Qualys API Release Notes

Version 8.12

Qualys 8.12 includes improvements to the Qualys API, giving you more ways to integrate your programs and API calls with Qualys Vulnerability Management (VM) and Qualys Policy Compliance (PC). Looking for our API user guides? Just log in to your Qualys account and go to Help > Resources.

What's New

URL to the Qualys API Server

Qualys maintains multiple Qualys platforms. The Qualys API server URL that you should use for API requests depends on the platform where your account is located.

Account Location	API Server URL
Qualys US Platform 1	https://qualysapi.qualys.com
Qualys US Platform 2	https://qualysapi.qg2.apps.qualys.com
Qualys US Platform 3	https://qualysapi.qg3.apps.qualys.com
Qualys EU Platform 1	https://qualysapi.qualys.eu
Qualys EU Platform 2	https://qualysapi.qg2.apps.qualys.eu
Qualys India Platform 1	https://qualysapi.qg1.apps.qualys.in
Qualys Private Cloud Platform	https://qualysapi.<customer_base_url>

The Qualys API documentation and sample code use the API server URL for the Qualys US Platform 1. If your account is located on another platform, please replace this URL with the appropriate server URL for your account.

Enhanced Asset Group API v2

API affected	/api/2.0/fo/asset/group/
New or Updated API	Updated
DTD or XSD changes	Yes

The Asset Group API v2 (/api/2.0/fo/asset/group/) has the following updates:
- Download the API results in a CSV format
- Fetch comments for an asset group

Input Parameters

New input parameters are described below.

Parameter	Description
output_format={csv|xml}	(Required) The requested output format: CSV or XML.

Additionally, the existing parameter show_attributes={value} now shows comments added to an asset group. Specifying ALL will list the comments along with all other attributes. Specify show_attributes=COMMENTS as part of a restricted group of attributes.

API Request and Sample Output

API request for XML:

```
curl -u "USERNAME:PASSWD" -H "X-Requested-With: Curl" -X "POST" -d
"action=list&output_format=xml&show_attributes=ALL&ids=19200"
"https://qualysapi.qualys.com/api/2.0/fo/asset/group/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE ASSET_GROUP_LIST_OUTPUT SYSTEM
"http://qualysapi.qualys.com/api/2.0/fo/asset/group/asset_group_list_outp
ut.dtd">
<ASSET_GROUP_LIST_OUTPUT>
    <RESPONSE>
        <DATETIME>2017-12-20T08:58:33Z</DATETIME>
        <ASSET_GROUP_LIST>
            <ASSET_GROUP>
                <ID>19200</ID>
                <TITLE>
                    <![CDATA[Assets2]]>
                </TITLE>
                <OWNER_USER_ID>17061</OWNER_USER_ID>
```

```
                    <NETWORK_ID>0</NETWORK_ID>
                    <LAST_UPDATE>2017-12-20T08:24:05Z</LAST_UPDATE>
                    <BUSINESS_IMPACT>High</BUSINESS_IMPACT>
                    <CVSS_ENVIRO_CDP>Not Defined</CVSS_ENVIRO_CDP>
                    <CVSS_ENVIRO_TD>Not Defined</CVSS_ENVIRO_TD>
                    <CVSS_ENVIRO_CR>Not Defined</CVSS_ENVIRO_CR>
                    <CVSS_ENVIRO_IR>Not Defined</CVSS_ENVIRO_IR>
                    <CVSS_ENVIRO_AR>Not Defined</CVSS_ENVIRO_AR>
                    <DEFAULT_APPLIANCE_ID>15094</DEFAULT_APPLIANCE_ID>
                    <APPLIANCE_IDS>15094</APPLIANCE_IDS>
                    <IP_SET>
                        <IP_RANGE>10.0.0.1-10.0.0.10</IP_RANGE>
                        <IP_RANGE>10.193.212.0-10.193.215.255</IP_RANGE>
                    </IP_SET>
                    <DNS_LIST>
                        <DNS>google.com</DNS>
                    </DNS_LIST>
                    <NETBIOS_LIST>
                        <NETBIOS>NETBIOS.COM</NETBIOS>
                    </NETBIOS_LIST>
                    <ASSIGNED_USER_IDS>18061</ASSIGNED_USER_IDS>
                    <COMMENTS>
                      <![CDATA[Asset group comments]]>
                    </COMMENTS>
                </ASSET_GROUP>
            </ASSET_GROUP_LIST>
        </RESPONSE>
    </ASSET_GROUP_LIST_OUTPUT>
```

API request for CSV:

```
curl -u "USERNAME:PASSWD" -H "X-Requested-With: Curl" -X "POST" -d
"action=list&output_format=csv&show_attributes=ALL&ids=19200"
"https://qualysapi.qualys.com/api/2.0/fo/asset/group/"
```

CSV output:

```
----BEGIN_RESPONSE_BODY_CSV
"ID","TITLE","IP_SET","DOMAIN_LIST","APPLIANCE_LIST","BUSINESS_IMPACT","O
WNER_USER_ID","OWNER_USER_NAME","LAST_UPDATE","NETWORK_IDS","NETBIOS_LIST
","DNS_LIST","HOST_IDS","EC2_ID_LIST","ASSIGNED_USER_IDS","ASSIGNED_UNIT_
IDS","CVSS_ENVIRO_CDP","CVSS_ENVIRO_TD","CVSS_ENVIRO_CR","CVSS_ENVIRO_IR"
,"CVSS_ENVIRO_AR","OWNER_UNIT_ID","COMMENTS"
"19200","Assets2","10.0.0.1-10.0.0.10,10.193.212.0-
10.193.215.255","0","1","High","17061","Abhishek Sawant(Manager)","2017-
12-20T08:24:05Z","0","NETBIOS.COM","google.com","0","0","18061","0","Not
Defined","Not Defined","Not Defined","Not Defined","Not
Defined","0","Asset group comments"
----END_RESPONSE_BODY_CSV
```

DTD Update

We updated the Asset Group List Output DTD (asset_group_list_output.dtd) to include the Comments element. New elements are in bold.

DTD: <base_url>/api/2.0/fo/asset/group/asset_group_list_output.dtd

```
<!ELEMENT ASSET_GROUP_LIST_OUTPUT (REQUEST?,RESPONSE)>

<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
POST_DATA?)>
<!ELEMENT DATETIME (#PCDATA)>
<!ELEMENT USER_LOGIN (#PCDATA)>
<!ELEMENT RESOURCE (#PCDATA)>
<!ELEMENT PARAM_LIST (PARAM+)>
<!ELEMENT PARAM (KEY, VALUE)>
...
<!ELEMENT RESPONSE (DATETIME, (ASSET_GROUP_LIST|ID_SET)?, WARNING?)>
<!ELEMENT ASSET_GROUP_LIST (ASSET_GROUP+)>
<!ELEMENT ASSET_GROUP (ID, TITLE?,
    OWNER_USER_ID?, OWNER_UNIT_ID?, (NETWORK_ID|NETWORK_IDS)?,
LAST_UPDATE?, BUSINESS_IMPACT?,
    CVSS_ENVIRO_CDP?, CVSS_ENVIRO_TD?, CVSS_ENVIRO_CR?, CVSS_ENVIRO_IR?,
CVSS_ENVIRO_AR?,
    DEFAULT_APPLIANCE_ID?, APPLIANCE_IDS?,
    IP_SET?, DOMAIN_LIST?, DNS_LIST?, NETBIOS_LIST?,
    EC2_ID_LIST?, HOST_IDS?,
    ASSIGNED_USER_IDS?, ASSIGNED_UNIT_IDS?, COMMENTS?
)>
...
<!-- UNIT_IDS -->
<!ELEMENT ASSIGNED_UNIT_IDS (#PCDATA)>

<!-- COMMENTS -->
<!ELEMENT COMMENTS (#PCDATA)>

<!-- WARNING -->
<!ELEMENT WARNING (CODE?, TEXT, URL?)>
<!ELEMENT CODE (#PCDATA)>
<!ELEMENT TEXT (#PCDATA)>
<!ELEMENT URL (#PCDATA)>

<!-- EOF -->
```

Asset Group List Output - DTD Change

API affected	/api/2.0/fo/asset/group/ with action=list
New or Updated API	Neither (DTD change only)
DTD or XSD changes	Yes

The Asset Group List Output DTD is used when you list the asset groups in your account. We've made several changes to this DTD. The updated DTD (asset_group_list_output.dtd) is shown below with changed and new elements in bold.

DTD update:

```
<!-- QUALYS ASSET_GROUP_LIST_OUTPUT DTD -->
<!-- $Revision$ -->
<!ELEMENT ASSET_GROUP_LIST_OUTPUT (REQUEST?,RESPONSE)>

<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
POST_DATA?)>
<!ELEMENT DATETIME (#PCDATA)>
<!ELEMENT USER_LOGIN (#PCDATA)>
<!ELEMENT RESOURCE (#PCDATA)>
<!ELEMENT PARAM_LIST (PARAM+)>
<!ELEMENT PARAM (KEY, VALUE)>
<!ELEMENT KEY (#PCDATA)>
<!ELEMENT VALUE (#PCDATA)>
<!-- if returned, POST_DATA will be urlencoded -->
<!ELEMENT POST_DATA (#PCDATA)>

<!ELEMENT RESPONSE (DATETIME, (ASSET_GROUP_LIST|ID_SET)?, WARNING?)>
<!ELEMENT ASSET_GROUP_LIST (ASSET_GROUP+)>
<!ELEMENT ID_SET (ID|ID_RANGE)+>
<!ELEMENT ID_RANGE (#PCDATA)>
<!ELEMENT ASSET_GROUP (ID, TITLE?,
OWNER_USER_ID?, OWNER_UNIT_ID?, (NETWORK_ID|NETWORK_IDS)?, LAST_UPDATE?,
BUSINESS_IMPACT?,
CVSS_ENVIRO_CDP?, CVSS_ENVIRO_TD?, CVSS_ENVIRO_CR?, CVSS_ENVIRO_IR?,
CVSS_ENVIRO_AR?,
DEFAULT_APPLIANCE_ID?, APPLIANCE_IDS?,
IP_SET?, DOMAIN_LIST?, DNS_LIST?, NETBIOS_LIST?,
HOST_IDS?, EC2_IDS?,
ASSIGNED_USER_IDS?, ASSIGNED_UNIT_IDS?, COMMENTS?)>

...
<!-- EC2_IDS -->
<!ELEMENT EC2_IDS (#PCDATA)>

<!-- HOST_IDS -->
```

```
<!ELEMENT HOST_IDS (#PCDATA)>

<!-- USER_IDS -->
<!ELEMENT ASSIGNED_USER_IDS (#PCDATA)>

<!-- UNIT_IDS -->
<!ELEMENT ASSIGNED_UNIT_IDS (#PCDATA)>

<!-- COMMENTS -->
<!ELEMENT COMMENTS (#PCDATA)>

<!-- WARNING -->
<!ELEMENT WARNING (CODE?, TEXT, URL?)>
<!ELEMENT CODE (#PCDATA)>
<!ELEMENT TEXT (#PCDATA)>
<!ELEMENT URL (#PCDATA)>

<!-- EOF -->
```

Compliance Authentication Report - DTD Change

API affected	/api/2.0/fo/report/ with action=fetch
New or Updated API	Neither (DTD change only)
DTD or XSD changes	Yes

The Compliance Authentication Report DTD is used when you download a saved authentication report from your account. We've made several changes to this DTD to add missing elements. The updated DTD (compliance_authentication_report.dtd) is shown below with changed and new elements highlighted in bold.

DTD update:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- QUALYS COMPLIANCE AUTHENTICATION REPORT DTD -->
<!-- $Revision$ -->

<!ELEMENT COMPLIANCE_AUTHENTICATION_REPORT (ERROR | (HEADER,
(BUSINESS_UNIT_LIST | ASSET_GROUP_LIST | ASSET_TAG_LIST | IPS_LIST)))>
<!ELEMENT ERROR (#PCDATA)>
<!ATTLIST ERROR number CDATA #IMPLIED>

<!ELEMENT HEADER (NAME, GENERATION_DATETIME, COMPANY_INFO, USER_INFO,
FILTERS)>
<!ELEMENT NAME (#PCDATA)>
<!ELEMENT GENERATION_DATETIME (#PCDATA)>

<!ELEMENT COMPANY_INFO (NAME, ADDRESS, CITY, STATE, COUNTRY, ZIP_CODE)>
<!ELEMENT ADDRESS (#PCDATA)>
<!ELEMENT CITY (#PCDATA)>
<!ELEMENT STATE (#PCDATA)>
<!ELEMENT COUNTRY (#PCDATA)>
<!ELEMENT ZIP_CODE (#PCDATA)>

<!ELEMENT USER_INFO (NAME, USERNAME?, ROLE)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT ROLE (#PCDATA)>

<!ELEMENT FILTERS (BUSINESS_UNIT_LIST | ASSET_GROUP_LIST | ASSET_TAG_LIST
| (IPS_LIST,  NETWORK?)>

<!ELEMENT BUSINESS_UNIT_LIST (BUSINESS_UNIT*)>
<!ELEMENT BUSINESS_UNIT
(NAME|AUTH_PASSED|AUTH_INSUFFICIENT|AUTH_FAILED|AUTH_NOT_ATTEMPTED|AUTH_N
OT_INSTALLED|AUTH_TOTAL|PASSED_PERCENTAGE|FAILED_PERCENTAGE|NOT_ATTEMPTED
_PERCENTAGE|TECHNOLOGY_LIST)*>
<!ELEMENT AUTH_PASSED (#PCDATA)>
```

```
<!ELEMENT AUTH_INSUFFICIENT (#PCDATA)>
<!ELEMENT AUTH_TOTAL (#PCDATA)>
<!ELEMENT PASSED_PERCENTAGE (#PCDATA)>

<!ELEMENT ASSET_TAG_LIST ((INCLUDED_TAGS, EXCLUDED_TAGS?) | ASSET_TAG)>
<!ELEMENT ASSET_TAG
(INCLUDED_TAGS|EXCLUDED_TAGS|AUTH_PASSED|AUTH_INSUFFICIENT|AUTH_FAILED|AU
TH_NOT_ATTEMPTED|AUTH_NOT_INSTALLED|AUTH_TOTAL|PASSED_PERCENTAGE|FAILED_P
ERCENTAGE|NOT_ATTEMPTED_PERCENTAGE|TECHNOLOGY_LIST)*>
<!ELEMENT INCLUDED_TAGS (TAG_ITEM+)>
<!ATTLIST INCLUDED_TAGS scope (any|all) #REQUIRED>
<!ELEMENT EXCLUDED_TAGS (TAG_ITEM+)>
<!ATTLIST EXCLUDED_TAGS scope (any|all) #REQUIRED>
<!ELEMENT TAG_ITEM (#PCDATA)>

<!ELEMENT ASSET_GROUP_LIST (ASSET_GROUP*)>
<!ELEMENT ASSET_GROUP
(NAME|AUTH_PASSED|AUTH_INSUFFICIENT|AUTH_FAILED|AUTH_NOT_ATTEMPTED|AUTH_N
OT_INSTALLED|AUTH_TOTAL|PASSED_PERCENTAGE|FAILED_PERCENTAGE|NOT_ATTEMPTED
_PERCENTAGE|TECHNOLOGY_LIST)*>

<!ELEMENT IPS_LIST (IPS+)>
<!ELEMENT IPS
(NAME|AUTH_PASSED|AUTH_INSUFFICIENT|AUTH_FAILED|AUTH_NOT_ATTEMPTED|AUTH_N
OT_INSTALLED|AUTH_TOTAL|PASSED_PERCENTAGE|FAILED_PERCENTAGE|NOT_ATTEMPTED
_PERCENTAGE|TECHNOLOGY_LIST)*>

<!ELEMENT AUTH_FAILED (#PCDATA)>
<!ELEMENT AUTH_NOT_ATTEMPTED (#PCDATA)>
<!ELEMENT AUTH_NOT_INSTALLED (#PCDATA)>
<!ELEMENT FAILED_PERCENTAGE (#PCDATA)>
<!ELEMENT NOT_ATTEMPTED_PERCENTAGE (#PCDATA)>

<!ELEMENT TECHNOLOGY_LIST (TECHNOLOGY*)>
<!ELEMENT TECHNOLOGY (NAME, HOST_LIST)>
<!ELEMENT HOST_LIST (HOST*)>
<!ELEMENT HOST (TRACKING_METHOD, IP, DNS?, NETBIOS?, HOST_TECHNOLOGY?,
INSTANCE?, STATUS, CAUSE?, NETWORK?, OS?, LAST_AUTH?, LAST_SUCCESS?)>
...
<!ELEMENT OS (#PCDATA)>
<!ELEMENT LAST_AUTH (#PCDATA)>
<!ELEMENT LAST_SUCCESS (#PCDATA)>
```

Dynamic Search List API - Support for CPE Type

API affected	/api/2.0/fo/qid/search_list/dynamic/
New or Updated API	Updated
DTD or XSD changes	Yes

The Dynamic Search List API lets you create/update dynamic search lists and get information about them. We've added API support for CPE "part" values (Operating System, Application, Hardware) in dynamic search lists, allowing you to target specific vulnerabilities for sending to the appropriate remediation teams.

Updates:

1) We added a new "cpe" input parameter for creating and updating dynamic search lists using the Dynamic Search List API.

2) When listing dynamic search lists the XML output will show the CPE value when specified in the search list criteria. The Dynamic Search List Output DTD was updated.

Input Parameters

Use this new input parameter when creating or updating a dynamic search list. For a complete list of input parameters, please refer to the API V2 User Guide.

Parameter	Description
cpe={value}	(Optional) The CPE value. Valid values are: Operating System, Application, Hardware, None. Multiple values are comma separated.

Examples

Create Dynamic Search List

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=create&title=api_with_cpe&cpe=Operating System,Hardware"
"https://qualysapi.qualys.com/api/2.0/fo/qid/search_list/dynamic/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
```

```
    <DATETIME>2018-01-08T19:28:08Z</DATETIME>
    <TEXT>New search list created successfully</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>172148</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

List Dynamic Search Lists

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl"
"https://qualysapi.qualys.com/api/2.0/fo/qid/search_list/dynamic/?action=
list&ids=172148">dl_ids.xml
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE DYNAMIC_SEARCH_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/qid/search_list/dynamic/dynamic_
list_output.dtd">
<DYNAMIC_SEARCH_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-01-08T19:30:27Z</DATETIME>
    <DYNAMIC_LISTS>
      <DYNAMIC_LIST>
        <ID>172148</ID>
        <TITLE><![CDATA[api_with_cpe]]></TITLE>
        <GLOBAL>No</GLOBAL>
        <OWNER><![CDATA[Patrick Slimmer (qualys_ps)]]></OWNER>
        <CREATED>2018-01-08T19:28:07Z</CREATED>
        <MODIFIED_BY><![CDATA[Patrick Slimmer (qualys_ps)]]></MODIFIED_BY>
        <MODIFIED>2018-01-08T19:28:07Z</MODIFIED>
        <QIDS>
          <QID>11887</QID>
          <QID>11889</QID>
          <QID>11892</QID>
          <QID>11893</QID>
          <QID>11899</QID>
          <QID>12446</QID>
          <QID>12757</QID>
          <QID>12838</QID>
          <QID>12941</QID>
          <QID>12950</QID>
          <QID>13108</QID>
```

```
            <QID>15079</QID>
            <QID>19109</QID>
            <QID>19867</QID>
            ...
            <QID>390066</QID>
            <QID>390067</QID>
            <QID>390069</QID>
            <QID>390070</QID>
            <QID>390072</QID>
            <QID>390073</QID>
            <QID>390076</QID>
            <QID>390079</QID>
            <QID>390080</QID>
            <QID>390084</QID>
          </QIDS>
          <CRITERIA>
            <DISCOVERY_METHOD><![CDATA[All]]></DISCOVERY_METHOD>
            <CPE><![CDATA[Operating System, Hardware]]></CPE>
          </CRITERIA>
        </DYNAMIC_LIST>
      </DYNAMIC_LISTS>
    </RESPONSE>
</DYNAMIC_SEARCH_LIST_OUTPUT>
```

DTD update:

The Dynamic Search List Output DTD (dynamic_list_output.dtd) was updated to include the new CPE element (in bold).

```
<!-- QUALYS DYNAMIC_SEARCH_LIST_OUTPUT DTD -->
<!-- $Revision$ -->
<!ELEMENT DYNAMIC_SEARCH_LIST_OUTPUT (REQUEST?,RESPONSE)>
...
<!ELEMENT CRITERIA (VULNERABILITY_TITLE?, DISCOVERY_METHOD?,
AUTHENTICATION_TYPE?, USER_CONFIGURATION?, CATEGORY?,
CONFIRMED_SEVERITY?, POTENTIAL_SEVERITY?, INFORMATION_SEVERITY?, VENDOR?,
PRODUCT?, CVSS_BASE_SCORE?, CVSS_TEMPORAL_SCORE?, CVSS3_BASE_SCORE?,
CVSS3_TEMPORAL_SCORE?, CVSS_ACCESS_VECTOR?, PATCH_AVAILABLE?,
VIRTUAL_PATCH_AVAILABLE?, CVE_ID?, EXPLOITABILITY?, ASSOCIATED_MALWARE?,
VENDOR_REFERENCE?, BUGTRAQ_ID?, VULNERABILITY_DETAILS?,
SUPPORTED_MODULES?, COMPLIANCE_DETAILS?, COMPLIANCE_TYPE?,
QUALYS_TOP_20?, OTHER?, NETWORK_ACCESS?, PROVIDER?,
CVSS_BASE_SCORE_OPERAND?, CVSS_TEMPORAL_SCORE_OPERAND?,
CVSS3_BASE_SCORE_OPERAND?, CVSS3_TEMPORAL_SCORE_OPERAND?, USER_MODIFIED?,
PUBLISHED?, SERVICE_MODIFIED?, CPE?)>
...
<!ELEMENT CPE (#PCDATA)>
<!-- EOF -->
```

New VM Scan Statistics API

API affected	/api/2.0/fo/scan/stats/?action=list
New or Updated API	New
DTD or XSD changes	Yes

The new VM Scan Statistics API (/api/2.0/fo/scan/stats/) allows customers to get details about vulnerability scans and assets that are waiting to be processed.

You'll see these sections in the XML output:

UNPROCESSED SCANS - The total number of scans that are not processed, including scans that are queued, running, loading, finished, etc.

VM RECRYPT BACKLOGS - The total number of assets across your finished scans that are waiting to be processed.

VM RECRYPT BACKLOGS BY SCAN - Scan details for vulnerability scans that are waiting to be processed. For each scan, you'll see the scan ID, scan title, scan status, processing priority and number of hosts that the scan finished but not processed.

VM RECRYPT BACKLOGS BY TASK - Processing task details for vulnerability scans that are waiting to be processed. For each task, you'll see the same scan details as VM RECRYPT BACKLOGS BY SCAN plus additional information like the total hosts alive for the scan, the number of hosts from the scan that have been processed, the number of hosts waiting to be processed, the scan start date, the task type and task status.

API Request and Sample Output

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl"
"https://qualysapi.qualys.com/api/2.0/fo/scan/stats/?action=list"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE TASK_PROCESSING SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/scan/stats/vm_recrypt_results.dt
d">
      <TASK_PROCESSING>
        <UNPROCESSED_SCANS><![CDATA[366]]></UNPROCESSED_SCANS>
        <VM_RECRYPT_BACKLOG><![CDATA[116]]></VM_RECRYPT_BACKLOG>
        <VM_RECRYPT_BACKLOG_BY_SCAN>
          <SCAN>
            <ID><![CDATA[189275]]></ID>
            <TITLE><![CDATA[API_V2_IP_Scan_1511513769]]></TITLE>
            <STATUS><![CDATA[Loading]]></STATUS>
```

```
            <PROCESSING_PRIORITY><![CDATA[None]]></PROCESSING_PRIORITY>
            <COUNT><![CDATA[2]]></COUNT>
          </SCAN>
          <SCAN>
            <ID><![CDATA[189281]]></ID>
            <TITLE><![CDATA[API_V2_AG_Scan_1511513846]]></TITLE>
            <STATUS><![CDATA[Loading]]></STATUS>
            <PROCESSING_PRIORITY><![CDATA[None]]></PROCESSING_PRIORITY>
            <COUNT><![CDATA[2]]></COUNT>
          </SCAN>
          <SCAN>
            <ID><![CDATA[190773]]></ID>
            <TITLE><![CDATA[API_V2_IP_Scan_]]></TITLE>
            <STATUS><![CDATA[Finished]]></STATUS>
            <PROCESSING_PRIORITY><![CDATA[None]]></PROCESSING_PRIORITY>
            <COUNT><![CDATA[2]]></COUNT>
          </SCAN>
          <SCAN>
            <ID><![CDATA[190775]]></ID>
            <TITLE><![CDATA[API_V2_IP_Scan_]]></TITLE>
            <STATUS><![CDATA[Finished]]></STATUS>
            <PROCESSING_PRIORITY><![CDATA[None]]></PROCESSING_PRIORITY>
            <COUNT><![CDATA[2]]></COUNT>
          </SCAN>
          ...
        </VM_RECRYPT_BACKLOG_BY_SCAN>
        <VM_RECRYPT_BACKLOG_BY_TASK>
          <SCAN>
            <ID><![CDATA[210337]]></ID>
            <TITLE><![CDATA[API_V2_AG_Scan_1515055579]]></TITLE>
            <STATUS><![CDATA[Loading]]></STATUS>
            <PROCESSING_PRIORITY><![CDATA[None]]></PROCESSING_PRIORITY>
            <NBHOST><![CDATA[]]></NBHOST>
            <TO_PROCESS><![CDATA[3]]></TO_PROCESS>
            <PROCESSED><![CDATA[0]]></PROCESSED>
            <SCAN_DATE><![CDATA[2018-01-04T08:46:13Z]]></SCAN_DATE>
            <SCAN_UPDATED_DATE><![CDATA[2018-01-
04T08:58:05Z]]></SCAN_UPDATED_DATE>
            <TASK_TYPE><![CDATA[VM Scan Processing]]></TASK_TYPE>
            <TASK_STATUS><![CDATA[Queued]]></TASK_STATUS>
            <TASK_UPDATED_DATE><![CDATA[2018-01-
12T08:17:09Z]]></TASK_UPDATED_DATE>
          </SCAN>
          <SCAN>
            <ID><![CDATA[215356]]></ID>
            <TITLE><![CDATA[API_V2_AG_Scan_1515742250]]></TITLE>
            <STATUS><![CDATA[Running]]></STATUS>
            <PROCESSING_PRIORITY><![CDATA[None]]></PROCESSING_PRIORITY>
            <NBHOST><![CDATA[]]></NBHOST>
```

```
            <TO_PROCESS><![CDATA[0]]></TO_PROCESS>
            <PROCESSED><![CDATA[0]]></PROCESSED>
            <SCAN_DATE><![CDATA[2018-01-12T07:30:42Z]]></SCAN_DATE>
            <SCAN_UPDATED_DATE><![CDATA[2018-01-
12T08:01:10Z]]></SCAN_UPDATED_DATE>
            <TASK_TYPE><![CDATA[VM Scan Processing]]></TASK_TYPE>
            <TASK_STATUS><![CDATA[Queued]]></TASK_STATUS>
            <TASK_UPDATED_DATE><![CDATA[2018-01-
12T08:17:11Z]]></TASK_UPDATED_DATE>
          </SCAN>
          <SCAN>
            <ID><![CDATA[215357]]></ID>
            <TITLE><![CDATA[API_V2_AG_Scan_1515742265]]></TITLE>
            <STATUS><![CDATA[Loading]]></STATUS>
            <PROCESSING_PRIORITY><![CDATA[None]]></PROCESSING_PRIORITY>
            <NBHOST><![CDATA[]]></NBHOST>
            <TO_PROCESS><![CDATA[0]]></TO_PROCESS>
            <PROCESSED><![CDATA[0]]></PROCESSED>
            <SCAN_DATE><![CDATA[2018-01-12T07:30:58Z]]></SCAN_DATE>
            <SCAN_UPDATED_DATE><![CDATA[2018-01-
12T08:14:45Z]]></SCAN_UPDATED_DATE>
            <TASK_TYPE><![CDATA[VM Scan Processing]]></TASK_TYPE>
            <TASK_STATUS><![CDATA[Queued]]></TASK_STATUS>
            <TASK_UPDATED_DATE><![CDATA[2018-01-
12T08:17:11Z]]></TASK_UPDATED_DATE>
          </SCAN>
          ...
        </VM_RECRYPT_BACKLOG_BY_TASK>
      </TASK_PROCESSING>
```

VM Recrypt Results DTD

DTD: <base_url>/api/2.0/fo/scan/stats/vm_recrypt_results.dtd

```
<!ELEMENT TASK_PROCESSING (UNPROCESSED_SCANS?, VM_RECRYPT_BACKLOG?,
VM_RECRYPT_BACKLOG_BY_SCAN?, VM_RECRYPT_BACKLOG_BY_TASK?)>

<!ELEMENT UNPROCESSED_SCANS (#PCDATA)>
<!ELEMENT VM_RECRYPT_BACKLOG (#PCDATA)>
<!ELEMENT VM_RECRYPT_BACKLOG_BY_SCAN (SCAN*)>
<!ELEMENT VM_RECRYPT_BACKLOG_BY_TASK (SCAN*)>

<!ELEMENT SCAN (ID?, TITLE?, STATUS?, PROCESSING_PRIORITY?, COUNT?,
NBHOST?, TO_PROCESS?, PROCESSED?, SCAN_DATE?, SCAN_UPDATED_DATE?,
TASK_TYPE?, TASK_STATUS?, TASK_UPDATED_DATE?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT STATUS (#PCDATA)>
<!ELEMENT PROCESSING_PRIORITY (#PCDATA)>
```

```
<!ELEMENT COUNT (#PCDATA)>
<!ELEMENT NBHOST (#PCDATA)>
<!ELEMENT TO_PROCESS (#PCDATA)>
<!ELEMENT PROCESSED (#PCDATA)>
<!ELEMENT SCAN_DATE (#PCDATA)>
<!ELEMENT SCAN_UPDATED_DATE (#PCDATA)>
<!ELEMENT TASK_TYPE (#PCDATA)>
<!ELEMENT TASK_STATUS (#PCDATA)>
<!ELEMENT TASK_UPDATED_DATE (#PCDATA)>
```

Host List Detection API - New ARF Filters for Kernel, Service and Configuration

API affected	/api/2.0/fo/asset/host/vm/detection/
New or Updated API	Updated
DTD or XSD changes	Yes

You can now filter your host detection list based on Acceptable Risk Factors (ARF) related to kernel, service and host configuration. The risk factor or exploitability of a detected vulnerability is based on an ARF rule, which is pre-defined by Qualys.

For example, let's say QID 12345 is related to a specific kernel (K1). If K1 is running, then QID 12345 can be exploited. If K1 is not running, then QID 12345 cannot be exploited.

We currently have 3 ARF rules:

ARF kernel rule - The risk factor is based on the status of the associated kernel (running or not running).

ARF service rule - The risk factor is based on the status of the associated service (running or not running).

ARF configuration rule - The risk factor is based on the host configuration settings (the configuration either prevents exploitation or not).

These sections appear in the API output:

AFFECT RUNNING KERNEL - A value of 1 indicates that the QID is exploitable because it was found on a running kernel. A value of 0 indicates that it is not exploitable because it was found on a non-running kernel.

AFFECT RUNNING SERVICE - A value of 1 indicates that the QID is exploitable because it was found on a running port/service. A value of 0 indicates that it is not exploitable because it was found on a non-running port/service.

AFFECT EXPLOITABLE CONFIG - A value of 1 indicates that the QID is exploitable due to the current host configuration. A value of 0 indicates that it is not exploitable due to the current host configuration.

Input Parameters

New input parameters for filtering the host detection list are described below.

Parameter	Description
arf_kernel_filter={0|1|2|3|4}	(Optional) Identify vulnerabilities found on running or non-running Linux kernels. Good to Know - It's possible that multiple kernels are detected on a single Linux host. You'll notice the scan results report the running kernel on each Linux host in Info Gathered QID 45097. When unspecified, vulnerabilities are not filtered based on kernel activity. <AFFECT_RUNNING_KERNEL> does not appear in the output. When set to 0, vulnerabilities are not filtered based on kernel activity. <AFFECT_RUNNING_KERNEL> appears in the output for kernel related vulnerabilities. When set to 1, exclude kernel related vulnerabilities that are not exploitable (found on non-running kernels). <AFFECT_RUNNING_KERNEL> appears in the output for kernel related vulnerabilities. When set to 2, only include kernel related vulnerabilities that are not exploitable (found on non-running kernels). <AFFECT_RUNNING_KERNEL> appears in the output with a value of 0 for each detection. When set to 3, only include kernel related vulnerabilities that are exploitable (found on running kernels). <AFFECT_RUNNING_KERNEL> appears in the output with a value of 1 for each detection. When set to 4, only include kernel related vulnerabilities. <AFFECT_RUNNING_KERNEL> appears in the output with a value of 0 or 1 for each detection.
	Note that active_kernels_only is now deprecated and will be removed in a future release. Please use arf_kernel_filter instead.

Parameter	Description
arf_service_filter={0|1|2|3|4}	(Optional) Identify vulnerabilities found on running or non-running ports/services. When unspecified, vulnerabilities are not filtered based on running ports/services. <AFFECT_RUNNING_SERVICE> does not appear in the output. When set to 0, vulnerabilities are not filtered based on running ports/services. <AFFECT_RUNNING_SERVICE> appears in the output for service related vulnerabilities. When set to 1, exclude service related vulnerabilities that are not exploitable (found on non-running ports/services). <AFFECT_RUNNING_SERVICE> appears in the output for service related vulnerabilities. When set to 2, only include service related vulnerabilities that are not exploitable (found on non-running ports/services). <AFFECT_RUNNING_SERVICE> appears in the output with a value of 0 for each detection. When set to 3, only include exploitable service related vulnerabilities (found on running ports/services). <AFFECT_RUNNING_SERVICE> appears in the output with a value of 1 for each detection. When set to 4, only include service related vulnerabilities. <AFFECT_RUNNING_SERVICE> appears in the output with a value of 0 or 1 for each detection.
arf_config_filter={0|1|2|3|4}	(Optional) Identify vulnerabilities that may or may not be exploitable due to the current host configuration. When unspecified, vulnerabilities are not filtered based on host configuration. <AFFECT_EXPLOITABLE_CONFIG> does not appear in the output. When set to 0, vulnerabilities are not filtered based on host configuration. <AFFECT_EXPLOITABLE_CONFIG> appears in the output for config related vulnerabilities. When set to 1, exclude vulnerabilities not exploitable due to host configuration. <AFFECT_EXPLOITABLE_CONFIG> appears in the output for config related detections. When set to 2, only include config related vulnerabilities that are not exploitable. <AFFECT_EXPLOITABLE_CONFIG> appears in the output with a value of 0 for each detection. When set to 3, only include config related vulnerabilities that are exploitable. <AFFECT_EXPLOITABLE_CONFIG> appears in the output with a value of 1 for each detection. When set to 4, only include config related vulnerabilities. <AFFECT_EXPLOITABLE_CONFIG> appears in the output with a value of 0 or 1 for each detection.

Examples

Sample API call with XML output

API request:

```
curl -u "username:password" -H "X-Requested-With:curl demo2" -d
"action=list&status=New,Active,Re-
Opened,Fixed&arf_kernel_filter=4&arf_service_filter=4&arf_config_filter=4
&ips=10.10.10.65"
"https://qualysapi.qualys.com/api/2.0/fo/asset/host/vm/detection/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE HOST_LIST_VM_DETECTION_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/asset/host/vm/detection/host_lis
t_vm_detection_output.dtd">
<HOST_LIST_VM_DETECTION_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-01-22T04:17:37Z</DATETIME>
    <HOST_LIST>
      <HOST>
        <ID>19011</ID>
        <IP>10.10.10.65</IP>
        <TRACKING_METHOD>IP</TRACKING_METHOD>
        <OS><![CDATA[Debian Linux 4.0]]></OS>
        <DNS><![CDATA[krb5.qualys.com]]></DNS>
        <LAST_SCAN_DATETIME>2018-01-09T05:30:22Z</LAST_SCAN_DATETIME>
        <LAST_VM_SCANNED_DATE>2018-01-15T14:13:03Z</LAST_VM_SCANNED_DATE>
        <LAST_VM_SCANNED_DURATION>617</LAST_VM_SCANNED_DURATION>
        <LAST_VM_AUTH_SCANNED_DATE>2017-11-
16T17:38:08Z</LAST_VM_AUTH_SCANNED_DATE>
       <LAST_VM_AUTH_SCANNED_DURATION>922</LAST_VM_AUTH_SCANNED_DURATION>
        <LAST_PC_SCANNED_DATE>2017-11-16T16:50:27Z</LAST_PC_SCANNED_DATE>
        <DETECTION_LIST>
          <DETECTION>
            <QID>82024</QID>
            <TYPE>Confirmed</TYPE>
            <SEVERITY>2</SEVERITY>
            <SSL>0</SSL>
            <RESULTS><![CDATA[IP_ID=0]]></RESULTS>
            <STATUS>Fixed</STATUS>
            <FIRST_FOUND_DATETIME>2016-09-
09T09:40:39Z</FIRST_FOUND_DATETIME>
            <LAST_FOUND_DATETIME>2017-11-
16T17:38:08Z</LAST_FOUND_DATETIME>
            <TIMES_FOUND>1064</TIMES_FOUND>
            <LAST_TEST_DATETIME>2018-01-09T04:59:58Z</LAST_TEST_DATETIME>
            <LAST_UPDATE_DATETIME>2018-01-
```

```
09T05:30:22Z</LAST_UPDATE_DATETIME>
            <LAST_FIXED_DATETIME>2017-12-
16T00:05:10Z</LAST_FIXED_DATETIME>
            <IS_IGNORED>0</IS_IGNORED>
            <IS_DISABLED>0</IS_DISABLED>
            <AFFECT_RUNNING_KERNEL>1</AFFECT_RUNNING_KERNEL>
            <AFFECT_RUNNING_SERVICE>1</AFFECT_RUNNING_SERVICE>
            <AFFECT_EXPLOITABLE_CONFIG>1</AFFECT_EXPLOITABLE_CONFIG>
            <LAST_PROCESSED_DATETIME>2018-01-
09T05:30:22Z</LAST_PROCESSED_DATETIME>
          </DETECTION>
          <DETECTION>
            <QID>82054</QID>
            <TYPE>Confirmed</TYPE>
            <SEVERITY>2</SEVERITY>
            <SSL>0</SSL>
            <RESULTS><![CDATA[Tested on port 111 with an injected SYN/RST
offset by 16 bytes. Tested on port 22 with an injected SYN/RST offset by
16 bytes.]]></RESULTS>
            <STATUS>Fixed</STATUS>
            <FIRST_FOUND_DATETIME>2016-09-
09T09:40:39Z</FIRST_FOUND_DATETIME>
            <LAST_FOUND_DATETIME>2017-11-
16T17:38:08Z</LAST_FOUND_DATETIME>
            <TIMES_FOUND>1494</TIMES_FOUND>
            <LAST_TEST_DATETIME>2018-01-09T04:59:58Z</LAST_TEST_DATETIME>
            <LAST_UPDATE_DATETIME>2018-01-
09T05:30:22Z</LAST_UPDATE_DATETIME>
            <LAST_FIXED_DATETIME>2017-12-
16T00:05:10Z</LAST_FIXED_DATETIME>
            <IS_IGNORED>0</IS_IGNORED>
            <IS_DISABLED>0</IS_DISABLED>
            <AFFECT_RUNNING_KERNEL>0</AFFECT_RUNNING_KERNEL>
            <AFFECT_RUNNING_SERVICE>0</AFFECT_RUNNING_SERVICE>
            <AFFECT_EXPLOITABLE_CONFIG>0</AFFECT_EXPLOITABLE_CONFIG>
            <LAST_PROCESSED_DATETIME>2018-01-
09T05:30:22Z</LAST_PROCESSED_DATETIME>
          </DETECTION>
        </DETECTION_LIST>
      </HOST>
    </HOST_LIST>
  </RESPONSE>
</HOST_LIST_VM_DETECTION_OUTPUT>
```

Sample API call with CSV output

API request:

```
curl -u "username:password" -H "X-Requested-With:curl demo2" -d
"action=list&status=New,Active,Re-
Opened,Fixed&arf_kernel_filter=4&arf_service_filter=4&arf_config_filter=4
&ips=10.10.10.65&output_format=CSV"
"https://qualysapi.qualys.com/api/2.0/fo/asset/host/vm/detection/"
```

CSV output:

```
----BEGIN_RESPONSE_HEADER_CSV
----END_RESPONSE_HEADER_CSV
----BEGIN_RESPONSE_BODY_CSV
"Host ID","IP Address","Tracking Method","Operating System","DNS
Name","Netbios Name","QG HostID","Last Scan Datetime","OS CPE","Last VM
Scanned Date","Last VM Scanned Duration","Last VM Auth Scanned Date","Last
VM Auth Scanned Duration","Last PC Scanned
Date","QID","Type","Port","Protocol","FQDN","SSL","Instance","Status","Se
verity","First Found Datetime","Last Found Datetime","Last Test
Datetime","Last Update Datetime","Last Fixed
Datetime","Results","Ignored","Disabled","Times Found","Service","Affect
Running Kernel","Affect Running Services","Affect Exploitable
Config","Last Processed Datetime"
"19011","10.10.10.65","IP","Debian Linux 4.0","krb5.qualys.com",,,"2018-
01-09T05:30:22Z",,"2018-01-15T14:13:03Z","617","2017-11-
16T17:38:08Z","922","2017-11-
16T16:50:27Z","82024","Confirmed",,,,"0",,"Fixed","2","2016-09-
09T09:40:39Z","2017-11-16T17:38:08Z","2018-01-09T04:59:58Z","2018-01-
09T05:30:22Z","2017-12-
16T00:05:10Z","IP_ID=0","0","0","1064",,"1","1","1","2018-01-
09T05:30:22Z"
"19011","10.10.10.65","IP","Debian Linux 4.0","krb5.qualys.com",,,"2018-
01-09T05:30:22Z",,"2018-01-15T14:13:03Z","617","2017-11-
16T17:38:08Z","922","2017-11-
16T16:50:27Z","82054","Confirmed",,,,"0",,"Fixed","2","2016-09-
09T09:40:39Z","2017-11-16T17:38:08Z","2018-01-09T04:59:58Z","2018-01-
09T05:30:22Z","2017-12-16T00:05:10Z","Tested on port 111 with an injected
SYN/RST offset by 16 bytes.
Tested on port 22 with an injected SYN/RST offset by 16
bytes.","0","0","1494",,"0","0","0","2018-01-09T05:30:22Z"
----END_RESPONSE_BODY_CSV
----BEGIN_RESPONSE_FOOTER_CSV
"Status Message"
"Finished"
----END_RESPONSE_FOOTER_CSV
```

DTD Update

We updated the Host List Detection Output DTD (host_list_vm_detection_output.dtd). New elements are in bold. (Note that AFFECT_RUNNING_KERNEL was present in earlier versions of the DTD.)

```
<!ELEMENT DETECTION_LIST (DETECTION+)>
<!ELEMENT DETECTION (QID, TYPE, SEVERITY?, PORT?, PROTOCOL?, FQDN?, SSL?,
INSTANCE?, RESULTS?, STATUS?, FIRST_FOUND_DATETIME?,
LAST_FOUND_DATETIME?, TIMES_FOUND?, LAST_TEST_DATETIME?,
LAST_UPDATE_DATETIME?, LAST_FIXED_DATETIME?,
FIRST_REOPENED_DATETIME?, LAST_REOPENED_DATETIME?, TIMES_REOPENED?,
SERVICE?, IS_IGNORED?, IS_DISABLED?, AFFECT_RUNNING_KERNEL?,
AFFECT_RUNNING_SERVICE?, AFFECT_EXPLOITABLE_CONFIG?,
LAST_PROCESSED_DATETIME? )>
...
<!ELEMENT AFFECT_RUNNING_KERNEL (#PCDATA)>
<!ELEMENT AFFECT_RUNNING_SERVICE (#PCDATA)>
<!ELEMENT AFFECT_EXPLOITABLE_CONFIG (#PCDATA)>
<!ELEMENT LAST_PROCESSED_DATETIME (#PCDATA)>
<!ELEMENT WARNING (CODE?, TEXT, URL?)>
<!ELEMENT CODE (#PCDATA)>
<!ELEMENT TEXT (#PCDATA)>
<!ELEMENT URL (#PCDATA)>
<!-- EOF -->
```

Scan Schedule API - Enhanced EC2 Details

API affected	/api/2.0/fo/schedule/scan/
New or Updated API	Updated
DTD or XSD changes	Yes

The Scan Schedule API v2 supports defining schedules for vulnerability scans. We now provide you more details about your EC2 connector. Using the list action, you can now view details such as the provider (Amazon Web Services-AWS), connector name, the unique UUID assigned to it, the region, type of scan, and so on.

Input Parameters

Use this new input parameter to view the cloud details in the output. For a complete list of input parameters, please refer to the API V2 User Guide.

Parameter	Description
show_cloud_details={**0**|1}	(Optional) Specify 1 to display the cloud details (PROVIDER, CONNECTOR, SCAN_TYPE, CLOUD_TARGET) in the XML output. Otherwise the details are not displayed in the output.

Examples

List Scan Schedule

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=list&show_cloud_details=1&id=174576"
"https://qualysapi.qualys.com//api/2.0/fo/schedule/scan/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SCHEDULE_SCAN_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/schedule/scan/schedule_scan_list
_output.dtd">
<SCHEDULE_SCAN_LIST_OUTPUT>
 <RESPONSE>
   <DATETIME>2018-01-17T11:45:49Z</DATETIME>
   <SCHEDULE_SCAN_LIST>
     <SCAN>
       <ID>174576</ID>
       <ACTIVE>1</ACTIVE>
       <TITLE><![CDATA[New-Scheduled-Scan]]></TITLE>
       <USER_LOGIN>quays_sp35</USER_LOGIN>
```

```
        <TARGET><![CDATA[Asset Tags Included]]></TARGET>
        <ISCANNER_NAME><![CDATA[External Scanner]]></ISCANNER_NAME>
        <EC2_INSTANCE>
          <CONNECTOR_UUID><![CDATA[202fa4b2-7da7-4b80-a9de-
5d3c0ce9ac27]]></CONNECTOR_UUID>
          <EC2_ENDPOINT><![CDATA[1507b6c1-07a7-4d88-acf2-
8c6b63e749c4]]></EC2_ENDPOINT>
          <EC2_ONLY_CLASSIC><![CDATA[0]]></EC2_ONLY_CLASSIC>
        </EC2_INSTANCE>
        <CLOUD_DETAILS>
          <PROVIDER>AWS</PROVIDER>
          <CONNECTOR>
            <UUID>202fa4b2-7da7-4b80-a9de-5d3c0ce9ac27</UUID>
            <NAME><![CDATA[conn-us-east-1]]></NAME>
          </CONNECTOR>
          <SCAN_TYPE>Internal</SCAN_TYPE>
          <CLOUD_TARGET>
            <PLATFORM>VPC</PLATFORM>
            <REGION>
              <UUID>1507b6c1-07a7-4d88-acf2-8c6b63e749c4</UUID>
              <CODE>us-east-1</CODE>
              <NAME><![CDATA[US East (N. Virginia)]]></NAME>
            </REGION>
            <VPC_SCOPE>All</VPC_SCOPE>
          </CLOUD_TARGET>
        </CLOUD_DETAILS>
        <ASSET_TAGS>
          <TAG_INCLUDE_SELECTOR>any</TAG_INCLUDE_SELECTOR>
          <TAG_SET_INCLUDE><![CDATA[EC2---NEW--TAG]]></TAG_SET_INCLUDE>
          <USE_IP_NT_RANGE_TAGS>0</USE_IP_NT_RANGE_TAGS>
        </ASSET_TAGS>
       ...
        </SCHEDULE>
      </SCAN>
    </SCHEDULE_SCAN_LIST>
  </RESPONSE>
```

DTD update:

The schedule scan list DTD (schedule_scan_list_output.dtd) was updated to include the new elements (in bold).

```
<!-- QUALYS SCHEDULE_SCAN_LIST_OUTPUT DTD -->
<!-- $Revision$ -->
<!ELEMENT SCHEDULE_SCAN_LIST_OUTPUT (REQUEST?,RESPONSE)>
...
<!ELEMENT CONNECTOR_UUID (#PCDATA)>
<!ELEMENT EC2_ENDPOINT (#PCDATA)>
<!ELEMENT EC2_ONLY_CLASSIC (#PCDATA)>
```

```
<!ELEMENT CLOUD_DETAILS (PROVIDER, CONNECTOR, SCAN_TYPE, CLOUD_TARGET)>
<!ELEMENT PROVIDER (#PCDATA)>
<!ELEMENT CONNECTOR (ID?, UUID, NAME)>
<!ELEMENT UUID (#PCDATA)>
<!ELEMENT NAME (#PCDATA)>
<!ELEMENT SCAN_TYPE (#PCDATA)>
<!ELEMENT CLOUD_TARGET (PLATFORM, REGION?, VPC_SCOPE, VPC_LIST?)>
<!ELEMENT PLATFORM (#PCDATA)>
<!ELEMENT REGION (UUID, CODE?, NAME?)>
<!ELEMENT CODE (#PCDATA)>
<!ELEMENT VPC_SCOPE (#PCDATA)>
<!ELEMENT VPC_LIST (VPC+)>
<!ELEMENT VPC (UUID)>

<!ELEMENT ASSET_GROUP_TITLE_LIST (ASSET_GROUP_TITLE+)>
...
<!ELEMENT DISTRIBUTION_GROUPS (DISTRIBUTION_GROUP+)>
<!ELEMENT DISTRIBUTION_GROUP (ID, TITLE)>
<!-- EOF -->
```

New element in Authentication Records List DTD

We've made DTD changes to add new element to the authentication record list output. This is pre-release functionality scheduled for a future release related to VMware vCenter authentication support.

List Authentication Records API

We've updated DTD auth_records.dtd. This new element was added **AUTH_VCENTER_IDS**. This will not appear in XML output at this time.

API affected	/api/2.0/fo/auth/?action=list
New or Updated API	Neither (DTD change only)
DTD or XSD changes	Yes

DTD update

Update DTD: https://<baseurl>/api/2.0/fo/auth/auth_records.dtd

New element **AUTH_VCENTER_IDS** is shown in bold below.

```
<!-- QUALYS AUTH_RECORDS_OUTPUT DTD -->

<!ELEMENT AUTH_RECORDS_OUTPUT (REQUEST?, RESPONSE)>

<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
                   POST_DATA?)>
<!ELEMENT DATETIME (#PCDATA)>
<!ELEMENT USER_LOGIN (#PCDATA)>
<!ELEMENT RESOURCE (#PCDATA)>
<!ELEMENT PARAM_LIST (PARAM+)>
<!ELEMENT PARAM (KEY, VALUE)>
<!ELEMENT KEY (#PCDATA)>
<!ELEMENT VALUE (#PCDATA)>
<!-- if returned, POST_DATA will be urlencoded -->
<!ELEMENT POST_DATA (#PCDATA)>

<!ELEMENT RESPONSE (DATETIME, AUTH_RECORDS?, WARNING_LIST?)>
<!ELEMENT AUTH_RECORDS (AUTH_UNIX_IDS?, AUTH_WINDOWS_IDS?,
AUTH_ORACLE_IDS?, AUTH_ORACLE_LISTENER_IDS?, AUTH_SNMP_IDS?,
AUTH_MS_SQL_IDS?, AUTH_IBM_DB2_IDS?, AUTH_VMWARE_IDS?, AUTH_MS_IIS_IDS?,
AUTH_APACHE_IDS?, AUTH_IBM_WEBSPHERE_IDS?, AUTH_HTTP_IDS?,
AUTH_SYBASE_IDS?, AUTH_MYSQL_IDS?, AUTH_TOMCAT_IDS?,
AUTH_ORACLE_WEBLOGIC_IDS?, AUTH_DOCKER_IDS?, AUTH_POSTGRESQL_IDS?,
AUTH_MONGODB_IDS?, AUTH_PALO_ALTO_FIREWALL_IDS?, AUTH_VCENTER_IDS?)>

<!ELEMENT AUTH_UNIX_IDS (ID_SET)>
```

```
<!ELEMENT AUTH_WINDOWS_IDS (ID_SET)>
<!ELEMENT AUTH_ORACLE_IDS (ID_SET)>
<!ELEMENT AUTH_ORACLE_LISTENER_IDS (ID_SET)>
<!ELEMENT AUTH_SNMP_IDS (ID_SET)>
<!ELEMENT AUTH_MS_SQL_IDS (ID_SET)>
<!ELEMENT AUTH_IBM_DB2_IDS (ID_SET)>
<!ELEMENT AUTH_VMWARE_IDS (ID_SET)>
<!ELEMENT AUTH_MS_IIS_IDS (ID_SET)>
<!ELEMENT AUTH_APACHE_IDS (ID_SET)>
<!ELEMENT AUTH_IBM_WEBSPHERE_IDS (ID_SET)>
<!ELEMENT AUTH_HTTP_IDS (ID_SET)>
<!ELEMENT AUTH_SYBASE_IDS (ID_SET)>
<!ELEMENT AUTH_MYSQL_IDS (ID_SET)>
<!ELEMENT AUTH_TOMCAT_IDS (ID_SET)>
<!ELEMENT AUTH_ORACLE_WEBLOGIC_IDS (ID_SET)>
<!ELEMENT AUTH_DOCKER_IDS (ID_SET)>
<!ELEMENT AUTH_POSTGRESQL_IDS (ID_SET)>
<!ELEMENT AUTH_MONGODB_IDS (ID_SET)>
<!ELEMENT AUTH_PALO_ALTO_FIREWALL_IDS (ID_SET)>
<!ELEMENT AUTH_VCENTER_IDS (ID_SET)>

<!ELEMENT WARNING_LIST (WARNING+)>
<!ELEMENT WARNING (CODE?, TEXT, URL?, ID_SET?)>
<!ELEMENT CODE (#PCDATA)>
<!ELEMENT TEXT (#PCDATA)>
<!ELEMENT URL (#PCDATA)>

<!ELEMENT ID_SET (ID|ID_RANGE)+>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT ID_RANGE (#PCDATA)>

<!-- EOF -->
```

Vault Support for VMware Authentication

Now users can configure VMware authentication records to use vaults to access credentials used for authentication.

Updated API endpoints:

Launch Scan API | Vault Support for VMware Authentication

List VMware Authentication Records API

We've updated DTD auth_vmware_list_output.dtd and XML output using this DTD. These new elements were added **LOGIN_TYPE** and **DIGITAL_VAULT**.

LOGIN_TYPE will be set to "basic" or "vault" depending on whether vault is enabled. DIGITAL_VAULT will show vault parameters if vault is enabled.

API affected	/api/2.0/fo/auth/vmware/?action=list
New or Updated API	Neither (DTD change only)
DTD or XSD changes	Yes

Example with vault Not Enabled

List VMware authentication record ID with vault not enabled, with all details. New element <LOGIN_TYPE> is set to "basic".

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -X "POST" -d
"action=list&ids=176569&details=All"
"https://qualysapi.qualys.com/api/2.0/fo/auth/vmware/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_VMWARE_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com.qualys.com/api/2.0/fo/auth/vmware/auth_vmwa
re_list_output.dtd">
<AUTH_VMWARE_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-01-22T20:26:44Z</DATETIME>
    <AUTH_VMWARE_LIST>
      <AUTH_VMWARE>
        <ID>176569</ID>
        <TITLE><![CDATA[VMware Auth Record]]></TITLE>
        <USERNAME><![CDATA[acme_sw2]]></USERNAME>
        <PORT>443</PORT>
        <SSL_VERIFY><![CDATA[none]]></SSL_VERIFY>
        <IP_SET>
```

```
            <IP>10.10.10.10</IP>
          </IP_SET>
          <LOGIN_TYPE><![CDATA[basic]]></LOGIN_TYPE>
          <NETWORK_ID>0</NETWORK_ID>
          <CREATED>
            <DATETIME>2018-01-22T20:26:17Z</DATETIME>
            <BY>acme_sw2</BY>
          </CREATED>
          <LAST_MODIFIED>
            <DATETIME>2018-01-22T20:26:17Z</DATETIME>
          </LAST_MODIFIED>
          <COMMENTS><![CDATA[VMware Auth Record comments]]></COMMENTS>
        </AUTH_VMWARE>
      </AUTH_VMWARE_LIST>
      <GLOSSARY>
        <USER_LIST>
          <USER>
            <USER_LOGIN>acme_sw2</USER_LOGIN>
            <FIRST_NAME>Sherry</FIRST_NAME>
            <LAST_NAME>Wang</LAST_NAME>
          </USER>
        </USER_LIST>
      </GLOSSARY>
    </RESPONSE>
</AUTH_VMWARE_LIST_OUTPUT>
```

Example with vault enabled

List VMware authentication record ID 170633 with vault enabled, showing all details. New element <LOGIN_TYPE> is set to "vault". New element <DIGITAL_VAULT> and its sub-elements are populated for the vault type used.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -X "POST" -d
"action=list&ids=170633&details=All"
"https://qualysapi.qualys.com/api/2.0/fo/auth/vmware/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_VMWARE_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/vmware/auth_vmware_list_out
put.dtd">
<AUTH_VMWARE_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-01-22T20:30:05Z</DATETIME>
    <AUTH_VMWARE_LIST>
      <AUTH_VMWARE>
        <ID>170633</ID>
```

```
        <TITLE><![CDATA[VWware record - using vault]]></TITLE>
        <USERNAME><![CDATA[acme_sw2]]></USERNAME>
        <PORT>443</PORT>
        <SSL_VERIFY><![CDATA[all]]></SSL_VERIFY>
        <IP_SET>
          <IP>10.10.10.11</IP>
        </IP_SET>
        <LOGIN_TYPE><![CDATA[vault]]></LOGIN_TYPE>
        <DIGITAL_VAULT>
          <DIGITAL_VAULT_ID><![CDATA[166657]]></DIGITAL_VAULT_ID>
          <DIGITAL_VAULT_TYPE><![CDATA[CA Access
Control]]></DIGITAL_VAULT_TYPE>
          <DIGITAL_VAULT_TITLE><![CDATA[5-CA Access
Control]]></DIGITAL_VAULT_TITLE>
          <VAULT_EP_NAME><![CDATA[name]]></VAULT_EP_NAME>
          <VAULT_EP_TYPE><![CDATA[type]]></VAULT_EP_TYPE>
          <VAULT_EP_CONT><![CDATA[container]]></VAULT_EP_CONT>
        </DIGITAL_VAULT>
        <NETWORK_ID>0</NETWORK_ID>
        <CREATED>
          <DATETIME>2018-01-11T21:14:37Z</DATETIME>
          <BY>acme_sw2</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2018-01-11T21:14:37Z</DATETIME>
        </LAST_MODIFIED>
      </AUTH_VMWARE>
    </AUTH_VMWARE_LIST>
    <GLOSSARY>
      <USER_LIST>
        <USER>
          <USER_LOGIN>acme_sw2</USER_LOGIN>
          <FIRST_NAME>Sherry</FIRST_NAME>
          <LAST_NAME>Wu</LAST_NAME>
        </USER>
      </USER_LIST>
    </GLOSSARY>
  </RESPONSE>
</AUTH_VMWARE_LIST_OUTPUT
```

DTD update

Updated DTD: https://<base_url>/api/2.0/fo/auth/vmware/auth_vmware_list_output.dtd

New elements **LOGIN_TYPE** and **DIGITAL_VAULT** are shown in bold below.

```
<!-- QUALYS AUTH_VMWARE_LIST_OUTPUT DTD -->

<!ELEMENT AUTH_VMWARE_LIST_OUTPUT (REQUEST?, RESPONSE)>

<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
POST_DATA?)>
<!ELEMENT DATETIME (#PCDATA)>
<!ELEMENT USER_LOGIN (#PCDATA)>
<!ELEMENT RESOURCE (#PCDATA)>
<!ELEMENT PARAM_LIST (PARAM+)>
<!ELEMENT PARAM (KEY, VALUE)>
<!ELEMENT KEY (#PCDATA)>
<!ELEMENT VALUE (#PCDATA)>
<!-- if returned, POST_DATA will be urlencoded -->
<!ELEMENT POST_DATA (#PCDATA)>

<!ELEMENT RESPONSE (DATETIME, (AUTH_VMWARE_LIST|ID_SET)?, WARNING_LIST?,
GLOSSARY?)>
<!ELEMENT AUTH_VMWARE_LIST (AUTH_VMWARE+)>

<!ELEMENT AUTH_VMWARE (ID, TITLE, USERNAME?, PORT, SSL_VERIFY?, HOSTS?,
IP_SET, LOGIN_TYPE?, DIGITAL_VAULT?, NETWORK_ID?, CREATED, LAST_MODIFIED,
COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>

<!ELEMENT HOSTS (HOST)+>
<!ELEMENT HOST (#PCDATA)>

<!ELEMENT SSL_VERIFY (#PCDATA)>
<!ELEMENT PORT (#PCDATA)>

<!ELEMENT IP_SET (IP|IP_RANGE)+>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT IP_RANGE (#PCDATA)>
<!ELEMENT LOGIN_TYPE (#PCDATA)>
<!ELEMENT NETWORK_ID (#PCDATA)>
<!ELEMENT CREATED (DATETIME, BY)>
<!ELEMENT BY (#PCDATA)>
<!ELEMENT LAST_MODIFIED (DATETIME)>
<!ELEMENT COMMENTS (#PCDATA)>

<!ELEMENT DIGITAL_VAULT (DIGITAL_VAULT_ID, DIGITAL_VAULT_TYPE,
```

```
DIGITAL_VAULT_TITLE, VAULT_FOLDER?, VAULT_FILE?, VAULT_SECRET_NAME?,
VAULT_SYSTEM_NAME?, VAULT_EP_NAME?, VAULT_EP_TYPE?, VAULT_EP_CONT?,
VAULT_NS_TYPE?, VAULT_NS_NAME?, VAULT_ACCOUNT_NAME?)>
<!ELEMENT DIGITAL_VAULT_ID (#PCDATA)>
<!ELEMENT DIGITAL_VAULT_TYPE (#PCDATA)>
<!ELEMENT DIGITAL_VAULT_TITLE (#PCDATA)>
<!ELEMENT VAULT_USERNAME (#PCDATA)>
<!ELEMENT VAULT_FOLDER (#PCDATA)>
<!ELEMENT VAULT_FILE (#PCDATA)>
<!ELEMENT VAULT_SECRET_NAME (#PCDATA)>
<!ELEMENT VAULT_SYSTEM_NAME (#PCDATA)>
<!ELEMENT VAULT_EP_NAME (#PCDATA)>
<!ELEMENT VAULT_EP_TYPE (#PCDATA)>
<!ELEMENT VAULT_EP_CONT (#PCDATA)>
<!ELEMENT VAULT_NS_TYPE (#PCDATA)>
<!ELEMENT VAULT_NS_NAME (#PCDATA)>
<!ELEMENT VAULT_ACCOUNT_NAME (#PCDATA)>

<!ELEMENT WARNING_LIST (WARNING+)>
<!ELEMENT WARNING (CODE?, TEXT, URL?, ID_SET?)>
<!ELEMENT CODE (#PCDATA)>
<!ELEMENT TEXT (#PCDATA)>
<!ELEMENT URL (#PCDATA)>
<!ELEMENT ID_SET (ID|ID_RANGE)+>
<!ELEMENT ID_RANGE (#PCDATA)>

<!ELEMENT GLOSSARY (USER_LIST?)>
<!ELEMENT USER_LIST (USER+)>
<!ELEMENT USER (USER_LOGIN, FIRST_NAME, LAST_NAME)>
<!ELEMENT FIRST_NAME (#PCDATA)>
<!ELEMENT LAST_NAME (#PCDATA)>
```

Create/Update VMware Authentication Record API

Now you can add vault support when creating/updating a VMware authentication record. We've added new input parameters per below.

API affected	/api/2.0/fo/auth/vmware/?action=create /api/2.0/fo/auth/vmware/?action=update
New or Updated API	Updated
DTD or XSD changes	No

New Input Parameters

Parameter	Description
login_type={**basic**|vault}	(Optional) By default a vault is not used to access credentials. Specify login_type=vault if a vault will be used.
vault_id={value}	(Required when login_type=vault, otherwise invalid)
vault_type={value}	(Required when login_type=vault, otherwise invalid) The vault type to use. Value can be one of: CA Access Control, Cyber-Ark PIM Suite, Cyber-Ark AIM, Lieberman ERPM, Quest Vault, Thycotic Secret Server, BeyondTrust PBPS
{vault definition}	Vault settings per vault type are provided in the Qualys API v2 User Guide, Chapter 9 Vault Support API.

Example

Create a VMware authentication record with vault support enabled.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -X "POST"
-d "action=create&ips=10.10.10.18&username=test&title=api-ca-access
control-vault&ssl_verify=all&login_type=vault&vault_type=Cyber-Ark PIM
Suite&vault_id=166655&folder=folder&file=file&hosts=www.test1.com&comment
s=my comments"
"https://qualysapi.qualys.com/api/2.0/fo/auth/vmware/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE BATCH_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
  <RESPONSE>
    <DATETIME>2018-01-23T17:57:03Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>177064</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Support for CertView scans (coming soon!)

We've made updates to the Scan API to support CertView scans when CertView GA is released (keep in mind CertView scans are not supported at this time).

Updated API endpoints:

Scan List API | Launch Scan API | Schedule Scan List API | Schedule Scan API | Add Asset API

Scan List API

Users can list CertView scans using the Scan List API - when CertView GA is released and CertView is enabled in the user's subscription.

API affected	/api/2.0/fo/scan/?action=list
New or Updated API	Updated
DTD or XSD changes	Yes

New input parameter

Parameter	Description
scan_type=certview	(Optional) List CertView scans only. **This option will be supported when CertView GA is released.**

Example

List CertView VM scans only, show asset group info, show option profile info:

API request:

```
curl -H "X-Requested-With: Curl Sample"
-b "QualysSession=71e6cda2a35d2cd404cddaf305ea0208; path=/api;
secure" "https://qualysapi.qualys.com/api/2.0/fo/scan/
?action=list&echo_request=1&show_ags=1&scan_type=certview"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SCAN_LIST_OUTPUT SYSTEM
"http://www.qualysapi.com/api/2.0/fo/scan/scan_list_output.dtd">
<SCAN_LIST_OUTPUT>
    <REQUEST>
        <DATETIME>2017-12-21T13:27:21Z</DATETIME>
        <USER_LOGIN>acme_ab</USER_LOGIN>
        <RESOURCE>http://wwww.qualysapi.com/api/2.0/fo/scan/</RESOURCE>
        <PARAM_LIST>
            <PARAM>
```

```
                <KEY>action</KEY>
                <VALUE>list</VALUE>
            </PARAM>
            <PARAM>
                <KEY>echo_request</KEY>
                <VALUE>1</VALUE>
            </PARAM>
            <PARAM>
                <KEY>show_ags</KEY>
                <VALUE>1</VALUE>
            </PARAM>
            <PARAM>
                <KEY>show_op</KEY>
                <VALUE>1</VALUE>
            </PARAM>
            <PARAM>
                <KEY>scan_type</KEY>
                <VALUE>certview</VALUE>
            </PARAM>
        </PARAM_LIST>
    </REQUEST>
    <RESPONSE>
        <DATETIME>2017-12-21T13:27:21Z</DATETIME>
        <SCAN_LIST>
            <SCAN>
                <REF>scan/1513832805.38958</REF>
                <TYPE>On-Demand</TYPE>
                <SCAN_TYPE>CertView</SCAN_TYPE>
                <TITLE>
                    <![CDATA[My Scan]]>
                </TITLE>
                <USER_LOGIN>acme_ab</USER_LOGIN>
                <LAUNCH_DATETIME>2017-12-21T05:06:45Z</LAUNCH_DATETIME>
                <DURATION>Pending</DURATION>
               <PROCESSING_PRIORITY>0 - No Priority</PROCESSING_PRIORITY>
                <PROCESSED>0</PROCESSED>
                <STATUS>
                    <STATE>Running</STATE>
                </STATUS>
                <TARGET>
                    <![CDATA[10.0.0.0-10.0.100.255, 10.1.0.0-
10.1.100.255]]>
                </TARGET>
                <ASSET_GROUP_TITLE_LIST>
                    <ASSET_GROUP_TITLE>
                        <![CDATA[Perf_Asset_300k_QWEB_12436_779]]>
                    </ASSET_GROUP_TITLE>
                </ASSET_GROUP_TITLE_LIST>
                <OPTION_PROFILE>
```

```
                    <TITLE>
                        <![CDATA[2008 SANS20 Options]]>
                    </TITLE>
                    <DEFAULT_FLAG>0</DEFAULT_FLAG>
                </OPTION_PROFILE>
            </SCAN>
        </SCAN_LIST>
    </RESPONSE>
</SCAN_LIST_OUTPUT>
```

DTD update

We've added new SCAN_TYPE tag that appears for CertView scan type only.

```
<!-- QUALYS SCAN_LIST_OUTPUT DTD -->
<!ELEMENT SCAN_LIST_OUTPUT (REQUEST?,RESPONSE)>

<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
POST_DATA?)>
<!ELEMENT DATETIME (#PCDATA)>
<!ELEMENT USER_LOGIN (#PCDATA)>
<!ELEMENT RESOURCE (#PCDATA)>
<!ELEMENT PARAM_LIST (PARAM+)>
<!ELEMENT PARAM (KEY, VALUE)>
<!ELEMENT KEY (#PCDATA)>
<!ELEMENT VALUE (#PCDATA)>
<!-- if returned, POST_DATA will be urlencoded -->
<!ELEMENT POST_DATA (#PCDATA)>

<!ELEMENT RESPONSE (DATETIME, SCAN_LIST?)>
<!ELEMENT SCAN_LIST (SCAN+)>
<!ELEMENT SCAN (ID?, REF, SCAN_TYPE?, TYPE, TITLE, USER_LOGIN,
LAUNCH_DATETIME, DURATION, PROCESSING_PRIORITY?, PROCESSED, STATUS?,
TARGET?, ASSET_GROUP_TITLE_LIST?, OPTION_PROFILE?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT REF (#PCDATA)>
<!ELEMENT SCAN_TYPE (#PCDATA)>
<!ELEMENT TYPE (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT LAUNCH_DATETIME (#PCDATA)>
<!ELEMENT DURATION (#PCDATA)>
<!ELEMENT PROCESSING_PRIORITY (#PCDATA)>
<!ELEMENT PROCESSED (#PCDATA)>
<!ELEMENT STATUS (STATE, SUB_STATE?)>
<!ELEMENT STATE (#PCDATA)>
<!ELEMENT SUB_STATE (#PCDATA)>
<!ELEMENT TARGET (#PCDATA)>
<!ELEMENT ASSET_GROUP_TITLE_LIST (ASSET_GROUP_TITLE+)>
<!ELEMENT ASSET_GROUP_TITLE (#PCDATA)>
```

```
<!ELEMENT OPTION_PROFILE (TITLE, DEFAULT_FLAG?)>
<!ELEMENT DEFAULT_FLAG (#PCDATA)>
<!-- EOF -->
```

Launch Scan API

Users can launch CertView scans - when CertView GA is released and CertView is enabled in the user's subscription.

API affected	/api/2.0/fo/schedule/scan/?action=launch
New or Updated API	Updated
DTD or XSD changes	No

New input parameters

Parameter	Description
scan_type=certview	(Optional) Launch a CertView type scan. **This option will be supported when CertView GA is released.**
fqdn={value}	(Optional) The target FQDN for a CertView type scan. For a CertView type scan you must specify at least one target i.e. IPs, asset groups or FQDNs. Multiple values are comma separated. **This option will be supported when CertView GA is released.**

Example

API request:

```
curl -H "X-Requested-With: Curl" -u "USERNAME:PASSWORD" -X "POST" -d
"action=launch&scan_title=My+CertView+Scan&ip=10.10.10.10&option_id=
43165&iscanner_name=scanner1&scan_type=certview&fqdn=myhost.fqdn.com"
"https://qualysapi.qualys.com/api/2.0/fo/scan/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-01-05T21:32:40Z</DATETIME>
    <TEXT>New vm scan launched</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>136992</VALUE>
      </ITEM>
```

```
      <ITEM>
        <KEY>REFERENCE</KEY>
        <VALUE>scan/1358285558.36992</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Schedule Scan List API

Users can list the scheduled CertView scans - when CertView GA is released and CertView is enabled in the user's subscription.

API affected	/api/2.0/fo/schedule/scan/?action=list
New or Updated API	Updated
DTD or XSD changes	Yes

New input parameters

Parameter	Description
scan_type=certview	(Optional) Launch a CertView type scan. **This option will be supported when CertView GA is released.**
fqdn={value}	(Optional) The target FQDN for a CertView type scan. For a CertView type scan you must specify at least one target i.e. IPs, asset groups or FQDNs. Multiple values are comma separated. **This option will be supported when CertView GA is released.**

Example

API request:

```
curl u "USERNAME:PASSWORD" -H "X-Requested-With: Curl"
"https://qualysapi.qualys.com/api/2.0/fo/schedule/scan/?action=list&id=17
3150&echo_request=1"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SCHEDULE_SCAN_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/schedule/scan/schedule_scan_list
_output.dtd">
<SCHEDULE_SCAN_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-01-10T11:36:12Z</DATETIME>
    <SCHEDULE_SCAN_LIST>
      <SCAN>
```

```
          <ID>173150</ID>
          <SCAN_TYPE>CertView</SCAN_TYPE>
          <ACTIVE>1</ACTIVE>
          <TITLE><![CDATA[API_Schedule_TitleV2_1515538934]]></TITLE>
          <USER_LOGIN>netwr_nd</USER_LOGIN>
          <TARGET><![CDATA[10.10.10.106]]></TARGET>
          <ISCANNER_NAME><![CDATA[External Scanner]]></ISCANNER_NAME>
          <USER_ENTERED_IPS>
            <RANGE>
              <START>10.10.10.106</START>
              <END>10.10.10.106</END>
            </RANGE>
          </USER_ENTERED_IPS>
          <OPTION_PROFILE>
            <TITLE><![CDATA[Initial Options]]></TITLE>
            <DEFAULT_FLAG>1</DEFAULT_FLAG>
          </OPTION_PROFILE>
          <PROCESSING_PRIORITY>0 - No Priority</PROCESSING_PRIORITY>
          <SCHEDULE>
            <DAILY frequency_days="1" />
            <START_DATE_UTC>2018-01-09T23:05:00Z</START_DATE_UTC>
            <START_HOUR>15</START_HOUR>
            <START_MINUTE>5</START_MINUTE>
            <END_AFTER_HOURS>1</END_AFTER_HOURS>
            <NEXTLAUNCH_UTC>2018-01-10T23:05:00</NEXTLAUNCH_UTC>
            <TIME_ZONE>
              <TIME_ZONE_CODE>US-CA</TIME_ZONE_CODE>
              <TIME_ZONE_DETAILS>(GMT-0800) United States:
America/Los_Angeles</TIME_ZONE_DETAILS>
            </TIME_ZONE>
            <DST_SELECTED>0</DST_SELECTED>
            <MAX_OCCURRENCE>1</MAX_OCCURRENCE>
          </SCHEDULE>
        </SCAN>
      </SCHEDULE_SCAN_LIST>
    </RESPONSE>
</SCHEDULE_SCAN_LIST_OUTPUT>
```

DTD update

We've added new SCAN_TYPE tag that appears for CertView scan type only.

```
<!-- QUALYS SCHEDULE_SCAN_LIST_OUTPUT DTD -->
<!-- $Revision$ -->
<!ELEMENT SCHEDULE_SCAN_LIST_OUTPUT (REQUEST?,RESPONSE)>

<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
POST_DATA?)>
<!ELEMENT DATETIME (#PCDATA)>
```

```
<!ELEMENT USER_LOGIN (#PCDATA)>
<!ELEMENT RESOURCE (#PCDATA)>
<!ELEMENT PARAM_LIST (PARAM+)>
<!ELEMENT PARAM (KEY, VALUE)>
<!ELEMENT KEY (#PCDATA)>
<!ELEMENT VALUE (#PCDATA)>
<!-- if returned, POST_DATA will be urlencoded -->
<!ELEMENT POST_DATA (#PCDATA)>

<!ELEMENT RESPONSE (DATETIME, SCHEDULE_SCAN_LIST?)>
<!ELEMENT SCHEDULE_SCAN_LIST (SCAN+)>
<!ELEMENT SCAN (ID, SCAN_TYPE?, ACTIVE, TITLE?, USER_LOGIN, TARGET,
NETWORK_ID?, ISCANNER_NAME?, EC2_INSTANCE?, CLOUD_DETAILS?,
ASSET_GROUP_TITLE_LIST?, ASSET_TAGS?, EXCLUDE_IP_PER_SCAN?,
USER_ENTERED_IPS?, OPTION_PROFILE?, PROCESSING_PRIORITY?, SCHEDULE,
NOTIFICATIONS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT ACTIVE (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT TARGET (#PCDATA)>
<!ELEMENT NETWORK_ID (#PCDATA)>
<!ELEMENT ISCANNER_NAME (#PCDATA)>
<!ELEMENT EC2_INSTANCE (CONNECTOR_UUID, EC2_ENDPOINT, EC2_ONLY_CLASSIC?)>
<!ELEMENT CONNECTOR_UUID (#PCDATA)>
<!ELEMENT EC2_ENDPOINT (#PCDATA)>
<!ELEMENT EC2_ONLY_CLASSIC (#PCDATA)>

<!ELEMENT CLOUD_DETAILS (PROVIDER, CONNECTOR, SCAN_TYPE, CLOUD_TARGET)>
<!ELEMENT PROVIDER (#PCDATA)>
<!ELEMENT CONNECTOR (ID?, UUID, NAME)>
<!ELEMENT UUID (#PCDATA)>
<!ELEMENT NAME (#PCDATA)>
<!ELEMENT SCAN_TYPE (#PCDATA)>
<!ELEMENT CLOUD_TARGET (PLATFORM, REGION?, VPC_SCOPE, VPC_LIST?)>
<!ELEMENT PLATFORM (#PCDATA)>
<!ELEMENT REGION (UUID, CODE?, NAME?)>
<!ELEMENT CODE (#PCDATA)>
<!ELEMENT VPC_SCOPE (#PCDATA)>
<!ELEMENT VPC_LIST (VPC+)>
<!ELEMENT VPC (UUID)>
...
<!-- EOF -->
```

Schedule Scan API

Users can schedule CertView scans - when CertView GA is released and CertView is enabled in the user's subscription.

API affected	/api/2.0/fo/schedule/scan/?action=create
New or Updated API	Updated
DTD or XSD changes	No

New input parameters

Parameter	Description
scan_type=certview	(Optional) Launch a CertView type scan. **This option will be supported when CertView GA is released.**
fqdn={value}	(Optional) The target FQDN for a CertView type scan. For a CertView type scan you must specify at least one target i.e. IPs, asset groups or FQDNs. Multiple values are comma separated. **This option will be supported when CertView GA is released.**

Example

API request:

```
curl -u "USERNAME:PASSWD" -H "X-Requested-With: curl" -X "POST" -d
"scan_title=My+CertView+Scan+Schedule&active=1&option_id=3456&target_from
=tags&tag_set_include=tag1,tag2,tag3&iscanner_name=scanner1&scan_type=cer
tview&occurrence=daily&frequency_days=5&time_zone_code=US-
CA&observe_dst=yes&start_hour=14&start_minute=0"
"https://qualysapi.qualys.com/api/2.0/fo/schedule/scan/?action=create"
```

XML output:

```
?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-01-05T21:32:40Z</DATETIME>
    <TEXT>New scan scheduled successfully</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>136992</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Add Asset API

Users can add IP assets to their CertView license using the Add Asset API - when CertView GA is released and CertView is enabled in the user's subscription.

API affected	/api/2.0/fo/asset/ip/?action=add
New or Updated API	Updated
DTD or XSD changes	Yes

New input parameter

Parameter	Description
enable_certview={**0**|1}	(Optional) Set to 1 to add IPs to your CertView license. By default IPs are not added to your CertView license. **This option will be supported when CertView GA is released.**

Example

Add IP assets to your CertView license:

API request (POSTED raw data in CSV format):

```
curl -H "X-Requested-With: Curl" -H "Content-Type:text/csv"
-u "USERNAME:PASSWORD" --data-binary @ips_list.csv
"https://qualysapi.qualys.com/api/2.0/fo/asset/ip/?action=add&enable_vm=1
&enable_certview=1&tracking_method=IP&owner=quays_es1"
```

API request ("ips" parameter):

```
curl -H "X-Requested-With: demo" -u "USERNAME:PASSWORD" -X "POST"
-d "action=add&enable_vm=1&enable_certview=1&ips=10.10.10.1,10.10.10.10-
10.10.10.20,10.10.10.200"
"https://qualysapi.qualys.com/api/2.0/fo/asset/ip/"
```

XML output:

Text response lists CertView.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2013-08-07T01:21:03Z</DATETIME>
    <TEXT>IPs successfully added to CertView</TEXT>
</RESPONSE>
</SIMPLE_RETURN>
```