniff
X-Frame-Options: DENY
X-XSS-Protection: 1 ; mode=block
Referrer-Policy: no-referrer
User Scoping for APIs

The user will get a response of the assets as per scope using count and list APIs. Typically Manager user has access to more assets than the reader user. So, Reader user won’t be able to get responses for the APIs requested for unauthorized assets.

For example:

If a Manager user has access to 100 assets: the Count API response will show 100 assets and List API will show details of all these 100 assets. Now, consider that the Manager user creates a ‘Reader’ sub user and assigns only 50 assets to this user and when Reader executes APIs, the response will contain data of the only 50 assets and not all 100 assets.
Assets Host Data APIs

Use these API functions to get host data from GAV/CSAM.

**Note:** The software.authorization and lifecycle related parameters are available only for CSAM subscription. Hence, you can use it in filter criteria and you can see it in the response if you’ve subscribed for CSAM.

**Permissions**
- User must have the GAV/CSAM module and the "App API Enabled" option enabled for that role.

**Count of Assets**
Get count of assets satisfying the specified filter criteria.

`rest/2.0/count/am/asset`

[POST]

**Input Parameters**

| filter (String) | Filter the events list by providing a filter in json and xml format. Make sure your filter criteria is provided in xml/json format in the request body. If you don't provide filter parameter, it will show details of all the assets. For more information on supported operators, refer [Supported Operators](#).

For example (json) -

```json
{
    "filters": [
        {
            "field": "software.product",
            "operator": "CONTAINS",
            "value": "Python"
        }
    ]
}
```

For example (xml) -

```xml
<FilterRequest>
    <filters>
        <Criteria field="software.product" operator="CONTAINS">Python</Criteria>
    </filters>
</FilterRequest>
```
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| assetLastUpdated (String)| Shows records updated on or after this date with the UTC format as yyyy-MM-ddTHH:mmZ e.g. 2019-03-01T11:30Z. This date gets updated whenever any activity happens on the asset. Few examples of such activity:  
- Vulnerability Management scan  
- Policy Compliance scan  
- Inventory collection  
- Security Configuration Assessment  
- CertView scan  
- AssetView or CloudView connector run  
- Secure Enterprise Mobility scan  
- Out-of-Band Configuration Assessment  
- Asset rename  
- Purge of VM, PC, OCA, CertView records  
- Agent manifest download  
- Asset Inventory asset identification updates |
| lastSeenAssetId (Integer)| Use to get the count of assets having asset id greater than the specified last seen assetid.  
**Note:** If you want to get a count of assets that fall after the specific asset id, refer to the following example:  
**Example:**  
https://gateway.qg1.apps.qualys.com/rest/2.0/count/am/asset?lastSeenAssetId=6920718  
**Consider the scenario, wherein you have 1000 assets.**  
In the API call, if you enter the asset id of the 200th asset in the lastSeenAssetId parameter, the first 200 assets are skipped from the count and the count of the rest of the assets, which is 800 is shown. |
| Authorization (String)   | (Required) Authorization token to authenticate to the Qualys Cloud Platform. Prepend token with "Bearer" and one space. For example - Bearer authToken |
Sample - Get count of all assets with filter criteria

**Request:**


Here, **filter.json** file is the request in json format.

**Sample Request body in json format (filter.json)**

```json
{
  "filters": [
    {
      "field": "software.authorization",
      "operator": "EQUALS",
      "value": "Authorized"
    }
  ]
}
```

**Response:**

```json
{
  "count": 850,
  "responseCode": "SUCCESS",
  "responseMessage": "Valid API Access"
}
```
Get Host details of specific asset

Get details of specific asset by providing an asset id.

```
rest/2.0/get/am/asset
```

[GET]

**Input Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>excludeFields (String)</td>
<td>Comma separated list of fields to be excluded from the asset object in the response. Default is None. You can choose from the list of fields specified in the section &quot;Limit your results&quot;. For example, to exclude openPort and software from the response: excludeFields=openPort,software</td>
</tr>
<tr>
<td>includeFields (String)</td>
<td>Comma separated list of fields to be included in the asset object in the response. Default is All. You can choose from the list of fields specified in the section &quot;Limit your results&quot;. For example, to include only operatingSystem and hardware in the response: includeFields=operatingSystem,hardware</td>
</tr>
<tr>
<td>assetId (Integer)</td>
<td>(Required) Use to specify theAssetId for which you want to retrieve the details.</td>
</tr>
<tr>
<td>softwareType (String)</td>
<td>(Optional) Specify the software type to get the host details for specified software type. Available values for softwareType parameter are Application, Unknown, and Others.</td>
</tr>
<tr>
<td>Authorization (String)</td>
<td>(Required) Authorization token to authenticate to the Qualys Cloud Platform. Prepend token with &quot;Bearer&quot; and one space. For example - Bearer authTokn</td>
</tr>
</tbody>
</table>

**Sample - Get host details of specified asset by assetid**

**Request:**

```
-H "Content-Type: application/json" -i 
"https://gateway.qgl.apps.qualys.com/rest/2.0/get/am/asset?assetId=8194990"
```

**Response:**

```
{  
    "responseMessage": "Valid API Access",
    "responseCode": "SUCCESS",
    "assetListData": {  
        "asset": [
            {  
                "assetId": 6920718,  
                "assetUUID": "50d20290-c66a-42e7-8c0a-  
            }
        ]
    }
}
```

15
ba6e92b6324c",
"hostId": 1437386,
"lastModifiedDate": "2021-04-06T10:02:33.000Z",
"agentId": null,
"createdDate": "2020-11-25T12:49:25.000Z",
"sensorLastUpdatedDate": "2021-04-06T10:02:33.000Z",
"assetType": "HOST",
"address": "10.115.110.95",
"dnsName": "localhost.localdomain",
"assetName": "localhost.localdomain",
"netbiosName": null,
"timeZone": "IST",
"biosDescription": null,
"lastBoot": null,
"totalMemory": 5806,
"cpuCount": null,
"lastLoggedOnUser": "root",
"hwUUID": "422a2b16-4c8b-588a-a20c-c1851ad7e376",
"biosSerialNumber": "VMware-42 2a 2b 16 4c 8b 58 8a-a2 0c c1 85 1a d7 e3 76",
"biosAssetTag": "No Asset Tag",
"isContainerHost": false,
"operatingSystem": {
  "osName": "The CentOS Project CentOS 7 (1810)",
  "fullName": "The CentOS Project CentOS 7 (1810)",
  "category": "Linux / Server",
  "category1": "Linux",
  "category2": "Server",
  "productName": "CentOS",
  "publisher": "The CentOS Project",
  "edition": null,
  "marketVersion": "7",
  "version": "1810",
  "update": null,
  "architecture": null,
  "lifecycle": {
    "gaDate": "2018-12-03T00:00:00.000Z",
    "eolDate": "2020-12-31T00:00:00.000Z",
    "eosDate": "2024-06-30T00:00:00.000Z",
    "stage": "EOL",
    "lifecycleConfidence": "Exact",
    "eolSupportStage": "Full updates",
    "eosSupportStage": "Maintenance Updates"
  }
},

16
"taxonomy": { 
  "id": null,
  "name": "Linux / Server",
  "category1": "Linux",
  "category2": "Server"
},
"productFamily": null,
"installDate": null
},
"hardware": { 
  "fullName": "VMware VMware Virtual Platform
VMware Virtual Platform",
  "category": "Virtualized / Virtual Machine",
  "category1": "Virtualized",
  "category2": "Virtual Machine",
  "manufacturer": "VMware",
  "productName": "VMware Virtual Platform",
  "model": "VMware Virtual Platform",
  "lifecycle": {
    "introDate": null,
    "gaDate": null,
    "eosDate": null,
    "obsoleeteDate": null,
    "stage": "Unknown",
    "lifeCycleConfidence": ""
  },
  "taxonomy": { 
    "id": null,
    "name": "Virtualized / Virtual Machine",
    "category1": "Virtualized",
    "category2": "Virtual Machine"
  },
  "productUrl": "https://www.linuxjournal.com/article/3458,",
  "productFamily": null
},
"userAccountListData": null,
"openPortListData": {
  "openPort": [
    {
      "port": 709,
      "description": "",
      "protocol": "UDP",
      "detectedService": "portmap/rpcbind",
    }
  ]
}
Get Host details of specific asset

"firstFound": "2020-11-25T12:46:42.000Z",
"lastUpdated": "2020-11-25T12:46:42.000Z"
},
{
  "port": 50000,
  "description": "",
  "protocol": "TCP",
  "detectedService": "IBM_DB2_Universal_Database",
  "firstFound": "2020-11-25T12:46:42.000Z",
  "lastUpdated": "2020-11-25T12:46:42.000Z"
},
{
  "port": 6000,
  "description": "",
  "protocol": "TCP",
  "detectedService": "x11",
  "firstFound": "2020-11-25T12:46:42.000Z",
  "lastUpdated": "2020-11-25T12:46:42.000Z"
},
{
  "port": 22,
  "description": "",
  "protocol": "TCP",
  "detectedService": "ssh",
  "firstFound": "2020-11-25T12:46:41.000Z",
  "lastUpdated": "2020-11-25T12:46:41.000Z"
},
{
  "port": 3389,
  "description": "",
  "protocol": "TCP",
  "detectedService": null,
  "firstFound": "2020-11-25T12:46:42.000Z",
  "lastUpdated": "2020-11-25T12:46:42.000Z"
},
}
Assets Host Data APIs
Get Host details of specific asset

```json
{
    "port": 111,
    "description": "",
    "protocol": "UDP",
    "detectedService": "rpc_udp",
    "firstFound": "2020-11-25T12:46:42.000Z",
    "lastUpdated": "2020-11-25T12:46:42.000Z"
},
{
    "port": 111,
    "description": "",
    "protocol": "TCP",
    "detectedService": "rpc",
    "firstFound": "2020-11-25T12:46:41.000Z",
    "lastUpdated": "2020-11-25T12:46:41.000Z"
}
}

"volumeListData": {
    "volume": [
        {
            "name": "tmpfs",
            "free": 2737078272,
            "size": 3043934208
        },
        {
            "name": "/dev/mapper/centos-home",
            "free": 18629619712,
            "size": 18700304384
        },
        {
            "name": "devtmpfs",
            "free": 3026444288,
            "size": 3026444288
        },
        {
            "name": "/dev/mapper/centos-root",
            "free": 19672580096,
            "size": 38304645120
        },
        {
            "name": "/dev/sda1",
```
"free": 876040192,
"size": 1063256064
]
],
"networkInterfaceListData": {
"networkInterface": [
{
    "hostname": "localhost.localdomain",
    "addressIpV4": "192.168.122.1",
    "addressIpV6": null,
    "macAddress": "52:54:00:77:e1:71",
    "interfaceName": "virbr0",
    "dnsAddress": null,
    "gatewayAddress": "",
    "manufacturer": null,
    "macVendorIntroDate": null,
    "addresses": null
},
{
    "hostname": "localhost.localdomain",
    "addressIpV4": "10.115.110.95",
    "addressIpV6": "fe80:0:0:0:250:56ff:feaa:e2da",
    "macAddress": "00:50:56:aa:e2:da",
    "interfaceName": "ens192",
    "dnsAddress": null,
    "gatewayAddress": "",
    "manufacturer": "VMware",
    "macVendorIntroDate": 946944000000,
    "addresses": null
}
]
},
"softwareListData": {
"software": [
{
    "id": -5698725809391962787,
    "fullName": "Python 2.7.5 64-Bit",
    "softwareType": "Application",
    "isIgnored": false,
    "ignoredReason": null,
    "category": "Application Development / Programming Languages",
    "category1": "Application Development",
    "category2": "Programming Languages"}]}
"productName": "Python",
"component": null,
"publisher": "Python",
"edition": null,
"marketVersion": "2",
"version": "2.7",
"update": "2.7.5",
"architecture": "64-Bit",
"installDate": "2020-03-27T16:11:47.000Z",
"installPath": null,
"lastUpdated": "2020-11-25T12:46:46.000Z",
"lastUseDate": null,
"language": null,
"formerlyKnownAs": null,
"isPackage": false,
"isPackageComponent": false,
"packageName": null,
"lifecycle": {
  "gaDate": "2010-07-03T00:00:00.000Z",
  "eolDate": "2020-01-01T00:00:00.000Z",
  "eosDate": "2020-01-01T00:00:00.000Z",
  "stage": "EOL/EOS",
  "lifeCycleConfidence": "Exact",
  "eolSupportStage": "End-of-life",
  "eosSupportStage": "End-of-life"
},
"supportStageDesc": "Python's policy is to drop support major versions once they reach their end of life",
"license": {
  "category": "Open Source",
  "subcategory": "Python License (Python-2.0)"
},
"authorization": "Authorized"
}
"isIgnored": false,
"ignoredReason": null,
"category": "Networking / Access Software",
"category1": "Networking",
"category2": "Access Software",
"productName": "OpenSSH",
"component": "Server",
"publisher": "OpenBSD",
"edition": null,
"marketVersion": "7",
"version": "7.4",
"update": "7.4p1",
"architecture": null,
"installDate": "2020-03-05T14:23:53.000Z",
"installPath": null,
"lastUpdated": "2020-11-25T12:46:53.000Z",
"lastUseDate": null,
"language": null,
"formerlyKnownAs": "OpenBSD Secure Shell",
"isPackage": true,
"isPackageComponent": false,
"packageName": null,
"lifecycle": {
  "gaDate": "2016-12-19T00:00:00.000Z",
  "eolDate": null,
  "eosDate": null,
  "stage": "EOL",
  "lifeCycleConfidence": "Calculated",
  "eolSupportStage": "",
  "eosSupportStage": ""
},
"supportStageDesc": null,
"license": {
  "category": "Open Source",
  "subcategory": "BSD 2-Clause License (FreeBSD/Simplified)"
},
"authorization": "Authorized"}
Get Host details of specific asset

```
{
  "provider": null,
  "cloudProvider": null,
  "agent": null,
  "sensor": {
    "activatedForModules": [
      "VM"
    ],
    "pendingActivationForModules": [],
    "lastVMScan": 1606306572000,
    "lastComplianceScan": 0,
    "lastFullScan": 1606306572000
  },
  "container": null,
  "inventory": {
    "source": "IP",
    "created": 1606308565000,
    "lastUpdated": 1617703353000
  },
  "activity": null,
  "tagList": {
    "tag": [
      {
        "tagId": 14151022,
        "tagName": "static split",
        "foregroundColor": 0,
        "backgroundColor": -65536,
        "businessImpact": null,
        "criticalityScore": 2
      }
    ]
  },
  "serviceList": null,
  "lastLocation": null,
  "criticality": {
    "score": 2,
    "isDefault": true,
    "lastUpdated": "2021-06-30T09:43:27.000Z"
  },
  "processor": null
}
```
Get Host Details of All Assets

Get details of all assets that satisfy the filter criteria to include or exclude specified fields. If you don’t provide filter parameter, it will show details of all the assets.

rest/2.0/search/am/asset

[POST]

Input Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>excludeFields (String)</td>
<td>Comma separated list of fields to be excluded from the asset object in the response. Default is None. You can choose from the list of fields specified in the section “Limit your results”. For example, to exclude openPort and software from the response: excludeFields=openPort,software</td>
</tr>
<tr>
<td>includeFields (String)</td>
<td>Comma separated list of fields to be included in the asset object in the response. Default is All. You can choose from the list of fields specified in the section “Limit your results”. For example, to include only operatingSystem and hardware in the response: includeFields=operatingSystem,hardware</td>
</tr>
<tr>
<td>assetLastUpdated (String)</td>
<td>Shows records updated on or after this date with the UTC format as yyyy-MM-ddTHH:mmZ e.g. 2019-03-01T11:30Z. This date gets updated whenever any activity happens on the asset. Few examples of such activity: - Vulnerability Management scan - Policy Compliance scan - Inventory collection - Security Configuration Assessment - CertView scan - AssetView or CloudView connector run - Secure Enterprise Mobility scan - Out-of-Band Configuration Assessment - Asset rename - Purge of VM, PC, OCA, CertView records - Agent manifest download - Asset Inventory asset identification updates</td>
</tr>
</tbody>
</table>
### lastSeenAssetId

**Type:** Integer  

Use to get the list of assets having asset id greater than the specified last seen assetid.

**Note:** You can get a list of a maximum of 100 assets in one API call. If you have more than 100 assets, you need to make multiple API calls to get the list of all your assets.

See the following snippet from the Response:

**Response:**

```json
{
    "responseMessage": "Valid API Access",
    "count": 1,
    "responseCode": "SUCCESS",
    "lastSeenAssetId": 6920718,
    "hasMore": 1,
}
```

The snippet shows:
- lastSeenAssetId: 6920718  
  It indicates the asset id of the last seen asset.
- hasMore: 1  
  It indicates that there are more assets.

If you want to get a list of your all assets, refer to the following example:

**Example:**

https://gateway.qg1.apps.qualys.com/rest/2.0/search/am/asset?lastSeenAssetId=6920718

**Consider the scenario, wherein you have 1000 assets.**

In the first API call, you get a list of 100 assets. The asset id of the 100th asset is shown in the lastSeenAssetId parameter. In the 2nd API call, you must enter the asset id of the 100th asset in the lastSeenAssetId parameter to get the list of the next 100 assets. Similarly, in every subsequent API call, you must enter the asset id of the 100th asset in the lastSeenAssetId parameter. Thus, you get the list of your total assets by running 10 API calls.

### pageSize

**Type:** Integer  

The number of records per page to be included in the response. If pageSize is not specified in the request, 100 records will be fetched by default. The maximum value supported for pageSize is 300.
filter (String) Filter the events list by providing a filter in json and xml format. Make sure your filter criteria is provided in xml/json format in the request body. If you don’t provide filter parameter, it will show details of all the assets. For more information on supported operators, refer Supported Operators.

For example (json) -

```json
{
   "filters": [
      {
         "field": "software.product",
         "operator": "CONTAINS",
         "value": "Python"
      }
   ]
}
```

For example (xml) -

```xml
<FilterRequest>
   <filters>
      <Criteria field="software.product" operator="CONTAINS">Python</Criteria>
   </filters>
</FilterRequest>
```

softwareType (String) (Optional) Specify the software type to get the host details for specified software type. Available values for softwareType parameter are Application, Unknown, and Others.

Authorization (String) (Required) Authorization token to authenticate to the Qualys Cloud Platform. Prepend token with "Bearer" and one space. For example - Bearer authToken
Sample - Get details of all asset

Request (without filter):


Request (with filter - xml):


Here, filter.xml file is the request in xml format.

Request (with filter - json):


Here, filter.json file is the request in json format.

Sample Request body in xml format (filter.xml)

<FilterRequest>
  <filters>
    <Criteria field="operatingSystem.category1" operator="EQUALS">Mac</Criteria>
  </filters>
</FilterRequest>

Sample Request body in json format (filter.json)

{
  "filters": [
    {
      "field": "operatingSystem.category1",
      "operator": "EQUALS",
      "value": "Mac"
    }
  ]
}

Response:

{
  "responseMessage": "Valid API Access",
}
"count": 1,
"responseCode": "SUCCESS",
"lastSeenAssetId": 6920718,
"hasMore": 1,
"assetListData": {
    "asset": [
        {
            "assetId": 6920718,
            "assetUUID": "50d20290-c66a-42e7-8c0a-ba6e92b6324c",
            "hostId": 1437386,
            "lastModifiedDate": "2021-04-06T10:02:33.000Z",
            "agentId": null,
            "createdDate": "2020-11-25T12:49:25.000Z",
            "sensorLastUpdatedDate": "2021-04-06T10:02:33.000Z",
            "assetType": "HOST",
            "address": "10.115.110.95",
            "dnsName": "localhost.localdomain",
            "assetName": "localhost.localdomain",
            "netbiosName": null,
            "timeZone": "IST",
            "biosDescription": null,
            "lastBoot": null,
            "totalMemory": 5806,
            "cpuCount": null,
            "lastLoggedOnUser": "root",
            "hwUUID": "422a2b16-4c8b-588a-a20c-c1851ad7e376",
            "biosSerialNumber": "VMware-42 2a 2b 16 4c 8b 58 8a-a2 0c c1 85 1a d7 e3 76",
            "biosAssetTag": "No Asset Tag",
            "isContainerHost": false,
            "operatingSystem": {
                "osName": "The CentOS Project CentOS 7 (1810)",
                "fullName": "The CentOS Project CentOS 7 (1810)",
                "category": "Linux / Server",
                "category1": "Linux",
                "category2": "Server",
                "productName": "CentOS",
                "publisher": "The CentOS Project",
                "edition": null,
                "marketVersion": "7",
                "version": "1810",
                "update": null,
                "architecture": null,
            }
        }
    ]
}
"lifecycle": { 
  "gaDate": "2018-12-03T00:00:00.000Z",
  "eolDate": "2020-12-31T00:00:00.000Z",
  "eosDate": "2024-06-30T00:00:00.000Z",
  "stage": "EOL",
  "lifeCycleConfidence": "Exact",
  "eolSupportStage": "Full updates",
  "eosSupportStage": "Maintenance Updates"
},
"taxonomy": { 
  "id": null,
  "name": "Linux / Server",
  "category1": "Linux",
  "category2": "Server"
},
"productFamily": null,
"installDate": null},
"hardware": { 
  "fullName": "VMware VMware Virtual Platform VMware Virtual Platform",
  "category": "Virtualized / Virtual Machine",
  "category1": "Virtualized",
  "category2": "Virtual Machine",
  "manufacturer": "VMware",
  "productName": "VMware Virtual Platform",
  "model": "VMware Virtual Platform",
  "lifecycle": { 
    "introDate": null,
    "gaDate": null,
    "eosDate": null,
    "obsoleteDate": null,
    "stage": "Unknown",
    "lifeCycleConfidence": " "
  },
  "taxonomy": { 
    "id": null,
    "name": "Virtualized / Virtual Machine",
    "category1": "Virtualized",
    "category2": "Virtual Machine"
  },
  "productUrl": "https://www.linuxjournal.com/article/3458,,",
  "productFamily": null
"userAccountListData": null,
"openPortListData": {
  "openPort": [
    {
      "port": 709,
      "description": "",
      "protocol": "UDP",
      "detectedService": "portmap/rpcbind",
      "firstFound": "2020-11-25T12:46:42.000Z",
      "lastUpdated": "2020-11-25T12:46:42.000Z"
    },
    {
      "port": 50000,
      "description": "",
      "protocol": "TCP",
      "detectedService": "IBM_DB2_Universal_Database",
      "firstFound": "2020-11-25T12:46:42.000Z",
      "lastUpdated": "2020-11-25T12:46:42.000Z"
    },
    {
      "port": 6000,
      "description": "",
      "protocol": "TCP",
      "detectedService": "x11",
      "firstFound": "2020-11-25T12:46:42.000Z",
      "lastUpdated": "2020-11-25T12:46:42.000Z"
    },
    {
      "port": 22,
      "description": "",
      "protocol": "TCP",
      "detectedService": "ssh",
      "firstFound": "2020-11-25T12:46:41.000Z",
      "lastUpdated": "2020-11-25T12:46:41.000Z"
    }
  ]
}
"port": 3389,
"description": "",
"protocol": "TCP",
"detectedService": null,
"firstFound": "2020-11-
25T12:46:42.0002",
"lastUpdated": "2020-11-
25T12:46:42.0002"
},
{
"port": 111,
"description": "",
"protocol": "UDP",
"detectedService": "rpc_udp",
"firstFound": "2020-11-
25T12:46:42.0002",
"lastUpdated": "2020-11-
25T12:46:42.0002"
},
{
"port": 111,
"description": "",
"protocol": "TCP",
"detectedService": "rpc",
"firstFound": "2020-11-
25T12:46:41.0002",
"lastUpdated": "2020-11-
25T12:46:41.0002"
}

"volumeListData": {
  "volume": [
    {
      "name": "tmpfs",
      "free": 2737078272,
      "size": 3043934208
    },
    {
      "name": "/dev/mapper/centos-home",
      "free": 18629619712,
      "size": 18700304384
    },
    {
      "name": "devtmpfs",
      "free": 3026444288,
      "size": 3026444288
    }
  ]
}

Assets Host Data APIs
Get Host Details of All Assets
"size": 3026444288
},
{
"name": "/dev/mapper/centos-root",
"free": 19672580096,
"size": 38304645120
},
{
"name": "/dev/sda1",
"free": 876040192,
"size": 1063256064
}
",
"networkInterfaceListData": {
"networkInterface": [
{
"hostname": "localhost.localdomain",
"addressIPv4": "192.168.122.1",
"addressIPv6": null,
"macAddress": "52:54:00:77:e1:71",
"interfaceName": "virbr0",
"dnsAddress": null,
"gatewayAddress": "",
"manufacturer": null,
"macVendorIntroDate": null,
"addresses": null
},
{
"hostname": "localhost.localdomain",
"addressIPv4": "10.115.110.95",
"addressIPv6": "fe80:0:0:0:250:56ff:feaa:e2da",
"macAddress": "00:50:56:aa:e2:da",
"interfaceName": "ens192",
"dnsAddress": null,
"gatewayAddress": "",
"manufacturer": "VMware",
"macVendorIntroDate": 946944000000,
"addresses": null
}
]
",
"softwareListData": {
"software": [

}
Assets Host Data APIs
Get Host Details of All Assets

(Python-2.0)"

},
"authorization": "Authorized"
},

{
"id": 9136542396418607016,
"fullName": "OpenBSD OpenSSH Server 7.4p1",
"softwareType": "Application",
"isIgnored": false,
"ignoredReason": null,
"category": "Networking / Access Software",
"category1": "Networking",
"category2": "Access Software",
"productName": "OpenSSH",
"component": "Server",
"publisher": "OpenBSD",
"edition": null,
"marketVersion": "7",
"version": "7.4",
"update": "7.4p1",
"architecture": null,
"installDate": "2020-03-05T14:23:53.000Z",
"installPath": null,
"lastUpdated": "2020-11-25T12:46:53.000Z",
"lastUseDate": null,
"language": null,
"formerlyKnownAs": "OpenBSD Secure Shell",
"isPackage": true,
"isPackageComponent": false,
"packageName": null,
"lifecycle": {
"gaDate": "2016-12-19T00:00:00.000Z",
"eolDate": null,
"eosDate": null,
"stage": "EOL",
"lifeCycleConfidence": "Calculated",
"eolSupportStage": "",
"eosSupportStage": ""
}
}
"supportStageDesc": null,
"license": {
  "category": "Open Source",
  "subcategory": "BSD 2-Clause License (FreeBSD/Simplified)"
},
"authorization": "Authorized"
],
"provider": null,
"cloudProvider": null,
"agent": null,
"sensor": {
  "activatedForModules": [ "VM"
],
  "pendingActivationForModules": [],
  "lastVMScan": 1606306572000,
  "lastComplianceScan": 0,
  "lastFullScan": 1606306572000
},
"container": null,
"inventory": {
  "source": "IP",
  "created": 1606308565000,
  "lastUpdated": 1617703353000
},
"activity": null,
"tagList": {
  "tag": [
    {
      "tagId": 14151022,
      "tagName": "static split",
      "foregroundColor": 0,
      "backgroundColor": -65536,
      "businessImpact": null,
      "criticalityScore": 3
    }
  ]
},
"serviceList": null,
"lastLocation": null,
"criticality": {
  "score": 2,
  "isDefault": true,
"lastUpdated": "2021-06-30T09:43:27.000Z"
},
"processor": null
]
}
}
}
Import Business Information Metadata

API affected

- rest/2.0/update/am/asset/business/metadata
- rest/2.0/upsert/am/businessapp/metadata

New or Updated APIs

New

With this release, we’ve added support to import asset business metadata and business app metadata using v2 APIs. This support is available for CSAM Paid and Trial subscriptions only. You’ll be able to import maximum 250 records in the single request.

Sample - Import Business App Metadata

These parameters are mandatory in the request body to import business app metadata: businessAppId, name, created, and lastUpdated

Request:


Request body:

```json
{
  "data": [
    {
      "businessAppId": "2fc86c650a0a0bb4003698b5331640df",
      "name": "Banking Service",
      "businessCriticality": "1 - Most Critical",
      "status": "Installed",
      "environment": "Production",
      "usedFor": "Production",
      "created": 1620643264000,
      "lastUpdated": 1620653309000,
      "operationalStatus": "Installed",
      "ownedBy": "Joey Bolick",
      "managedBy": "Byron Fortuna",
      "supportedBy": "John Doe",
      "supportGroup": "IT Operations"
    },
    {
      "businessAppId": "5678f28f933a31003b4bb095e57ff88",
      "name": "Customer Support Portal",
      "businessCriticality": "3 - Low",
      "status": "Installed",
      "environment": "Development",
      "usedFor": "Development"
    }
  ]
}
```
"created": 1620643264000,
"lastUpdated": 1620653309000,
"operationalStatus": "Installed",
"ownedBy": "Joey Bolick",
"managedBy": "Byron Fortuna",
"supportedBy": "John Doe",
"supportGroup": "Application Security"}
]
}
}
}
}

Response:
{
  "requestId": "8e9b3fd5-bb89-4666-a472-4bc575835a2",
  "responseMessage": "Business app metadata imported successfully",
  "responseCode": "SUCCESS",
  "failedIds": null
}

**Note:** If the business app data for the associated business app id of the asset is not present then the association of that business app with the asset will not happen but the rest of all the data will get updated.

**Sample - Import Asset Business Metadata**

**API request:**


**Request Body**

{  
  "data": [  
    {  
      "qualysAssetId": "6420613",
      "metadata": {  
        "operationalStatus": "Operational",
        "environment": "Production",
        "company": "ACME US",
        "department": "IT Operations",
        "ownedBy": "Joey Bolick",
        "managedBy": "Byron Fortuna",
        "supportedBy": "John Doe",
        "supportGroup": "IT Operations",
        "businessAppIds": ["2fc86c650a00bb4003698b5331640df"],
        "assignedLocation": {  
          "name": "401 Biscayne St, Miami FL",
          "street": "401 Biscayne St, Miami FL",
          "city": "Miami",
          }  
      }  
    }  
  ]
}
"state": "FL",
"country": "USA"
}
}
,
{
"qualysAssetId": "6286688",
"metadata": {
"operationalStatus": "Repair",
"environment": "Development",
"company": "ACME Italy",
"department": "Customer Support",
"ownedBy": "Joey Bolick",
"managedBy": "Byron Fortuna",
"supportedBy": "John Doe",
"supportGroup": "Customer Support",
"businessAppIds": [
"27d415a8c0a8000b00ffe2ab0f82e8d2",
"5678f28f933a31003b4bb095e57ff8b88"
],
"assignedLocation": {
"name": "123 Plazuela Roma Italy",
"street": "123 Plazuela Roma Italy",
"city": "Roma",
"state": "Roma",
"country": "Italy"
}
}
}
]

Response:
{
"requestId": "9017b662-01c9-4e74-97c5-eae6d29f08ed",
"responseMessage": "Asset metadata imported successfully",
"responseCode": "SUCCESS",
"failedIds": null
}
Import Business Information Metadata

Use these API functions to import asset business metadata and business app metadata using v2 APIs. This support is available for CSAM Paid and Trial subscriptions only. You’ll be able to import maximum 250 records in the single request.

Permissions

- User must have the GAV/CSAM module and the "App API Enabled" option enabled for that role.

Note:

1) For updating business information metadata, you need to send new request with desired attributes to be changed along with all the attributes. If you don’t include an attribute in the request, the value of the attribute will be override with NULL value.

2) If you have changed business app metadata then you need to explicitly send a request to change the asset business metadata for those assets.

Import Business App Metadata

Import business app metadata as per input criteria in the request body.

rest/2.0/upsert/am/businessapp/metadata

[POST]

Input Parameters for Business Information Metadata

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Character Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>(Required to import business app metadata) Name of the business application</td>
<td>255</td>
</tr>
<tr>
<td>businessAppid</td>
<td>(Required to import business app metadata) Unique ID of the business application</td>
<td>32</td>
</tr>
<tr>
<td>operationalStatus</td>
<td>Operational status of the application</td>
<td>255</td>
</tr>
<tr>
<td>businessCriticality</td>
<td>How critical the application is to the business.</td>
<td>255</td>
</tr>
<tr>
<td>environment</td>
<td>Designates how this business app is used, e.g. Production, Staging, QA, etc</td>
<td>255</td>
</tr>
<tr>
<td>ownedBy</td>
<td>Person who owns the application from the business side.</td>
<td>255</td>
</tr>
<tr>
<td>managedBy</td>
<td>Person who owns the application from the IT side</td>
<td>255</td>
</tr>
</tbody>
</table>
Import Business Information Metadata

Import Business App Metadata

<table>
<thead>
<tr>
<th>supportedBy</th>
<th>User supporting the business application</th>
<th>255</th>
</tr>
</thead>
<tbody>
<tr>
<td>supportGroup</td>
<td>Group supporting the business application</td>
<td>255</td>
</tr>
<tr>
<td>created</td>
<td>(Required to import business app metadata) Business app created date</td>
<td>NA</td>
</tr>
<tr>
<td>lastUpdated</td>
<td>(Required to import business app metadata) Business app last updated date</td>
<td>NA</td>
</tr>
</tbody>
</table>

Request:


Request body:

```json
{
  "data": [
    {
      "businessAppId": "2fc86c650a0a0bb4003698b5331640df",
      "name": "Banking Service",
      "businessCriticality": "1 - Most Critical",
      "status": "Installed",
      "environment": "Production",
      "usedFor": "Production",
      "created": 1620643264000,
      "lastUpdated": 1620653309000,
      "operationalStatus": "Installed",
      "ownedBy": "Joey Bolick",
      "managedBy": "Byron Fortuna",
      "supportedBy": "John Doe",
      "supportGroup": "IT Operations"
    },
    {
      "businessAppId": "5678f28f933a31003b4bb095e57ff88",
      "name": "Customer Support Portal",
      "businessCriticality": "3 - Low",
      "status": "Installed",
      "environment": "Development",
      "usedFor": "Development",
      "created": 1620643264000,
      "lastUpdated": 1620653309000,
      "operationalStatus": "Installed"
    }
  ]
}
```
"ownedBy": "Joey Bolick",
"managedBy": "Byron Fortuna",
"supportedBy": "John Doe",
"supportGroup": "Application Security"
}
]

Response:
{
"requestId": "8e9b3fd5-bb89-4666-a472-4bc5758335a2",
"responseMessage": "Business app metadata imported successfully",
"responseCode": "SUCCESS",
"failedIds": null
}

**Note:** If the business app data for the associated business app id of the asset is not present then the association of that business app with the asset will not happen but the rest of all the data will get updated.

**Import Asset Business Metadata**

Import asset business metadata as per input criteria in the request body.

**rest/2.0/update/am/asset/business/metadata**

[POST]

**Input Parameters**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Character Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>qualysAssetId</td>
<td>(Required to import asset business metadata) Unique ID of the asset</td>
<td>NA</td>
</tr>
<tr>
<td>businessAppIds</td>
<td>Unique IDs of the business application</td>
<td>NA</td>
</tr>
<tr>
<td>operationalStatus</td>
<td>Operational status of the asset</td>
<td>128</td>
</tr>
<tr>
<td>environment</td>
<td>The environment this asset is connected to / runs on</td>
<td>128</td>
</tr>
<tr>
<td>ownedBy</td>
<td>Person who owns the asset from the business side.</td>
<td>255</td>
</tr>
<tr>
<td>managedBy</td>
<td>Person who owns the asset from the IT side</td>
<td>255</td>
</tr>
<tr>
<td>supportedBy</td>
<td>User supporting the asset</td>
<td>255</td>
</tr>
<tr>
<td>supportGroup</td>
<td>Group supporting the asset</td>
<td>255</td>
</tr>
<tr>
<td>company</td>
<td>The Company or Subsidiary</td>
<td>128</td>
</tr>
</tbody>
</table>
**Import Business Information Metadata**

**Import Asset Business Metadata**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>department</td>
<td>The departmental organizational structure</td>
<td>128</td>
</tr>
<tr>
<td>assignedLocation.name</td>
<td>The assigned location name, for example building name</td>
<td>255</td>
</tr>
<tr>
<td>assignedLocation.city</td>
<td>The assigned location's city</td>
<td>128</td>
</tr>
<tr>
<td>assignedLocation.state</td>
<td>The assigned location's state</td>
<td>128</td>
</tr>
<tr>
<td>assignedLocation.country</td>
<td>The assigned location's country</td>
<td>128</td>
</tr>
</tbody>
</table>

**API request:**

```bash
```

**Request Body**

```json
{
  "data": [
    {
      "qualysAssetId": "6420613",
      "metadata": {
        "operationalStatus": "Operational",
        "environment": "Production",
        "company": "ACME US",
        "department": "IT Operations",
        "ownedBy": "Joey Bolick",
        "managedBy": "Byron Fortuna",
        "supportedBy": "John Doe",
        "supportGroup": "IT Operations",
        "businessAppIds": ["2fc86c650a0a0bb4003698b5331640df"],
        "assignedLocation": {
          "name": "401 Biscayne St, Miami FL",
          "street": "401 Biscayne St, Miami FL",
          "city": "Miami",
          "state": "FL",
          "country": "USA"
        }
      }
    },
    {
      "qualysAssetId": "6286688",
      "metadata": {
        "operationalStatus": "Repair",
        "environment": "Development",
        "company": "ACME Italy",
        "department": "Customer Support",
        "ownedBy": "Joey Bolick",
        "managedBy": "Byron Fortuna",
        "supportedBy": "John Doe",
```

44
"supportGroup": "Customer Support",
"businessAppIds": [
  "27d415a8c0a8000b00ffe2ab0f828e8d2",
  "5678f28f933a31003b4bb095e57ff88"
],
"assignedLocation": {
  "name": "123 Plazuela Roma Italy",
  "street": "123 Plazuela Roma Italy",
  "city": "Roma",
  "state": "Roma",
  "country": "Italy"
}
}

Response:
{
  "requestId": "9017b662-01c9-4e74-97c5-eae6d29f08ed",
  "responseMessage": "Asset metadata imported successfully",
  "responseCode": "SUCCESS",
  "failedIds": null
}
Import Business Information Metadata
Import Asset Business Metadata
Appendix

This appendix describes the types of error messages returned from GAV/CSAM API requests, list of operators with supported attributes.

Error Messages

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>400</td>
<td>The request could not be understood by the server due to malformed syntax. This error also occurs if you provide wrong (or unsupported) operator in the request.</td>
</tr>
</tbody>
</table>
| 403 Forbidden | This response code is returned for the following scenarios:  
- If the Asset Inventory License is in “Pending Activation”.  
- If “App API Enabled” option is not checked.  
- If “App API Enabled” option is checked, but the license expiration date (for Trial/Full customers) has elapsed.  
- If the customer’s license subscription cannot be validated. |
| 404 Not found | The server has not found anything matching the Request |
| 416 Requested Range Not Satisfiable | Please provide a Page Size value less than the max page size limit set. |
| 500 Failure | The server encountered an unexpected condition which prevented it from fulfilling the request |

Supported Operators

This section of the appendix lists supported operators for tokens.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUMERIC_OPERATORS</td>
<td>EQUALS, IN, NOT_EQUALS, GREATER, LESSER, GREATER_THAN_EQUAL, and LESS_THAN_EQUAL</td>
</tr>
<tr>
<td>NUMERIC_AND_NOT_EQUAL_OPERATORS</td>
<td>EQUALS, IN, GREATER, LESSER, GREATER_THAN_EQUAL, LESS_THAN_EQUAL</td>
</tr>
<tr>
<td>STRING_OPERATORS</td>
<td>CONTAINS, IN, EQUALS, and NOT_EQUALS</td>
</tr>
<tr>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
<td>CONTAINS, IN, and EQUALS</td>
</tr>
<tr>
<td>Operator</td>
<td>Values</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>DATE_OPERATORS</td>
<td>EQUALS, NOT_EQUALS, GREATER, LESSER, GREATER_THAN_EQUAL, and LESS_THAN_EQUAL</td>
</tr>
<tr>
<td>BOOLEAN_OPERATORS</td>
<td>EQUALS</td>
</tr>
<tr>
<td>ENUM_OPERATORS</td>
<td>EQUALS, NOT_EQUALS, and IN</td>
</tr>
<tr>
<td>UUID_OPERATORS</td>
<td>EQUALS and IN</td>
</tr>
<tr>
<td>IP_OPERATORS</td>
<td>EQUALS and IN</td>
</tr>
</tbody>
</table>

Following table lists different attributes with supported operators:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Asset Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>asset.assetID</td>
<td>NUMERIC_OPERATORS</td>
</tr>
<tr>
<td>asset.name</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.created</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>asset.lastUpdated</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>asset.type</td>
<td>ENUM_OPERATORS</td>
</tr>
<tr>
<td>asset.lastLoggedOnUser</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.totalMemory</td>
<td>NUMERIC_OPERATORS</td>
</tr>
<tr>
<td>asset.timezone</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.trackingMethod</td>
<td>ENUM_OPERATORS</td>
</tr>
<tr>
<td>asset.domainRole</td>
<td>ENUM_OPERATORS</td>
</tr>
<tr>
<td>asset.riskScore</td>
<td>ENUM_OPERATORS</td>
</tr>
<tr>
<td>asset.lastBoot</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>asset.netbiosName</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.hostID</td>
<td>NUMERIC_OPERATORS</td>
</tr>
<tr>
<td>asset.isContainerHost</td>
<td>BOOLEAN_OPERATORS</td>
</tr>
<tr>
<td>asset.biosAssetTag</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.biosDescription</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.biosHardwareUUID</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.biosSerialNumber</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.agentID</td>
<td>UUID_OPERATORS</td>
</tr>
<tr>
<td>asset.criticalityScore</td>
<td>NUMERIC_OPERATORS</td>
</tr>
<tr>
<td>accounts.username</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>provider</td>
<td>ENUM_OPERATORS</td>
</tr>
<tr>
<td>isDockerHost</td>
<td>BOOLEAN_OPERATORS</td>
</tr>
</tbody>
</table>
### Supported Operators

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Inventory Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>inventory.source</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>inventory.created</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>inventory.lastUpdated</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td><strong>Processor Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>processors</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>processors.speed</td>
<td>NUMERIC_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td><strong>Container Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>container.noOfContainers</td>
<td>NUMERIC_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>container.noOfImages</td>
<td>NUMERIC_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>container.version</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td><strong>Interface Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>interfaces.hostname</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>interfaces.interfaceName</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>interfaces.macAddress</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>interfaces.manufacturer</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>interfaces.address</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>interfaces.dnsAddress</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>interfaces.gatewayAddress</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td><strong>Open Ports Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>openPorts.description</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>openPorts.detectedService</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>openPorts.protocol</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>openPorts.port</td>
<td>NUMERIC_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>openPorts.firstFound</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>openPorts.lastUpdated</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td><strong>Services Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>services.description</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>services.name</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>services.status</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td><strong>Sensors Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>sensors.lastComplianceScan</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>sensors.lastFullScan</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>sensors.lastVmScan</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>Attribute</td>
<td>Operator</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td><strong>Tag Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>tags.name</td>
<td>EQUALS, IN, CONTAINS</td>
</tr>
<tr>
<td>tags.businessImpact</td>
<td>EQUALS, IN, CONTAINS</td>
</tr>
<tr>
<td><strong>Volume Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>volumes.free</td>
<td>NUMERIC_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>volumes.size</td>
<td>NUMERIC_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>volumes.name</td>
<td>STRING_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td><strong>Agent Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>agent.version</td>
<td>NUMERIC_AND_NOT_EQUAL_OPERATORS</td>
</tr>
<tr>
<td>agent.connectedFrom</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>agent.errorStatus</td>
<td>BOOLEAN_OPERATORS</td>
</tr>
<tr>
<td>agent.lastActivity</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>agent.lastCheckedIn</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td><strong>Hardware Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>hardware</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>hardware.category</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>hardware.category1</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>hardware.category2</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>hardware.manufacturer</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>hardware.model</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>hardware.product</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>hardware.lifecycle.stage</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>hardware.lifecycle.eos</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>hardware.lifecycle.ga</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>hardware.lifecycle.intro</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>hardware.lifecycle.obs</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td><strong>Software Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>software.architecture</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.category</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.category1</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.category2</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.component</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.edition</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.marketVersion</td>
<td>STRING_OPERATORS</td>
</tr>
</tbody>
</table>
### Supported Operators

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>software.name</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.product</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.publisher</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.supportStage</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.version</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.update</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.isFilePackage</td>
<td>BOOLEAN_OPERATORS</td>
</tr>
<tr>
<td>software.isFilePackageComponent</td>
<td>BOOLEAN_OPERATORS</td>
</tr>
<tr>
<td>software.license.category</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.license.subcategory</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.lifecycle.stage</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>software.installDate</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>software.lastUseDate</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>software.lastUpdated</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>software.lifecycle.eol</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>software.lifecycle.eos</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>software.lifecycle.ga</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>software.authorization</td>
<td>EQUALS, NOT_EQUALS, and IN</td>
</tr>
</tbody>
</table>

### Operating System Attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>operatingSystem</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.category</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.category1</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.category2</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.architecture</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.component</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.edition</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.marketVersion</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.name</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.publisher</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.version</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.update</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.lifecycle.stage</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.installDate</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.lifecycle.eol</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>Attribute</td>
<td>Operator</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>operatingSystem.lifecycle.eos</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>operatingSystem.lifecycle.ga</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td><strong>AWS Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>aws.ec2.availabilityZone</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.instanceType</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.publicDNS</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.privateDNS</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.accountId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.imageId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.instanceId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.instanceState</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.region.code</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.subnetId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.vpcId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.hostname</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.privateIpAddress</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.publicIpAddress</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>aws.tags.key</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.tags.value</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.spotInstance</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.launchDate</td>
<td>DATE_OPERATORS</td>
</tr>
<tr>
<td>aws.ec2.hasAgent</td>
<td>BOOLEAN_OPERATORS</td>
</tr>
<tr>
<td><strong>Azure Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>azure.vm.imageOffer</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.imagePublisher</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.imageVersion</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.name</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.size</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.vmlId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.resourceGroupName</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.virtualNetwork</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.state</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.subnet</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.subscriptionId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>Attribute</td>
<td>Operator</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>azure.vm.location</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.platform</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.macAddress</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.tags.value</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.tags.name</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.privateIpAddress</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.publicIpAddress</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>azure.vm.hasAgent</td>
<td>BOOLEAN_OPERATORS</td>
</tr>
<tr>
<td><strong>GCP Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>gcp.compute.hostname</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>gcp.compute.instanceId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>gcp.compute.machineType</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>gcp.compute.network</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>gcp.compute.projectId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>gcp.compute.projectNumber</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>gcp.compute.macAddress</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>gcp.compute.state</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>gcp.compute.zone</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>gcp.compute.privateIpAddress</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>gcp.compute.publicIpAddress</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td><strong>OCI Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>oci.compute.availabilityDomain</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.canonicalRegionName</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.compartmentId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.compartmentName</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.displayName</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.faultDomain</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.hostName</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.imageId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.ocid</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.region</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.shape</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.state</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.tenantId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>Attribute</td>
<td>Operator</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>oci.compute.tenantName</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.compute.timeCreated</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.tags</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.tags.key</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.tags.namespace</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.tags.type</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.tags.value</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.macAddr</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.nicIndex</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.privateIp</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.publicIp</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.subnetCidrBlock</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.subnetId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.subnetName</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.vcnId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.vcnName</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.virtualRouterIp</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.vlanTag</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>oci.vnic.vnicId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>IBM Cloud Attributes</td>
<td></td>
</tr>
<tr>
<td>ibm.tags</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>ibm.tags.name</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>ibm.tags.value</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>ibm.virtualServer.datacenterId</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>ibm.virtualServer.deviceName</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>ibm.virtualServer.domain</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>ibm.virtualServer.id</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>ibm.virtualServer.location</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>ibm.virtualServer.privateIpAddress</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>ibm.virtualServer.privateVlan</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>ibm.virtualServer.publicIpAddress</td>
<td>IP_OPERATORS</td>
</tr>
<tr>
<td>ibm.virtualServer.publicVlan</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>ibm.virtualServer.state</td>
<td>STRING_OPERATORS</td>
</tr>
</tbody>
</table>

Geo IP Attributes
### Supported Operators

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>asset.lastLocation</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.lastLocation.city</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.lastLocation.country</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.lastLocation.continent</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.lastLocation.postal</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.lastLocation.state</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td><strong>Business Information Attributes</strong></td>
<td></td>
</tr>
<tr>
<td>asset.org.company</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.org.department</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.ownedBy</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.managedBy</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.supportedBy</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.supportGroup</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.environment</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.operationalStatus</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.assignedLocation.name</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.assignedLocation.city</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.assignedLocation.state</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>asset.assignedLocation.country</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>businessApp.name</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>businessApp.id</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>businessApp.businessCriticality</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>businessApp.ownedBy</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>businessApp.supportGroup</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>businessApp.operationalStatus</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>businessApp.environment</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>businessApp.managedBy</td>
<td>STRING_OPERATORS</td>
</tr>
<tr>
<td>businessApp.supportedBy</td>
<td>STRING_OPERATORS</td>
</tr>
</tbody>
</table>

**Note:** Following tokens are available only for CSAM License Subscriber:

hardware.lifecycle.stage, hardware.lifecycle.eos, hardware.lifecycle.ga, hardware.lifecycle.intro, hardware.lifecycle.obs, software.authorization, software.license.category, software.license.subcategory, software.lifecycle.eol, software.lifecycle.eos, software.lifecycle.ga, software.lifecycle.stage, software.isPackage, software.isPackageComponent,
operatingSystem.lifecycle.eol, operatingSystem.lifecycle.eos, operatingSystem.lifecycle.ga, and operatingSystem.lifecycle.stage

Following are some example to understand the different supported operators by comparing QQL(UI) tokens:

**Example 1 - hardware.category1:Computers**

*Request Body in XML:*

```xml
<FilterRequest>
  <filters>
    <Criteria field="hardware.category1" operator="CONTAINS">
      <value>Computers</value>
    </Criteria>
  </filters>
</FilterRequest>
```

*OR Request Body in Json:*

```json
{
  "filters": [
    {
      "field": "hardware.category1",
      "operator": "CONTAINS",
      "value": "Computers"
    }
  ]
}
```

**Example 2 - hardware.manufacturer:Apple OR hardware.manufacturer:HPE**

*Request Body in XML:*

```xml
<FilterRequest>
  <filters>
    <Criteria field="hardware.manufacturer" operator="IN">
      <value>Apple,HPE</value>
    </Criteria>
  </filters>
</FilterRequest>
```

**Example 3 - software:(product:Python and update:2.7.5)**

*Request Body in XML:*

```xml
<FilterRequest>
</FilterRequest>
```
<filters>
    <Criteria field="software.product" operator="CONTAINS">
        <value>Python</value>
    </Criteria>
    <Criteria field="software.update" operator="CONTAINS">
        <value>2.7.5</value>
    </Criteria>
</filters>

Example 4 - operatingSystem.category1: `Mac` and hardware.category: Notebook

Request Body in XML:

    <FilterRequest>
        <filters>
            <Criteria field="operatingSystem.category1" operator="EQUALS">
                <value>Mac</value>
            </Criteria>
            <Criteria field="hardware.category" operator="EQUALS">
                <value>Notebook</value>
            </Criteria>
        </filters>
    </FilterRequest>

Example 5 - operatingSystem.category1: `Mac` or hardware.category: Notebook

Request Body in XML:

    <FilterRequest>
        <filters>
            <Criteria field="operatingSystem.category1" operator="EQUALS">
                <value>Mac</value>
            </Criteria>
            <Criteria field="hardware.category" operator="EQUALS">
                <value>Notebook</value>
            </Criteria>
            <operation>OR</operation>
        </filters>
    </FilterRequest>
Request Body in Json:

```json
{
  "filters": [
  {
    "field": "operatingSystem.category1",
    "operator": "EQUALS",
    "value": "Mac"
  },
  {
    "field": "hardware.category",
    "operator": "EQUALS",
    "value": "Notebook"
  }
  ],
  "operation": "OR"
}
```

**Example 6 - operatingSystem.category1:`Mac` and hardware.category:Notebook**

Request Body in XML:

```xml
<FilterRequest>
  <filters>
    <Criteria field="operatingSystem.category1" operator="EQUALS">
      <value>Mac</value>
    </Criteria>
    <Criteria field="hardware.category" operator="EQUALS">
      <value>Notebook</value>
    </Criteria>
  </filters>
  <operation>AND</operation>
</FilterRequest>
```