Qualys Cloud Platform (VM, SCA, PC) v8.x

API Release Notes

Version 8.15

August 17, 2018

This new version of the Qualys Cloud Platform (VM, SCA, PC) includes improvements to the Qualys API. You'll find all the details in our user guides, available at the time of release. 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.

Posture Profile API - DTD Change for show_remediation_info

APIs affected	/api/2.0/fo/compliance/posture/info/
New or Updated API	Updated (DTD change only)
DTD or XSD changes	Yes

In the Posture Profile Information DTD the V value in element <!ELEMENT TP (LABEL, V+)> replaced with <!ELEMENT TP (LABEL, V*)> to ensure that the validation does not fail. This is an optional value.

DTD update:

DTD: <base_url>/api/2.0/fo/compliance/posture/info/posture_info_list_output.dtd

```
...
<!ATTLIST DPV lastUpdated CDATA #IMPLIED>

<!ELEMENT CAUSE_OF_FAILURE (DIRECTORY_FIM_UDC, UNEXPECTED?, MISSING?,
ADDED_DIRECTORIES?, REMOVED_DIRECTORIES?,
PERMISSON_CHANGED_DIRECTORIES?, CONTENT_CHANGED_DIRECTORIES?)>
<!ELEMENT DIRECTORY_FIM_UDC (#PCDATA)>
<!ELEMENT UNEXPECTED (V*)>
<!ELEMENT MISSING (V*)>
<!ATTLIST MISSING logic CDATA #FIXED "OR">
<!ELEMENT ADDED_DIRECTORIES (V*)>
<!ELEMENT REMOVED_DIRECTORIES (V*)>
<!ELEMENT PERMISSON_CHANGED_DIRECTORIES (V*)>
<!ELEMENT CONTENT_CHANGED_DIRECTORIES (V*)>

<!ELEMENT LABEL (#PCDATA)>
...
<!ELEMENT DPD_LIST (DPD+)>
<!ELEMENT DPD (LABEL, ID?, NAME?, DESC)>
<!ELEMENT DESC (#PCDATA)>

<!ELEMENT TP_LIST (TP+)>
<!ELEMENT TP (LABEL, V*)>

<!ELEMENT FV_LIST (FV+)>
<!ELEMENT FV (LABEL, V*)>

<!ELEMENT TM_LIST (TM+)>
<!ELEMENT TM (LABEL, PAIR+)>
<!ELEMENT PAIR (K, V)>
<!ELEMENT K (#PCDATA)>
...
```

Posture Profile API - New Parameter to Show Cause of Failure

APIs affected	/api/2.0/fo/compliance/posture/info/
New or Updated API	Updated (DTD change only)
DTD or XSD changes	Yes

We added a new parameter to the Posture Profile API to show the cause of failure of Directory Integrity Monitoring UDCs (user defined controls).

Input Parameter:

Parameter	Description
cause_of_failure={**0**|1}	(Optional) Set flag to 1 to display the cause of failure of Directory Integrity Monitoring UDCs (user defined controls). When set to 1 and Directory Integrity Monitoring UDC control failed assessment the cause of failure info is shown in XML response, i.e. added, removed directories, directories where content changed, permissions changed etc. When set to 0 or unspecified, cause of failure is not displayed for these UCDs.

Sample to display cause of failure

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X demo 2'-D
headers.15
"https://qualysapi.qualys.com/api/2.0/fo/compliance/posture/info/?action=
list&policy_id=53054&details=All&ips=10.10.30.112&cause_of_failure=1"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE POSTURE_INFO_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/compliance/posture/info/posture_
info_list_output.dtd">
<POSTURE_INFO_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-08-15T23:14:02Z</DATETIME>
...

      <INFO>
        <ID>4148553</ID>
        <HOST_ID>621618</HOST_ID>
        <CONTROL_ID>1072</CONTROL_ID>
        <TECHNOLOGY_ID>37</TECHNOLOGY_ID>
        <INSTANCE></INSTANCE>
```

```
        <STATUS>Failed</STATUS>
        <POSTURE_MODIFIED_DATE>2018-05-
01T19:19:11Z</POSTURE_MODIFIED_DATE>
        <EVIDENCE>
          <BOOLEAN_EXPR><![CDATA[(:dp_5 in #fv_3 or :dp_5 == $tp_1
)]]></BOOLEAN_EXPR>
          <DPV_LIST>
            <DPV lastUpdated="2018-05-02T21:05:10Z">
              <LABEL>:dp_5</LABEL>
              <V><![CDATA[1]]></V>
            </DPV>
          </DPV_LIST>
        </EVIDENCE>
        <CAUSE_OF_FAILURE>
          <DIRECTORY_FIM_UDC>0</DIRECTORY_FIM_UDC>
          <UNEXPECTED>
            <V><![CDATA[1]]></V>
          </UNEXPECTED>
          <MISSING logic="OR">
            <V><![CDATA[0]]></V>
          </MISSING>
        </CAUSE_OF_FAILURE>
      </INFO>
...

</RESPONSE>
</POSTURE_INFO_LIST_OUTPUT>
```

Sample to not display cause of failure

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X demo 2'-D
headers.15
"https://qualysapi.qualys.com/api/2.0/fo/compliance/posture/info/?action=
list&policy_id=53054&details=All&ips=10.10.30.112&cause_of_failure=0"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE POSTURE_INFO_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/compliance/posture/info/posture_
info_list_output.dtd">
<POSTURE_INFO_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-08-15T23:14:02Z</DATETIME>
...
```

```
      <INFO>
        <ID>4148553</ID>
        <HOST_ID>621618</HOST_ID>
        <CONTROL_ID>1072</CONTROL_ID>
        <TECHNOLOGY_ID>37</TECHNOLOGY_ID>
        <INSTANCE></INSTANCE>
        <STATUS>Failed</STATUS>
        <POSTURE_MODIFIED_DATE>2018-05-
01T19:19:11Z</POSTURE_MODIFIED_DATE>
        <EVIDENCE>
          <BOOLEAN_EXPR><![CDATA[(:dp_5 in #fv_3 or :dp_5 == $tp_1
)]]></BOOLEAN_EXPR>
          <DPV_LIST>
            <DPV lastUpdated="2018-05-02T21:05:10Z">
              <LABEL>:dp_5</LABEL>
              <V><![CDATA[1]]></V>
            </DPV>
          </DPV_LIST>
        </EVIDENCE>
      </INFO>
...

</RESPONSE>
</POSTURE_INFO_LIST_OUTPUT>
```

New EC2 Information in Host Based Report

APIs affected	/api/2.0/fo/report
New or Updated API	Updated
DTD or XSD changes	Yes

You will now see three new fields: Account ID, Region Code and Subnet ID in host based reports when you create your report using the Scan or PCI Scan template with the EC2 Related Information option checked.

Example: Download the report

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=fetch&id=409313&echo_request=1"
"https://qualysapi.qualys.com/api/2.0/fo/report/”
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE ASSET_DATA_REPORT SYSTEM
"https://qualysapi.qualys.com/asset_data_report.dtd">
<ASSET_DATA_REPORT><ASSET_DATA_REPORT>
  <HEADER>
    <COMPANY><![CDATA[AFCO]]></COMPANY>
    <USERNAME>user_john</USERNAME>
    <GENERATION_DATETIME>2018-06-22T04:48:46Z</GENERATION_DATETIME>
    <TEMPLATE><![CDATA[8.14 Template with EC2 options
checked]]></TEMPLATE>
        ...
        <EC2_INFO>
        <PUBLIC_DNS_NAME><![CDATA[ec2-54-152-127-191.compute-
1.amazonaws.com]]></PUBLIC_DNS_NAME>
        <IMAGE_ID><![CDATA[ami-467ca739]]></IMAGE_ID>
        <VPC_ID><![CDATA[vpc-f0243088]]></VPC_ID>
        <INSTANCE_STATE><![CDATA[RUNNING]]></INSTANCE_STATE>
        <PRIVATE_DNS_NAME><![CDATA[ip-10-0-0-
95.ec2.internal]]></PRIVATE_DNS_NAME>
        <INSTANCE_TYPE><![CDATA[t2.large]]></INSTANCE_TYPE>
        <ACCOUNT_ID><![CDATA[205767712438]]></ACCOUNT_ID>
        <REGION_CODE><![CDATA[us-east-1]]></REGION_CODE>
        <SUBNET_ID><![CDATA[subnet-e33a58be]]></SUBNET_ID>
      </EC2_INFO>
    ...
  </HOST_LIST>
  <GLOSSARY>
```

```
    <VULN_DETAILS_LIST>
      <VULN_DETAILS id="qid_6">
        <QID id="qid_6">6</QID>
        <TITLE><![CDATA[DNS Host Name]]></TITLE>
  ...
</TEMPLATE_DETAILS>
  </APPENDICES>
</ASSET_DATA_REPORT>
```

DTD Update

We updated the Asset Data Report DTD (asset_data_report.dtd) to include the ACCOUNT_ID, REGION_CODE and SUBNET_ID elements.

```
<!ELEMENT ASSET_DATA_REPORT (ERROR | (HEADER, RISK_SCORE_PER_HOST?,
HOST_LIST?, GLOSSARY?, NON_RUNNING_KERNELS?, APPENDICES?))>

<!ELEMENT ERROR (#PCDATA)*>
<!ATTLIST ERROR number CDATA #IMPLIED>
...
<!ELEMENT ACCOUNT_ID (#PCDATA)>
<!ELEMENT REGION_CODE (#PCDATA)>
<!ELEMENT SUBNET_ID (#PCDATA)>
...
<!ELEMENT NON_RUNNING_KERNEL (NRK_QID*, IP*, SEVERITY*)>
<!ELEMENT NRK_QID (#PCDATA)>
<!-- EOF -->
```

New MariaDB Authentication API

API affected	/api/2.0/fo/auth/
New or Updated API	Updated
DTD or XSD changes	Yes
API affected	/api/2.0/fo/auth/mariadb/
New or Updated API	New
DTD or XSD changes	New

MariaDB authentication is now supported for compliance scans. The new MariaDB Authentication API (<baseurl>/api/2.0/fo/auth/mariadb/) lets you list, create, update and delete MariaDB authentication records. User permissions for this API are the same as other authentication record APIs.

List all record types

Use the Authentication Record List API (/api/2.0/fo/auth/?action=list) to list records. You'll see <AUTH_MARIADB_IDS> in the output when you have MariaDB records.

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_RECORDS_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/auth_records.dtd">
<AUTH_RECORDS_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-07-17T21:58:50Z</DATETIME>
    <AUTH_RECORDS>
      <AUTH_UNIX_IDS>
        <ID_SET>
          <ID>195368</ID>
          <ID>195385</ID>
        </ID_SET>
      </AUTH_UNIX_IDS>
      <AUTH_WINDOWS_IDS>
        <ID_SET>
          <ID>243033</ID>
```

```
            <ID_RANGE>243418-243420</ID_RANGE>
            <ID>246432</ID>
          </ID_SET>
        </AUTH_WINDOWS_IDS>
        <AUTH_MARIADB_IDS>
          <ID_SET>
            <ID>284866</ID>
          </ID_SET>
        </AUTH_MARIADB_IDS>
      </AUTH_RECORDS>
    </RESPONSE>
</AUTH_RECORDS_OUTPUT>
```

Updated DTD:

<baseurl>/api/2.0/fo/auth/auth_records.dtd

The element AUTH_MARIADB_IDS was added to identify MariaDB record IDs.

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

<!ELEMENT AUTH_RECORDS_OUTPUT (REQUEST?, RESPONSE)>
...
<!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?,
AUTH_MARIADB_IDS?)>
...
<!ELEMENT AUTH_MONGODB_IDS (ID_SET)>
<!ELEMENT AUTH_PALO_ALTO_FIREWALL_IDS (ID_SET)>
<!ELEMENT AUTH_VCENTER_IDS (ID_SET)>
<!ELEMENT AUTH_MARIADB_IDS (ID_SET)>
...
```

List MariaDB records

Use the new MariaDB Authentication Record List API (/api/2.0/fo/auth/mariadb/?action=list) to list MariaDB records.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=list"
"https://qualysapi.qualys.com/api/2.0/fo/auth/mariadb/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_MARIADB_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/mariadb/auth_mariadb
_list_output.dtd">
<AUTH_MARIADB_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-07-17T21:57:32Z</DATETIME>
    <AUTH_MARIADB_LIST>
      <AUTH_MARIADB>
        <ID>284866</ID>
        <TITLE><![CDATA[MariaDB_Auth1]]></TITLE>
        <USERNAME><![CDATA[root]]></USERNAME>
        <DATABASE><![CDATA[mariadb]]></DATABASE>
        <PORT>22</PORT>
        <IP_SET>
          <IP>10.10.31.86</IP>
        </IP_SET>
        <LOGIN_TYPE><![CDATA[basic]]></LOGIN_TYPE>
        <SSL_VERIFY>false</SSL_VERIFY>
        <WINDOWS_CONF_FILE><![CDATA[]]></WINDOWS_CONF_FILE>
        <UNIX_CONF_FILE><![CDATA[/etc/my.cnf]]></UNIX_CONF_FILE>
        <NETWORK_ID>0</NETWORK_ID>
        <CREATED>
          <DATETIME>2018-07-17T21:56:47Z</DATETIME>
          <BY>seenu_yn</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2018-07-17T21:56:47Z</DATETIME>
        </LAST_MODIFIED>
      </AUTH_MARIADB>
    </AUTH_MARIADB_LIST>
```

```
  </RESPONSE>
</AUTH_MARIADB_LIST_OUTPUT>
```

New DTD:

<baseurl>/api/2.0/fo/auth/mariadb/auth_mariadb_list_output.dtd

```
<!-- QUALYS AUTH_MARIADB_LIST_OUTPUT DTD -->
<!-- $Revision: 68724 $ -->
<!ELEMENT AUTH_MARIADB_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_MARIADB_LIST|ID_SET)?,
WARNING_LIST?, GLOSSARY?)>
<!ELEMENT AUTH_MARIADB_LIST (AUTH_MARIADB+)>

<!ELEMENT AUTH_MARIADB (ID, TITLE, USERNAME, DATABASE, PORT,
HOSTS?, IP_SET?, LOGIN_TYPE?, DIGITAL_VAULT?, SSL_VERIFY,
WINDOWS_CONF_FILE, UNIX_CONF_FILE, CLIENT_CERT?, CLIENT_KEY?,
NETWORK_ID?, CREATED, LAST_MODIFIED, COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT DATABASE (#PCDATA)>
<!ELEMENT PORT (#PCDATA)>
<!ELEMENT SSL_VERIFY (#PCDATA)>
<!ELEMENT WINDOWS_CONF_FILE (#PCDATA)>
<!ELEMENT UNIX_CONF_FILE (#PCDATA)>
<!ELEMENT CLIENT_CERT (#PCDATA)>
<!ELEMENT CLIENT_KEY (#PCDATA)>
```

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

<!ELEMENT IP_SET (IP|IP_RANGE)+>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT IP_RANGE (#PCDATA)>

<!ELEMENT LOGIN_TYPE (#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 NETWORK_ID (#PCDATA)>

<!ELEMENT CREATED (DATETIME, BY)>
<!ELEMENT BY (#PCDATA)>
<!ELEMENT LAST_MODIFIED (DATETIME)>
<!ELEMENT COMMENTS (#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)>

        <!-- EOF -->
```

Create/Update MariaDB record

Use these parameters to create or update a MariaDB record. For an update request, all parameters are optional except "ids" which is required.

Parameter	Description
action={action}	(Required) Specify create, update, delete (using POST) or list (using GET or POST).
echo_request={**0**|1}	(Optional) Specify 1 to view (echo) input parameters in the XML output. By default these are not included.
ids={value}	(Required to update or delete record) Record IDs to update/delete. Specify record IDs and/or ID ranges (for example, 1359-1407). Multiple entries are comma separated.
title={value}	(Required to create record) A title for the record. The title must be unique. Maximum 255 characters (ascii).
comments={value}	(Optional to create or update record) User defined comments. Maximum of 1999 characters.
MariaDB	
ssl_verify={**0**|1}	(Optional to create or update record, and valid for server that supports SSL) Specify 1 for a complete SSL certificate validation. - If unspecified (or ssl_verify=0), Qualys scanners authenticate with MySQL Servers that don't use SSL or MariaDB servers that use SSL. However, in the SSL case, the server SSL certificate verification will be skipped. - If ssl_verify=1, the Qualys scanners will only send a login request after verifying that a connection the MariaDB server uses SSL, the server SSL certificate is valid and matches the scanned host.
hosts={value}	(Optional to create or update record) A list of FQDNs for the hosts that correspond to all host IP addresses on which a custom SSL certificate signed by a trusted root CA is installed. Multiple hosts are comma separated.

Parameter	Description
database={value}	(Required to create record, optional to update record) The database name to authenticate to. Specify a valid MariaDB database name.
port={value}	(Required to create record, optional to update record) The port the database name is running on. The default is 3306.
windows_config_file={value}	(Optional to create or update record) The path to the Windows mariadb config file. Access to this config file is required to run certain checks on Windows hosts.
	Note: You must include one or both of these parameters in a create request: windows_config_file and unix_config_file.
unix_config_file={value}	(Optional to create or update record) The path to the Unix mariadb config file. Access to this config file is required to run certain checks on Unix hosts.
	Note: You must include one or both of these parameters in a create request: windows_config_file and unix_config_file.
client_cert={value}	(Optional to create or update record) PEM-encoded X.509 certificate. Specify if certificate authentication is required by your server to establish an SSL connection.
client_key={value}	(Optional to create or update record) PEM-encoded RSA private key. Specify if certificate authentication is required by your server to establish an SSL connection.
Login credentials	
login_type={**basic**|vault}	(Optional) The login type is basic by default. You can choose vault (for vault based authentication).
username={value}	(Required to create record, optional to update record) The username to be used for authentication to MariaDB server.
password={value}	(Required to create record, optional to update record) The password to be used for authentication to MariaDB server.
Vault	
vault_type={value}	(Required to create record when login_type=vault) The vault type to be used for authentication.
vault_id={value}	(Required to create record when login_type=vault and you want to retrieve private key from vault) The vault ID where you want to retrieve the private key from. Certain vaults support this capability.
{vault parameters}	(Required to create record when login_type=vault) Vault specific parameters required depend on the vault type you've selected. See the API v2 User Guide for vault parameters.

Parameter	Description
Target Hosts	
ips={value}	(Required to create record) The IP address(es) the server will log into using the record's credentials. Multiple entries are comma separated. (Optional to update record) IPs specified will overwrite existing IPs in the record, and existing IPs will be removed.
add_ips={value}	(Optional to update record) Add IPs to the IPs list for this record. Multiple IPs/ranges are comma separated.
remove_ips={value}	(Optional to update record) IPs to be removed from your record. You may enter a combination of IPs and ranges. Multiple entries are comma separated. This parameter and the ips parameter cannot be specified in the same request.
network_id={value}	(Optional and valid when the networks feature is enabled) The network ID for the record.

Example: Create MariaDB record (with basic login)

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl sample" -d
"action=create&title=MariaDB_Auth1&username=root&password=abc123&i
ps=10.10.31.86&echo_request=0&unix_config_file=/etc/my.cnf&port=22
&database=mariadb"
"https://qualysapi.qualys.com/api/2.0/fo/auth/mariadb/"
```

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>2018-07-17T21:56:47Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>284866</ID>
        </ID_SET>
      </BATCH>
```

```
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Example: Update MariaDB record

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl sample" -d
"action=update&ids=42002&add_ips=10.0.0.2-10.0.0.5"
"https://qualysapi.qualys.com/api/2.0/fo/auth/mariadb/"
```

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>2018-06-25T22:14:33Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Updated</TEXT>
                <ID_SET>
                    <ID>42001</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

Delete MariaDB records

Use these parameters to delete records.

Parameter	Description
action=delete	(Required) POST method may be used.
ids={value}	(Required) MariaDB authentication record IDs for the records you want to delete. Multiple records are comma separated.

Example: Delete MariaDB records

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl Sample" -d
"action=delete&ids=125708,125709"
"https://qualysapi.qualys.com/api/2.0/fo/auth/mariadb/"
```

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>2018-05-23T20:06:30Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Deleted</TEXT>
                <ID_SET>
                    <ID>32011</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

New JBoss Server Authentication API

APIs affected	/api/2.0/fo/auth/jboss
New or Updated API	New
DTD or XSD changes	Updated

We have now added a new API to support JBoss Server Authentication. Using the JBoss Server API (.../api/2.0/fo/auth/jboss) you can create, update, list, and delete JBoss Server records.

Supported technologies:

Windows - WildFly/JBoss EAP

Unix - WildFly/JBoss EAP

Input Parameters

Parameter	Description
action={action}	(Required) Specify create, update, delete (using POST) or list (using GET or POST).
ids={value}	(Required) Specify a single or comma separated valid JBoss type auth record ID(s).
title={value}	(Required to create record) A title for the record. The title must be unique.
comment={value}	(Optional to create or update record) User defined comments.
Windows platform	
windows_working_mode={value}	(Optional) Input values should be standalone_mode or domain_controller_mode.
windows_home_path={value}	Required if windows working mode is selected.
windows_base_path={value}	Required if windows working mode is selected.
windows_conf_dir_path={value}	Required if windows working mode is selected.
windows_conf_file_path={value}	Required if windows working mode is selected.
windows_conf_host_file_path={value}	Required if selected Windows working mode is domain controller.

Parameter	Description
Unix platform	
unix_working_mode={value}	(Optional) Input values should be standalone_mode or domain_controller_mode.
unix_home_path={value}	Required if Unix working mode is selected.
unix_base_path={value}	Required if Unix working mode is selected.
unix_conf_dir_path={value}	Required if Unix working mode is selected.
unix_conf_file_path={value}	Required if Unix working mode is selected.
unix_conf_host_file_path={value}	Required if selected Unix working mode is domain controller.
Target Hosts	
ips={value}	(Required to create record) The IP address(es) the server will log into using the record's credentials. Multiple entries are comma separated. (Optional to update record) IPs specified will overwrite existing IPs in the record, and existing IPs will be removed.
add_ips={value}	(Optional and valid only to update record) IPs to be added to an existing record. You may enter a combination of IPs and IP ranges. Multiple entries are comma separated.
remove_ips={value}	(Optional and valid to update record) IPs to be removed from your record. You may enter a combination of IPs and ranges. Multiple entries are comma separated.
network_id={value}	(Optional to create or update record, and valid when the networks feature is enabled) The network ID for the record.

Sample - Create JBoss Server record

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X POST
"action=create&title=jbos_rec&windows_working_mode=standalone_mode&window
s_base_path=c:\&windows_home_path=c:\&windows_conf_file_path=c:\&windows_
conf_dir_path=c:\&comment=record creation&ips=10.10.10.224"
"https://qualysapi.qualys.com/api/2.0/fo/auth/jboss/"
```

XML output:

```
<!DOCTYPE BATCH_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
```

```
  <RESPONSE>
    <DATETIME>2018-08-03T10:42:32Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>296004</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Sample - Update JBoss Server record

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X POST
"action=update&ids=296004&windows_working_mode=domain_controller_mode&win
dows_base_path=c:\&windows_home_path=c:\&windows_conf_file_path=c:\&windo
ws_conf_dir_path=c:\&comment=record
creation&windows_conf_host_file_path=c:\&ips=10.10.10.224"
"https://qualysapi.qualys.com/api/2.0/fo/auth/jboss/"
```

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>2018-08-03T10:43:58Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Updated</TEXT>
        <ID_SET>
          <ID>296004</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

List JBoss Server Record

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -d
"action=list&ids=296004"
```

```
"https://qualysapi.qualys.com/api/2.0/fo/auth/jboss/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_JBOSS_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/jboss/auth_jboss_list_outpu
t.dtd">
<AUTH_JBOSS_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-08-03T10:44:39Z</DATETIME>
    <AUTH_JBOSS_LIST>
      <AUTH_JBOSS>
        <ID>296004</ID>
        <TITLE><![CDATA[jbos_rec]]></TITLE>
        <IP_SET>
          <IP>10.10.10.224</IP>
        </IP_SET>
        <WINDOWS>
          <HOME_PATH><![CDATA[c:\]]></HOME_PATH>
          <DOMAIN_MODE><![CDATA[true]]></DOMAIN_MODE>
          <BASE_PATH><![CDATA[c:\]]></BASE_PATH>
          <CONF_DIR_PATH><![CDATA[c:\]]></CONF_DIR_PATH>
          <CONF_FILE_PATH><![CDATA[c:\]]></CONF_FILE_PATH>
          <CONF_HOST_FILE_PATH><![CDATA[c:\]]></CONF_HOST_FILE_PATH>
        </WINDOWS>
        <NETWORK_ID>0</NETWORK_ID>
        <CREATED>
          <DATETIME>2018-08-03T10:42:32Z</DATETIME>
          <BY>abc_pk</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2018-08-03T10:43:58Z</DATETIME>
        </LAST_MODIFIED>
        <COMMENTS><![CDATA[record creation]]></COMMENTS>
      </AUTH_JBOSS>
    </AUTH_JBOSS_LIST>
  </RESPONSE>
</AUTH_JBOSS_LIST_OUTPUT>
```

DTD:

<baseurl>/api/2.0/fo/auth/jboss/auth_jboss_list_output.dtd

```
<!-- QUALYS AUTH_JBOSS_LIST_OUTPUT DTD -->
<!ELEMENT AUTH_JBOSS_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_JBOSS_LIST|ID_SET)?, WARNING_LIST?,
GLOSSARY?)>
<!ELEMENT AUTH_JBOSS_LIST (AUTH_JBOSS+)>

<!ELEMENT AUTH_JBOSS (ID, TITLE, WINDOWS?, UNIX?, IP_SET, NETWORK_ID?,
CREATED, LAST_MODIFIED, COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT WINDOWS (HOME_PATH?, DOMAIN_MODE?, BASE_PATH?, CONF_DIR_PATH?,
CONF_FILE_PATH?, CONF_HOST_FILE_PATH?)>
<!ELEMENT HOME_PATH (#PCDATA)>
<!ELEMENT DOMAIN_MODE (#PCDATA)>
<!ELEMENT BASE_PATH (#PCDATA)>
<!ELEMENT CONF_DIR_PATH (#PCDATA)>
<!ELEMENT CONF_FILE_PATH (#PCDATA)>
<!ELEMENT CONF_HOST_FILE_PATH (#PCDATA)>
<!ELEMENT UNIX (HOME_PATH?, DOMAIN_MODE?, BASE_PATH?, CONF_DIR_PATH?,
CONF_FILE_PATH?, CONF_HOST_FILE_PATH?)>
<!ELEMENT IP_SET (IP|IP_RANGE)+>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT IP_RANGE (#PCDATA)>
<!ELEMENT NETWORK_ID (#PCDATA)>
<!ELEMENT CREATED (DATETIME, BY)>
<!ELEMENT BY (#PCDATA)>
<!ELEMENT LAST_MODIFIED (DATETIME)>
<!ELEMENT COMMENTS (#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)>
<!-- EOF -->
```

Delete JBoss Server Record

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -d
"action=delete&ids=291727"
"https://qualysapi.qualys.com/api/2.0/fo/auth/jboss/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE BATCH_RETURN SYSTEM
"https://qualysapi.p04.eng.sjc01.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
  <RESPONSE>
    <DATETIME>2018-08-03T11:16:18Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Deleted</TEXT>
        <ID_SET>
          <ID>291727</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

MySQL DB Authentication API - Support for Vaults

APIs affected	/api/2.0/fo/auth/mysql/
New or Updated API	Updated
DTD or XSD changes	Updated

Now API users can configure MySQL authentication records to use vaults to access credentials used for authentication. Vaults are already supported for MySQL authentication in the UI.

Input Parameters

We have added new input parameters for adding vault information.

Parameter	Description
login_type={**basic**|vault}	(Optional) The login type is basic by default. Specify login_type=vault to use an authentication vault.
vault_id={value}	(Required only when action=create and login_type= vault) The ID of the vault you want to use.
vault_type={value}	(Required only when action=create and login_type= vault) The vault to be used for authentication. For MySQL authentication, valid values are: BeyondTrust PBPS, CyberArk AIM, CyberArk PIM Suite, Quest Vault, Thycotic Secret Server
{vault parameters}	(Required only when action=create and login_type=vault) Vault specific parameters required depend on the vault type you've selected. See the Qualys API (VM, SCA, PC) User Guide, Chapter 6 Vault Support API.

Some parameters are now required

The database and port parameters are now required when creating a new record (for basic and vault login types). You must also include at least one of these parameters in your create request: windows_config_file and unix_config_file.

Example: List MySQL record

You'll now see vault information in the XML output when you list MySQL authentication records with vaults.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=list&ids=284212"
"https://qualysapi.qualys.com/api/2.0/fo/auth/mysql/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_MYSQL_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/mysql/auth_mysql_list_outpu
t.dtd">
<AUTH_MYSQL_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-07-17T17:09:18Z</DATETIME>
    <AUTH_MYSQL_LIST>
      <AUTH_MYSQL>
        <ID>284212</ID>
        <TITLE><![CDATA[api-Thycotic Secret Server_tss]]></TITLE>
        <USERNAME><![CDATA[test_tss]]></USERNAME>
        <DATABASE><![CDATA[mysql]]></DATABASE>
        <PORT>22</PORT>
        <HOSTS>
          <HOST><![CDATA[www.test.com]]></HOST>
        </HOSTS>
        <IP_SET>
          <IP>10.10.10.181</IP>
        </IP_SET>
        <LOGIN_TYPE><![CDATA[vault]]></LOGIN_TYPE>
        <DIGITAL_VAULT>
          <DIGITAL_VAULT_ID><![CDATA[166638]]></DIGITAL_VAULT_ID>
          <DIGITAL_VAULT_TYPE><![CDATA[Thycotic Secret
Server]]></DIGITAL_VAULT_TYPE>
          <DIGITAL_VAULT_TITLE><![CDATA[3_Secret
Server]]></DIGITAL_VAULT_TITLE>
          <VAULT_SECRET_NAME><![CDATA[secret]]></VAULT_SECRET_NAME>
        </DIGITAL_VAULT>
        <SSL_VERIFY>true</SSL_VERIFY>

<WINDOWS_CONF_FILE><![CDATA[c:\mysql\myu.ini]]></WINDOWS_CONF_FILE>
        <UNIX_CONF_FILE><![CDATA[]]></UNIX_CONF_FILE>
        <NETWORK_ID>0</NETWORK_ID>
        <CREATED>
          <DATETIME>2018-07-16T21:53:55Z</DATETIME>
          <BY>seenu_yn</BY>
```

```
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2018-07-16T21:55:05Z</DATETIME>
        </LAST_MODIFIED>
        <COMMENTS><![CDATA[test comments]]></COMMENTS>
      </AUTH_MYSQL>
    </AUTH_MYSQL_LIST>
  </RESPONSE>
</AUTH_MYSQL_LIST_OUTPUT>
```

Example: Create new MySQL record using vault

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=create&ips=10.10.10.181&username=USERNAME&title=NewMySQLRecord&ss
l_verify=1&hosts=www.test.com&login_type=vault&vault_type=Thycotic Secret
Server&vault_id=166638&secret_name=secret&comments=test
comments&port=22&database=mysql&windows_config_file=c:\mysql\myu.ini"
"https://qualysapi.qualys.com/api/2.0/fo/auth/mysql/"
```

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>2018-07-17T21:14:05Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>272380</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Example: Update MySQL record

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=update&ids=272380&ips=10.10.10.19&username=USERNAME&title=NewMySQ
LRecord&ssl_verify=0&login_type=vault&vault_type=CyberArk PIM
Suite&vault_id=248308&folder=folder&file=file&hosts=www.qualys.com&commen
ts=test comments updated"
"https://qualysapi.qualys.com/api/2.0/fo/auth/mysql/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE BATCH_RETURN SYSTEM
"qualysapi.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
  <RESPONSE>
    <DATETIME>2018-07-17T21:53:55Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>284212</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

DTD Update

We have added new elements in the DTD for displaying vault information.

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

<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
POST_DATA?)>
...
<!ELEMENT AUTH_MYSQL (ID, TITLE, USERNAME, DATABASE, PORT, HOSTS?,
IP_SET?, LOGIN_TYPE?, DIGITAL_VAULT?, SSL_VERIFY, WINDOWS_CONF_FILE,
UNIX_CONF_FILE, CLIENT_CERT?, CLIENT_KEY?, NETWORK_ID?, CREATED,
LAST_MODIFIED, COMMENTS?)>
....
<!ELEMENT LOGIN_TYPE (#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 FIRST_NAME (#PCDATA)>
<!ELEMENT LAST_NAME (#PCDATA)>
<!-- EOF -->
```

List Tomcat Records - DTD Change

API affected	/api/2.0/fo/auth/tomcat/?action=list
New or Updated API	Updated (DTD change only)
DTD or XSD changes	Yes

The Auth Tomcat List Output DTD is used when you list Tomcat authentication records in your account. In this DTD, we changed the element SERVICE_NAME to SERVICE_NAME_WINDOWS.

Updated DTD:

<baseurl>/api/2.0/fo/auth/tomcat/auth_tomcat_list_output.dtd

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

...

<!ELEMENT AUTH_TOMCAT (ID, TITLE, IP_SET, INSTALLATION_PATH?,
INSTANCE_PATH?, AUTO_DISCOVER_INSTANCES?,
INSTALLATION_PATH_WINDOWS?, INSTANCE_PATH_WINDOWS?,
SERVICE_NAME_WINDOWS?, NETWORK_ID?, CREATED, LAST_MODIFIED,
COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT INSTALLATION_PATH (#PCDATA)>
<!ELEMENT INSTANCE_PATH (#PCDATA)>
<!ELEMENT AUTO_DISCOVER_INSTANCES (#PCDATA)>
<!ELEMENT INSTALLATION_PATH_WINDOWS (#PCDATA)>
<!ELEMENT INSTANCE_PATH_WINDOWS (#PCDATA)>
<!ELEMENT SERVICE_NAME_WINDOWS (#PCDATA)>
<!ELEMENT IP_SET (IP|IP_RANGE)+>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT IP_RANGE (#PCDATA)>
<!ELEMENT NETWORK_ID (#PCDATA)>
<!ELEMENT CREATED (DATETIME, BY)>
<!ELEMENT BY (#PCDATA)>
<!ELEMENT LAST_MODIFIED (DATETIME)>
<!ELEMENT COMMENTS (#PCDATA)>
...
```

Scanner Appliance - IPv6 Support for VLANs and Static Routes

API affected	api/2.0/fo/appliance/?action=update
New or Updated API	Updated
DTD or XSD changes	No
API affected	/api/2.0/fo/appliance/physical/?action=update
New or Updated API	Updated
DTD or XSD changes	No
API affected	/api/2.0/fo/appliance/?action=list&output_mode=full
New or Updated API	Updated (output change)
DTD or XSD changes	Yes

We now support IPv6 addresses when defining VLANs and static routes for virtual and physical scanner appliances. Appliances can have a mix of IPv4 configurations and IPv6 configurations.

Set VLANs on Scanner Appliance

Use the “set_vlans” parameter to specify one or more VLANs.

The format for a single VLAN is:

<VLAN_ID> | <IPv4_ADDRESS> | <NETMASK> | <VLAN_NAME> | ipv6_static or ipv6_auto | <IPv6_ADDRESS> with pipe (|) used as a delimiter.

To skip IPv4 attributes (IPv4 address and netmask), you must include an empty space in place of each attribute. Multiple VLANs can be assigned using a comma separated list.

Attribute	Description
<VLAN_ID>	Customer-defined ID (not assigned by Qualys). Must be in the range 0 to 4096, inclusive.
<IPv4_ADDRESS>	A valid IPv4 IP address (dotted quad), such as 10.10.10.1. Leave empty when specifying an IPv6 address.
<NETMASK>	A valid network mask (dotted quad), such as 255.255.255.0. eave empty when specifying an IPv6 address.
<VLAN_NAME>	A valid name (can be empty). The name can be a maximum of 256 ASCII characters. The character : (colon) is permitted. These characters are not permitted: , (comma), < (less than), > (greater than), " (double quote), & (ampersand), | (pipe), = (equals).

Attribute	Description
ipv6_static or ipv6_auto	Specify ipv6_static to provide a static IPv6 address. Specify ipv6_auto to auto-configure IPv6 using SLAAC on the VLAN.
<IPv6_ADDRESS>	A valid IPv6 address is required when "ipv6_static" is specified, such as fdd1:0:1:107::500. Leave empty when "ipv6_auto" is specified.

Examples for Virtual Scanner Appliance

API request (one IPv6 VLAN):

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&id=126209&set_vlans=1234|||Name1234|ipv6_static|fdd
1:0:1:109::500"
"https://qualysapi.qualys.com/api/2.0/fo/appliance/"
```

API request (one IPv4 VLAN):

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&id=126209&set_vlans=5678|123.123.123.123|255.255.25
5.255|Name5678"
"https://qualysapi.qualys.com/api/2.0/fo/appliance/"
```

API request (mix of IPv6 and IPv4 VLANs):

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&id=126209&set_vlans=1234|||Name1234|ipv6_static|fdd
1:0:1:108::500,5678|123.123.123.123|255.255.255.255|Name5678,9012|
244.244.244.244|255.255.255.0|Name9012|ipv6_auto,3456|12.12.12.12|
255.255.255.0|Name3456|ipv6_static|fdd1:0:1:107::500"
"https://qualysapi.qualys.com/api/2.0/fo/appliance/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-07-20T11:45:03Z</DATETIME>
    <TEXT>Virtual scanner updated successfully</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>126209</VALUE>
```

```
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Example for Physical Scanner Appliance

You'll set VLANs on physical scanner appliances in the same way as virtual appliances. The only difference is the URL endpoint.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&id=126209&set_vlans=1234|||Name1234|ipv6_static|fdd
1:0:1:109::500"
"https://qualysapi.qualys.com/api/2.0/fo/appliance/physical"
```

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>2018-07-20T11:58:06Z</DATETIME>
    <TEXT>Physical scanner updated successfully</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>126209</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Set Static Routes on Scanner Appliance

Use the "set_routes" parameter to specify one or more static routes.

The format for a single static route is:

<IPv4_ADDRESS> | <NETMASK> | <IPv4_GATEWAY> | <VLAN_NAME> | <IPv6_ADDRESS> | <IPv6_GATEWAY>, with pipe (|) used as the delimiter.

To skip IPv4 attributes (IPv4 address, netmask and gateway), you must include an empty space in place of each attribute. Multiple static routes can be assigned using a comma separated list.

Attribute	Description
<IPv4_ADDRESS>	A valid IPv4 IP address (dotted quad), such as 10.10.26.0. Leave empty when specifying an IPv6 address.
<NETMASK>	A valid network mask (dotted quad), such as 255.255.255.0. Leave empty when specifying an IPv6 address.
<IPv4_GATEWAY>	A valid IPv4 address (dotted quad), such as 10.10.25.255. Leave empty when specifying an IPv6 address.
<VLAN_NAME>	A valid name (can be empty). The name can be a maximum of 256 ASCII characters. The character : (colon) is permitted. These characters are not permitted: , (comma), < (less than), > (greater than), " (double quote), & (ampersand), | (pipe), = (equals).
<IPv6_ADDRESS>	A valid IPv6 address (with or without the prefix), such as fdd1:0:1:107::500.
<IPv6_GATEWAY>	A valid IPv6 gateway address, such as 2001:470:8418:280d::1.

Examples for Virtual Scanner Appliance

API request (one IPv6 static route):

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&id=126209&set_routes=|||Name1|fdd1:0:1:107::500|200
1:470:8418:280d::1"
"https://qualysapi.qualys.com/api/2.0/fo/appliance/"
```

API request (one IPv4 static route):

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&id=126209&set_routes=192.0.0.0|255.255.255.0|10.100
.11.157|Name2"
"https://qualysapi.qualys.com/api/2.0/fo/appliance/"
```

API request (mix of IPv6 and IPv4 static routes):

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&id=126209&set_routes=192.0.0.0|255.255.255.0|10.100
.11.157|Name2,192.168.0.0|255.255.0.0|10.100.11.157|Name3,192.168.
10.0||10.100.11.157|Name4,192.167.0.0|255.255.0.0|10.100.11.157|Na
me5|fdd1:0:1:107::500|2001:470:8418:280d::1,|||Name1|fdd1:0:1:107:
:500/64|2001:470:8418:280d::1"
```

```
"https://qualysapi.qualys.com/api/2.0/fo/appliance/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-07-20T11:45:03Z</DATETIME>
    <TEXT>Virtual scanner updated successfully</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>126209</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Example for Physical Scanner Appliance

You'll set static routes on physical scanner appliances in the same way as virtual appliances. The only difference is the URL endpoint.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&id=126209&set_routes=|||Name1|fdd1:0:1:107::500|200
1:470:8418:280d::1"
"https://qualysapi.qualys.com/api/2.0/fo/appliance/physical"
```

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>2018-07-20T11:58:06Z</DATETIME>
    <TEXT>Physical scanner updated successfully</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
```

```
          <VALUE>126209</VALUE>
        </ITEM>
      </ITEM_LIST>
    </RESPONSE>
  </SIMPLE_RETURN>
```

List Scanner Appliances

When you list appliances with output_mode=full, you'll see IPv6 VLAN and static route configurations when set on an appliance. <IPV6_SLAAC /> indicates that ipv6_auto was specified for auto-configuring IPv6 using SLAAC on the VLAN.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=list&echo_request=1&output_mode=full&ids=126209"
"https://qualysapi.qualys.com/api/2.0/fo/appliance/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE APPLIANCE_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/appliance/appliance_list_
output.dtd">
<APPLIANCE_LIST_OUTPUT>
  <REQUEST>
    <DATETIME>2018-07-20T12:10:48Z</DATETIME>
...
  <RESPONSE>
    <DATETIME>2018-07-20T12:10:48Z</DATETIME>
    <APPLIANCE_LIST>
      <APPLIANCE>
        <ID>126209</ID>
...
        <VLANS>
          <SETTING>Enabled</SETTING>
          <VLAN>
            <ID>3456</ID>
            <NAME>Name3456</NAME>
            <IP_ADDRESS>12.12.12.12</IP_ADDRESS>
            <NETMASK>255.255.255.0</NETMASK>
            <IPV6_ADDRESS>fdd1:0:1:107::500</IPV6_ADDRESS>
          </VLAN>
          <VLAN>
```

```
            <ID>9012</ID>
            <NAME>Name9012</NAME>
            <IP_ADDRESS>244.244.244.244</IP_ADDRESS>
            <NETMASK>255.255.255.0</NETMASK>
            <IPV6_SLAAC />
          </VLAN>
          <VLAN>
            <ID>5678</ID>
            <NAME>Name5678</NAME>
            <IP_ADDRESS>123.123.123.123</IP_ADDRESS>
            <NETMASK>255.255.255.255</NETMASK>
          </VLAN>
          <VLAN>
            <ID>1234</ID>
            <NAME>Name1234</NAME>
            <IPV6_ADDRESS>fdd1:0:1:108::500</IPV6_ADDRESS>
          </VLAN>
        </VLANS>
        <STATIC_ROUTES>
          <ROUTE>
            <NAME>Name1</NAME>
            <IPV6_GATEWAY>2001:470:8418:280d::1</IPV6_GATEWAY>
            <IPV6_ADDRESS>fdd1:0:1:107::500</IPV6_ADDRESS>
            <IPV6_NETWORK>fdd1:0:1:107::/64</IPV6_NETWORK>
          </ROUTE>
          <ROUTE>
            <NAME>Name5</NAME>
            <IPV6_GATEWAY>2001:470:8418:280d::1</IPV6_GATEWAY>
            <IPV6_ADDRESS>fdd1:0:1:107::500</IPV6_ADDRESS>
          </ROUTE>
        </STATIC_ROUTES>
        ...
      </APPLIANCE>
    </APPLIANCE_LIST>
  </RESPONSE>
</APPLIANCE_LIST_OUTPUT>
```

Updated DTD:

<baseurl>/api/2.0/fo/appliance/appliance_list_output.dtd

Updates were made to the VLANS and STATIC_ROUTES elements to support IPv6.

```
<!-- QUALYS APPLIANCE_LIST_OUTPUT DTD -->

<!ELEMENT APPLIANCE_LIST_OUTPUT (REQUEST?,RESPONSE)>
...
    <!ELEMENT RESPONSE (DATETIME, APPLIANCE_LIST?, LICENSE_INFO?)>
        <!ELEMENT APPLIANCE_LIST (APPLIANCE+)>
            <!ELEMENT APPLIANCE (ID, UUID, NAME, NETWORK_ID?,
SOFTWARE_VERSION, RUNNING_SLICES_COUNT, RUNNING_SCAN_COUNT,
STATUS, CMD_ONLY_START?, MODEL_NUMBER?, TYPE?, SERIAL_NUMBER?,
ACTIVATION_CODE?, INTERFACE_SETTINGS*, PROXY_SETTINGS?,
IS_CLOUD_DEPLOYED?, CLOUD_INFO?, VLANS?, STATIC_ROUTES?,
ML_LATEST?, ML_VERSION?, VULNSIGS_LATEST?, VULNSIGS_VERSION?,
ASSET_GROUP_COUNT?, ASSET_GROUP_LIST?, ASSET_TAGS_LIST?,
LAST_UPDATED_DATE?, POLLING_INTERVAL?, USER_LOGIN?,
HEARTBEATS_MISSED?, SS_CONNECTION?, SS_LAST_CONNECTED?,
FDCC_ENABLED?, USER_LIST?, UPDATED?, COMMENTS?, RUNNING_SCANS?,
MAX_CAPACITY_UNITS?)>
...
                <!ELEMENT VLANS (SETTING, VLAN*)>
                  <!ELEMENT VLAN (ID, NAME, IP_ADDRESS?, NETMASK?,
IPV6_ADDRESS?, IPV6_SLAAC?)>
                          <!ELEMENT IPV6_SLAAC EMPTY>
                <!ELEMENT STATIC_ROUTES (ROUTE*)>
                    <!ELEMENT ROUTE (NAME, IP_ADDRESS?, NETMASK?,
GATEWAY?, IPV6_ADDRESS?, IPV6_NETWORK?, IPV6_GATEWAY?)>
                          <!ELEMENT IPV6_NETWORK (#PCDATA)>
                          <!ELEMENT IPV6_GATEWAY (#PCDATA)>
...
```

Option Profile API - Export System Profiles

API affected	/api/2.0/fo/subscription/option_profile/
New or Updated API	Updated
DTD or XSD changes	No

When you export option profiles from your subscription you now have the option to include the System option profiles in the output. System option profiles are Initial Options, Initial PC Options, Payment Card Industry (PCI) Options, 2008 SANS20 Options and Qualys Top 20 Options. These are not exported by default.

Use these parameters to export System option profiles.

Parameter	Description
action=export	(Required) The GET or POST method may be used.
include_system_option_profiles={**0**|1}	(Optional) When unspecified or set to 0, system option profiles are not included in the output. Specify 1 to include system option profiles in the output.

Example (System profiles included)

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl"
"https://qualysapi.qualys.com/api/2.0/fo/subscription/option_profi
le/?action=export&include_system_option_profiles=1"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE OPTION_PROFILES SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/subscription/option_profi
le/option_profile_info.dtd">
<OPTION_PROFILES>
  <OPTION_PROFILE>
    <BASIC_INFO>
      <ID>5486</ID>
      <GROUP_NAME><![CDATA[Initial Options]]></GROUP_NAME>
      <GROUP_TYPE>user</GROUP_TYPE>
      <USER_ID><![CDATA[Patrick Slimmer (qualys_ps)]]></USER_ID>
      <UNIT_ID>0</UNIT_ID>
      <SUBSCRIPTION_ID>9054</SUBSCRIPTION_ID>
      <IS_DEFAULT>1</IS_DEFAULT>
```

```
      <IS_GLOBAL>1</IS_GLOBAL>
      <IS_OFFLINE_SYNCABLE>0</IS_OFFLINE_SYNCABLE>
      <UPDATE_DATE>2016-08-30T05:49:59Z</UPDATE_DATE>
    </BASIC_INFO>
...
  <OPTION_PROFILE>
    <BASIC_INFO>
      <ID>5487</ID>
      <GROUP_NAME><![CDATA[Qualys Top 20 Options]]></GROUP_NAME>
      <GROUP_TYPE>rv10</GROUP_TYPE>
      <USER_ID><![CDATA[Patrick Slimmer (qualys_ps)]]></USER_ID>
      <UNIT_ID>0</UNIT_ID>
      <SUBSCRIPTION_ID>9054</SUBSCRIPTION_ID>
      <IS_DEFAULT>0</IS_DEFAULT>
      <IS_GLOBAL>1</IS_GLOBAL>
      <IS_OFFLINE_SYNCABLE>0</IS_OFFLINE_SYNCABLE>
      <UPDATE_DATE>2016-08-30T05:50:00Z</UPDATE_DATE>
    </BASIC_INFO>
...
  <OPTION_PROFILE>
    <BASIC_INFO>
      <ID>5488</ID>
      <GROUP_NAME><![CDATA[2008 SANS20 Options]]></GROUP_NAME>
      <GROUP_TYPE>sans20</GROUP_TYPE>
      <USER_ID><![CDATA[Patrick Slimmer (qualys_ps)]]></USER_ID>
      <UNIT_ID>0</UNIT_ID>
      <SUBSCRIPTION_ID>9054</SUBSCRIPTION_ID>
      <IS_DEFAULT>0</IS_DEFAULT>
      <IS_GLOBAL>1</IS_GLOBAL>
      <IS_OFFLINE_SYNCABLE>0</IS_OFFLINE_SYNCABLE>
      <UPDATE_DATE>2016-08-30T05:50:00Z</UPDATE_DATE>
    </BASIC_INFO>
...
  <OPTION_PROFILE>
    <BASIC_INFO>
      <ID>5489</ID>
      <GROUP_NAME><![CDATA[Payment Card Industry (PCI)
Options]]></GROUP_NAME>
      <GROUP_TYPE>pci</GROUP_TYPE>
      <USER_ID><![CDATA[Patrick Slimmer (qualys_ps)]]></USER_ID>
      <UNIT_ID>0</UNIT_ID>
```

```
      <SUBSCRIPTION_ID>9054</SUBSCRIPTION_ID>
      <IS_GLOBAL>1</IS_GLOBAL>
      <IS_OFFLINE_SYNCABLE>0</IS_OFFLINE_SYNCABLE>
      <UPDATE_DATE>2016-08-30T05:50:00Z</UPDATE_DATE>
    </BASIC_INFO>
...
  <OPTION_PROFILE>
    <BASIC_INFO>
      <ID>5490</ID>
      <GROUP_NAME><![CDATA[Initial PC Options]]></GROUP_NAME>
      <GROUP_TYPE>compliance</GROUP_TYPE>
      <USER_ID><![CDATA[Patrick Slimmer (qualys_ps)]]></USER_ID>
      <UNIT_ID>0</UNIT_ID>
      <SUBSCRIPTION_ID>9054</SUBSCRIPTION_ID>
      <IS_GLOBAL>1</IS_GLOBAL>
      <UPDATE_DATE>2016-08-30T05:50:00Z</UPDATE_DATE>
    </BASIC_INFO>
...
</OPTION_PROFILES>
```

More Option Profile functions for VM, PCI, PC

APIs affected	/api/2.0/fo/subscription/option_profile/vm/ /api/2.0/fo/subscription/option_profile/pc/ /api/2.0/fo/subscription/option_profile/pci/
New or Updated API	New
DTD or XSD changes	Updated

You can now create, update, list and delete option profiles for VM, PCI, and PC.

Samples for creating VM, PCI, and PC Option Profiles are provided here. For information on other operations and APIs for VM, PCI, and PC, refer to the *Qualys API for VM and Compliance User Guide*.

VM | PCI | PC

Create VM Option Profile

You can create a VM Option Profile by using the API (/api/2.0/fo/subscription/option_profile/vm/) with action=create. The POST access method should be used to make an API request.

Input Parameters:

Parameter	Description
action=create	(Required)
title={value}	(Required) A title for easy identification.
owner={value}	(Optional) The owner of the option profile(s), or the user who created the option profile.
default={**0**|1}	(Optional) Make this profile the default for all scans and maps. Specify 1 to make default. There can only be one default profile for the subscription.
global={**0**|1}	(Optional) Share this profile with other users by making it global. Are you a Manager? This profile will be available to all users. Are you a Unit Manager? This profile will be available to all users in your business unit. Specify 1 to make global.
offline_scanner={**0**|1}	(Optional) Specify to 1 to download this profile to your offline scanners during the next sync.

Parameter	Description
scan_tcp_ports={none|full|**standard**|light}	(Required) We use ports to send packets to the host in order to determine whether the host is alive and also to do fingerprinting for the discovery of services. Specify "full" to scan all ports, "standard" to scan standard ports or "light" to scan fewer ports. See *Qualys API for VM and Compliance User Guide* for a list of ports used for standard or light scan. We will scan the standard list of ports unless you choose a different option in the profile.
scan_tcp_ports_additional={port1,port2}	(Optional) Specify additional ports to scan (up to 12500 ports).
3_way_handshake={**0**|1}	(Optional) Specify 1 to let the scanning engine perform a 3-way handshake with target hosts. After a connection between the service and the target host is established, the connection will be closed. This option should be enabled only if you have a configuration that does not allow an SYN packet to be followed by an RST packet. Also, when this is enabled, TCP based OS detection is not performed on target hosts. Without TCP based OS detection, the service may not be able to identify the operating system installed on target hosts and perform OS-specific vulnerability checks
Scan	
scan_udp_ports={none|full|**standard**|light}	(Required) Specify "full" to scan all ports, "standard" to scan standard ports or "light" to scan fewer ports. See *Qualys API for VM and Compliance User Guide* for a list of UDP ports used for standard or light scan. We will scan the standard list of ports unless you choose a different option in the profile.
scan_udp_ports_additional ={port1,port2}	(Optional) Specify additional ports to scan (up to 20500 ports).
authoritative_option={**0**|1}	(Optional) Specify 1 to enable Authoritative Scan Option. By enabling the authoritative scan option your light scan will work like a full or standard scan. We will update the vulnerability status for all vulnerabilities found, regardless of which ports they were detected on.
scan_dead_hosts={**0**|1}	(Optional) Specify 1 to enable scanning dead hosts. A dead host is a host that is unreachable - it didn't respond to any pings. Your scan may run longer if you choose to scan dead hosts.
close_vuln_on_dead_hosts={**0**|1}	(Optional) Specify 1 to quickly close vulnerabilities for hosts that are not found alive after a set number of scans. When enabled, we'll mark existing tickets associated with dead hosts as Closed/Fixed and update the vulnerability status to Fixed.
not_found_alive_times={value}	(Optional) Specify the number of times the host is not found alive after which the vulnerability should be closed. This setting is available only when close_vuln_on_dead_hosts=1.

Parameter	Description
purge_host_data={**0**|1}	(Optional) Specify 1 to purge host data. This option is especially useful if you have systems that are regularly decommissioned or replaced. By specifying this option you're telling us you want to purge the host if we detect a change in the host's Operating System (OS) vendor at scan time, for example the OS changed from Linux to Windows or Debian to Ubuntu. We will not purge the host for an OS version change like Linux 2.8.13 to Linux 2.9.4.
external_scanners_use={value}	(Optional) Specify the maximum number of external scanners to use for scanning perimeter assets. (This option is available when your subscription is configured with multiple external scanners).
scan_parallel_scaling={**0**|1}	(Optional) Specify 1 to enable parallel scaling. This setting can be useful in subscriptions which have physical and virtual scanner appliances with different performance characteristics (e.g., CPU, RAM). Specify this option to dynamically scale up the number of hosts to scan in parallel (at scan time) to a calculated value which is based upon the computing resources available on each appliance. Note that the number of hosts to scan in parallel value determines how many hosts each appliance will target concurrently, not how many appliances will be used for the scan.
scan_overall_performance={high|**normal**|low|custom}	(Optional) The profile "normal" is recommended in most cases. The settings for scan_external_scanners, scan_scanner_appliances, scan_total_process, scan_http_process, scan_packet_delay, and scan_intensity change as per the specified profile. Normal - Well balanced between intensity and speed. High - Recommended only when scanning a single IP or a small number of IPs. Optimized for speed and shorter scan times. Low - Recommended if responsiveness for individual hosts and services is low. Optimized for low bandwidth network connections and highly utilized networks. May take longer to complete.
scan_external_scanners={value}	(Optional) Specify the number of external scanners to be used for associated scans. This setting is available only if you have multiple external scanners in your subscription. For example, if you have 10 external scanners in your subscription, you can configure this setting to any number between 1 to 10.
scan_scanner_appliances={value}	(Optional) Specify the number of scanner appliances to scan at the same time (per scan task). Launching several concurrent scans on the same scanner appliance has a multiplying effect on bandwidth usage and may exceed available scanner resources. Don't have scanner appliances? Disregard the Scanner Appliance setting.

Parameter	Description
scan_total_process={value}	(Optional) Specify the maximum number of processes to run at the same time per host. Note that the total number of processes includes the HTTP processes.
scan_http_process={value}	(Optional) Specify the maximum number of HTTP processes to run at the same time.
scan_packet_delay={minimum|short|**medium**|long|maximum}	(Optional) Specify the delay between groups of packets sent to each host during a scan. With a short delay, packets are sent more frequently. With a long delay, packets are sent less frequently.
scan_intensity={**normal**|medium|low|minimum}	(Optional) This setting determines the aggressiveness (parallelism) of port scanning and host discovery at the port level. Lowering the intensity level has the effect of serializing port scanning and host discovery. This is useful for certain network conditions like cascading firewalls and lower scan prioritization on the network. Tip - If you are scanning through a firewall we recommended you reduce the intensity level. Unauthenticated scans see more of a performance difference using this option.
load_balancer={**0**|1}	(Optional) Specify 1 to check each target host to determine if it's a load balancer. When a load balancer is detected, we determine the number of Web servers behind it and report QID 86189 "Presence of a Load-Balancing Device Detected" in your results.
password_brute_forcing_system={**minimal**|limited|standard|exhaustive}	(Optional) How vulnerable are your hosts to password-cracking techniques? we'll attempt to guess the password for each detected login ID on each target host scanned. Specify the level of brute forcing you prefer ("minimal" to "exhaustive").
password_brute_forcing_custom={value1,value2}	(Optional) Specify titles of the login/password pairs you create for password brute forcing on the Qualys Cloud Platform UI.
vulnerability_detection={**complete**|custom|runtime}	(Optional) With a "complete" scan we'll scan for all vulnerabilities (QIDs) in the KnowledgeBase applicable to each host being scanned. Specify "custom" to limit the scan to specified QIDs only. Then add the QIDs you want to scan. Specify “runtime” to scan QIDs at runtime.
custom_search_list_ids={value1, value2}	(Optional) Specify ids of search lists you want to use in your scan.
custom_search_list_title={value1, value2}	(Optional) Specify titles of search lists you want to use in your scan.
basic_host_information_checks={0|**1**}	(Optional) Adds basic host information checks (hostname, OS, etc) to your Custom scans. These are already included in Complete scans. This setting is enabled by default.

Parameter	Description
oval_checks={**0**|1}	(Optional) Specify 1 to add a search list with QID 105186 (a diagnostic check for OVAL).
all_qrdi_checks={**0**|1}	(Optional) Specify 1 to scan target assets for all QRDI vulnerabilities in your subscription, i.e. all custom vulnerability checks defined with QRDI (Qualys Remote Detection Interface).
exclude_search_list_ids={value1, value2}	(Optional) Specify ids of search lists you want to exclude from your scan.
authentication={value1, value2}	(Optional) Want to run authenticated scans? When you use authentication we'll perform a more in-depth assessment and get you the most accurate results with fewer false positives. Specify one or more technologies for the hosts you want to scan. Be sure you've configured authentication records (under Scans > Authentication) before running your scan. The following options are available: - Windows - Unix - Oracle - Oracle Listener - SNMP - VMware - DB2 - HTTP - MySQL - MongoDB - Tomcat Server - Palo Alto Networks Firewall
enable_additional_certificate_detection={**0**|1}	(Optional) Want to detect additional certificates beyond ports? You need to enable authentication and then run new vulnerability scans. Specify 1 to enable this option before scanning and see additional certificate records (under Assets > Certificates).
enable_dissolvable_agent={**0**|1}	(Optional) Specify 1 to enable dissolvable agent. This is required for certain scan features like Windows Share Enumeration. How does it work? At scan time the Agent is installed on Windows devices to collect data, and once the scan is complete it removes itself completely from target systems.
enable_windows_share_enumeration={**0**|1}	(Optional) Specify 1 to use Windows Share Enumeration to find and report details about Windows shares that are readable by everyone. This test is performed using QID 90635. Make sure 1) the Dissolvable Agent is enabled, 2) QID 90635 is included in the Vulnerability Detection section, and 3) a Windows authentication record is defined.
enable_lite_os_scan={**0**|1}	(Optional) Only interested in OS detection? Specify 1 to include QID 45017 in the scan (under Vulnerability Detection).

Parameter	Description
custom_http_header={value}	(Optional) Specify a custom value in order to drop defenses (such as logging, IPs, etc) when authorized scans are being run.
custom_http_definition_key={value}	(Optional) Specify a custom HTTP header definition key
custom_http_definition_header={value}	(Optional) Specify a value for the custom HTTP header definition key defined in custom_http_definition_key.
host_alive_testing={**0**|1}	(Optional) Specify 1 to run a quick scan to determine which of your target hosts are alive without also performing other scan tests. The Appendix section of your Scan Results report will list the hosts that are alive and hosts that are not alive. You may see some Information Gathered QIDs in the results for hosts found alive.
not_overwrite_os={**0**|1}	(Optional) Specify 1 if you're running a light or custom scan and you don't want to overwrite the OS detected by a previous scan.
Map	
basic_information_gathering=[**all**|register|netblockonly|none]	(Required) Perform basic information gathering on: All: All Hosts (hosts detected by the map), Register: Registered Hosts (hosts in your account), Netblockonly: Netblock Hosts (hosts added by a user to the netblock for the target domain) or None.
map_tcp_ports_standard_scan={0|**1**}	(Optional) Specify 1 to enable standard scan of TCP ports. Standard Scan includes 13 ports: 21-23, 25, 53, 80, 88, 110-111, 135, 139, 443, 445.
map_tcp_ports_additional={value1,value2}	(Optional) Specify additional TCP ports to scan. You can specify up to 20 ports including the standard scan ports.
map_udp_ports_standard_scan={**0**|1}	(Optional) Specify 1 to enable standard scan of UDP ports. Standard Scan includes 6 ports: 53, 111, 135, 137, 161, 500.
map_udp_ports_additional={value1,value2}	(Optional) Specify additional UDP ports to scan. You can specify up to 10 ports including the standard scan ports.
perform_live_host_sweep={0|**1**}	(Optional) Default setting is 1. Specify 0 to only discover devices using DNS discovery methods (DNS, Reverse DNS and DNS Zone Transfer.) Active probes will not be sent. As a result, we may not be able to detect all hosts in the netblock, and undetected hosts will not be analyzed.

Parameter	Description
disable_dns_traffic={**0**|1}	(Optional) Specify 1 if you want to disable DNS traffic for maps. This is valid only when the target domain name includes one or more netblocks, e.g. none:[10.10.10.2-10.10.10.100]. We'll perform network discovery only for the IP addresses in the netblocks. No forward or reverse DNS lookups, DNS zone transfers or DNS guessing/bruteforcing will be made, and DNS information will not be included in map results.
map_overall_performance={high|**normal**|low|custom}	(Optional) The profile "normal" is recommended in most cases. The settings for map_external_scanners, map_scanner_appliances, map_netblock_size, and map_packet_delay change as per the specified profile. Normal - Well balanced between intensity and speed. High - Optimized for speed. May be faster to complete but may overload firewalls and other networking devices. Low - Optimized for low bandwidth network connections. May take longer to complete.
map_external_scanners={value}	(Optional) Specify the number of external scanners for netblocks to map at the same time per scanner. This setting is available only if you have multiple external scanners in your subscription. For example, if you have 10 external scanners in your subscription, you can configure this setting to any number between 1 to 10.
map_scanner_appliances={value}	(Optional) Specify the number of scanner appliances for netblocks to map at the same time per scanner. Launching several concurrent scans on the same scanner appliance has a multiplying effect on bandwidth usage and may exceed available scanner resources. Don't have scanner appliances? Disregard the Scanner Appliance setting.
map_netblock_size={1024 IPs|4096 IPs|8192 IPs|**16384 IPs**|32768 IPs|65536 IPs}	(Optional) Specify the max number of IPs per netblock being mapped. The netblock specified for the domain is broken into smaller netblocks for processing. Each of these smaller netblocks equals a single map process. Use this setting to define how many IPs should be included in each process.
map_packet_delay={**minimum**|short|medium|long|maximum}	(Optional) This is the delay between groups of packets sent to the netblocks being mapped. With a short delay, packets are sent more frequently, resulting in more bandwidth utilization and a shorter mapping time. With a long delay, packets are sent less frequently, resulting in less bandwidth utilization and a longer mapping time.
map_authentication={VMware}	(Optional) Authentication enables the scanner to log into hosts at scan time to extend detection capabilities. See the online help to learn how to configure this option.
Additional	

Parameter	Description
additional_tcp_ports={0|**1**}	(Optional) Specify 1 to enable host discovery on additional TCP ports. Default setting is 1.
additional_tcp_ports_ standard_scan={0|**1**}	(Optional) Specify 1 to enable standard scan of additional TCP ports. Standard Scan includes 13 ports: 21-23, 25, 53, 80, 88, 110-111, 135, 139, 443, 445. Default setting is 1.
additional_tcp_ports_ additional={value1,value2}	(Optional) Specify additional TCP ports to scan. You can specify up to 20 ports including the standard scan ports.
additional_udp_ports={0|**1**}	(Optional) Specify 1 to enable host discovery on additional UDP ports. Default setting is 1.
additional_udp_ports_type= {**standard**|custom}	(Optional) Specify "standard" to enable standard scan of additional UDP ports. Standard Scan includes 6 ports: 53, 111, 135, 137, 161, 500. Default is "standard". Specify "custom" to provide a custom list of ports using additional_udp_ports_custom.
additional_udp_ports_ custom={value1,value2}	(Optional) Specify additional UDP ports to scan. You can specify up to 10 ports including the standard scan ports.
icmp={0|**1**}	(Optional) Specify 1 to only discover live hosts that respond to an ICMP ping. Default setting is 1.
blocked_resources={**0**|1}	(Optional) Specify 1 in order to add ports protected by your firewall/IDS to prevent them from being scanned.
protected_ports={**default**| custom}	(Optional) Ports protected by your firewall/IDS. Specify "default" to provide a list of default blocked ports: 0-1, 111, 513-514, 2049, 4100, 6000-6005, 7100, 8000. Default setting is "default". Specify "custom" to provide a custom list of protected ports using protected_ports_custom.
protected_ports_custom= {value1,value2}	(Optional) Specify a custom list of protected ports.
protected_ips={**all**|custom}	(Optional) IP addresses and ranges protected by your firewall/IDS. Default is "all".
protected_ips_custom= {value1,value2}	(Optional) Specify a custom list of IP addresses and ranges protected by your firewall/IDS.
ignore_firewall_generated_ tcp_rst_packets={**0**|1}	(Optional) Specify 1 to identify firewall-generated TCP RESET packets and ignore them.
ignore_all_tcp_rst_packets= {**0**|1}	(Optional) Specify 1 to ignore all TCP RESET packets - firewall-generated and live-host-generated.

Parameter	Description
ignore_firewall_generated_tcp_syn_ack_packets={**0**|1}	(Optional) Specify 1 to determine if TCP SYN-ACK packets are generated by a filtering device and ignore packets that appear to originate from such devices.
not_send_tcp_ack_or_syn_ack_packets_during_host_discovery={**0**|1}	(Optional) Specify 1 if you do not want to send TCP ACK or SYN-ACK packets. Out of state TCP packets are not SYN packets and do not belong to an existing TCP session.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X POST
"action=create&title=99&global=1&scan_tcp_ports=full&scan_udp_ports=stand
ard&&scan_overall_performance=normal&vulnerability_detection=complete&bas
ic_information_gathering=all"
"http://qualysapi.qualys.com/api/2.0/fo/subscription/option_profile/vm/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"http://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-04-26T06:40:03Z</DATETIME>
    <TEXT>Option profile successfully added.</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>32112</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Create PCI Option Profile

You can create a PCI Option Profile by using the API (/api/2.0/fo/subscription/option_profile/pci/) with action=create. The POST access method should be used to make an API request.

Input Parameters:

Parameter	Description
action=create	(Required)
title={value}	(Required) A title for easy identification.

Parameter	Description
owner={value}	(Optional) The owner of the option profile(s), or the user who created the option profile.
global={**0**|1}	(Optional) Share this profile with other users by making it global. Are you a Manager? This profile will be available to all users. Are you a Unit Manager? This profile will be available to all users in your business unit. Specify 1 to make global.
offline_scanner={**0**|1}	(Optional) Specify to 1 to download this profile to your offline scanners during the next sync.
scan_parallel_scaling={**0**|1}	(Optional) Specify 1 to enable parallel scaling. This setting can be useful in subscriptions which have physical and virtual scanner appliances with different performance characteristics (e.g., CPU, RAM). Specify this option to dynamically scale up the number of hosts to scan in parallel (at scan time) to a calculated value which is based upon the computing resources available on each appliance. Note that the number of hosts to scan in parallel value determines how many hosts each appliance will target concurrently, not how many appliances will be used for the scan.
Scan	
scan_overall_performance={high|**normal**|low|custom}	(Optional) The profile “normal” is recommended in most cases. The settings for scan_external_scanners, scan_scanner_appliances, scan_total_process, scan_http_process, scan_packet_delay, and scan_intensity change as per the specified profile. Normal - Well balanced between intensity and speed. High - Recommended only when scanning a single IP or a small number of IPs. Optimized for speed and shorter scan times. Low - Recommended if responsiveness for individual hosts and services is low. Optimized for low bandwidth network connections and highly utilized networks. May take longer to complete.
scan_external_scanners={value}	(Optional) Specify the number of external scanners to be used for associated scans. This setting is available only if you have multiple external scanners in your subscription. For example, if you have 10 external scanners in your subscription, you can configure this setting to any number between 1 to 10.
scan_scanner_appliances={value}	(Optional) Specify the number of scanner appliances to scan at the same time (per scan task). Launching several concurrent scans on the same scanner appliance has a multiplying effect on bandwidth usage and may exceed available scanner resources. Don't have scanner appliances? Disregard the Scanner Appliance setting.

Parameter	Description
scan_total_process={value}	(Optional) Specify the maximum number of processes to run at the same time per host. Note that the total number of processes includes the HTTP processes.
scan_http_process={value}	(Optional) Specify the maximum number of HTTP processes to run at the same time.
scan_packet_delay={minimum|short|**medium**|long|maximum}	(Optional) Specify the delay between groups of packets sent to each host during a scan. With a short delay, packets are sent more frequently. With a long delay, packets are sent less frequently.
scan_intensity={**normal**|medium|low|minimum}	(Optional) This setting determines the aggressiveness (parallelism) of port scanning and host discovery at the port level. Lowering the intensity level has the effect of serializing port scanning and host discovery. This is useful for certain network conditions like cascading firewalls and lower scan prioritization on the network. Tip - If you are scanning through a firewall we recommended you reduce the intensity level. Unauthenticated scans see more of a performance difference using this option.
scan_dead_hosts={**0**|1}	(Optional) Specify 1 to enable scanning dead hosts. A dead host is a host that is unreachable - it didn't respond to any pings. Your scan may run longer if you choose to scan dead hosts.
close_vuln_on_dead_hosts={**0**|1}	(Optional) Specify 1 to quickly close vulnerabilities for hosts that are not found alive after a set number of scans. When enabled, we'll mark existing tickets associated with dead hosts as Closed/Fixed and update the vulnerability status to Fixed.
not_found_alive_times={value}	(Optional) Specify the number of times the host is not found alive after which the vulnerability should be closed. This setting is available only when close_vuln_on_dead_hosts=1.
purge_host_data={**0**|1}	(Optional) Specify 1 to purge host data. This option is especially useful if you have systems that are regularly decommissioned or replaced. By specifying this option you're telling us you want to purge the host if we detect a change in the host's Operating System (OS) vendor at scan time, for example the OS changed from Linux to Windows or Debian to Ubuntu. We will not purge the host for an OS version change like Linux 2.8.13 to Linux 2.9.4.
Additional	
additional_tcp_ports_additional={value1,value2}	(Optional) Specify additional TCP ports to scan. You can specify up to 7 additional ports apart from the 13 standard scan ports used by default: 21-23, 25, 53, 80, 88, 110-111, 135, 139, 443, 445.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X POST
"action=create&title=jp pci
333&global=1&offline_scanner=1&external_scanners_use=3&scan_parallel_scal
ing=1&scan_overall_performance=high&additional_tcp_ports_additional=80,35
"
"http://qualysapi.qualys.com/api/2.0/fo/subscription/option_profile/pci/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"http://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-04-26T13:04:21Z</DATETIME>
    <TEXT>Option profile successfully added.</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>32113</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Create PC Option Profile

You can create a PC Option Profile by using the API (/api/2.0/fo/subscription/option_profile/pc/) with action=create. The POST access method should be used to make an API request.

Input Parameters:

Parameter	Description
action=create	(Required)
title={value}	(Required) The title for the option profile.
owner={value}	(Optional) The owner of the option profile(s), or the user who created the option profile.
global={**0**|1}	(Optional) Share this profile with other users by making it global. Are you a Manager? This profile will be available to all users. Are you a Unit Manager? This profile will be available to all users in your business unit. Specify 1 to make global.

Parameter	Description
scan_parallel_scaling={**0**|1}	(Optional) Specify 1 to enable parallel scaling. This setting can be useful in subscriptions which have physical and virtual scanner appliances with different performance characteristics (e.g., CPU, RAM). Specify this option to dynamically scale up the number of hosts to scan in parallel (at scan time) to a calculated value which is based upon the computing resources available on each appliance. Note that the number of hosts to scan in parallel value determines how many hosts each appliance will target concurrently, not how many appliances will be used for the scan.
Scan	
scan_overall_performance={high|**normal**|low|custom}	(Required) The profile "normal" is recommended in most cases. The settings for scan_external_scanners, scan_scanner_appliances, scan_total_process, scan_http_process, scan_packet_delay, and scan_intensity change as per the specified profile. Normal - Well balanced between intensity and speed. High - Recommended only when scanning a single IP or a small number of IPs. Optimized for speed and shorter scan times. Low - Recommended if responsiveness for individual hosts and services is low. Optimized for low bandwidth network connections and highly utilized networks. May take longer to complete.
scan_external_scanners={value}	(Optional) Specify the number of external scanners to be used for associated scans. This setting is available only if you have multiple external scanners in your subscription. For example, if you have 10 external scanners in your subscription, you can configure this setting to any number between 1 to 10.
scan_scanner_appliances={value}	(Optional) Specify the number of scanner appliances to scan at the same time (per scan task). Launching several concurrent scans on the same scanner appliance has a multiplying effect on bandwidth usage and may exceed available scanner resources. Don't have scanner appliances? Disregard the Scanner Appliance setting.
scan_total_process={value}	(Optional) Specify the maximum number of processes to run at the same time per host. Note that the total number of processes includes the HTTP processes.
scan_http_process={value}	(Optional) Specify the maximum number of HTTP processes to run at the same time.
scan_packet_delay={minimum|short|**medium**|long|maximum}	(Optional) Specify the delay between groups of packets sent to each host during a scan. With a short delay, packets are sent more frequently. With a long delay, packets are sent less frequently.

Parameter	Description
scan_intensity={**normal**|medium|low|minimum}	(Optional) This setting determines the aggressiveness (parallelism) of port scanning and host discovery at the port level. Lowering the intensity level has the effect of serializing port scanning and host discovery. This is useful for certain network conditions like cascading firewalls and lower scan prioritization on the network. Tip - If you are scanning through a firewall we recommended you reduce the intensity level. Unauthenticated scans see more of a performance difference using this option.
scan_by_policy={**0**|1}	(Optional) Specify 1 to enable scan by policy. The Scan by Policy option allows you to restrict your scans to the controls in specified policies. You can choose up to 20 policies, one policy at a time. Once you've specified a policy, all controls in that policy will be scanned including any special control types in the policy. This is regardless of the Control Types settings in the profile.
policy_names={value1, value2}	(Optional) Specify policy names to scan by policy.
policy_ids={value1,value2}	(Optional) Specify policy IDs to scan by policy.
auto_update_expected_value={**0**|1}	(Optional) Specify 1 to update the control expected value used for posture evaluation with the actual value returned by the scan.
fim_controls_enabled={**0**|1}	(Optional) Specify 1 to perform file integrity monitoring based on user defined file integrity checks. A file integrity check is a user defined control that checks for changes to a specific file. You should set auto_update_expected_value=1 in order to use this parameter.
custom_wmi_query_checks ={**0**|1}	(Optional) Specify 1 to run Windows WMI query checks. When enabled, WMI query checks will be performed for user defined WMI Query Check controls.
enable_dissolvable_agent= {**0**|1}	(Optional) Specify 1 to enable dissolvable agent. This is required for certain scan features like Windows Share Enumeration. How does it work? At scan time the Agent is installed on Windows devices to collect data, and once the scan is complete it removes itself completely from target systems.
enable_password_auditing= {**0**|1}	(Optional) Specify 1 to check for service provided password auditing controls (control IDs 3893, 3894 and 3895). These controls are used to identify 1) user accounts with empty passwords, 2) user accounts with the password equal to the user name, and 3) user accounts with passwords equal to an entry in a user-defined password dictionary. This setting is available only if enable_dissolvable_agent=1.

Parameter	Description
custom_password_dictionary={value1,value2}	(Optional) Specify passwords in order to create a password dictionary. This is used when evaluating control ID 3895, which identifies user accounts where the password is equal to an entry in the password dictionary.
enable_windows_share_enumeration={**0**|1}	(Optional) Specify 1 to use Windows Share Enumeration to find and report details about Windows shares that are readable by everyone. This test is performed using QID 90635. Make sure 1) the Dissolvable Agent is enabled, 2) QID 90635 is included in the Vulnerability Detection section, and 3) a Windows authentication record is defined.
enable_windows_directory_search={**0**|1}	(Optional) Specify 1 if you've set up Windows Directory Search controls and want to include them in the scan. This custom control allows you to search for files/directories based on various criteria like file name and user access permissions.
scan_ports={standard|**targeted**}	(Required) Specify "standard" to enable standard scan of TCP ports. See *Qualys API for VM and Compliance User Guide* for a list of ports used for standard scan. Specify "targeted" to perform a targeted scan. Which ports are included in a targeted scan? For Unix hosts, these well known ports are scanned: 22 (SSH), 23 (telnet) and 513 (rlogin). Any one of these services is sufficient for authentication. If services (SSH, telnet, rlogin) are not running on these well known ports for the hosts you will be scanning, specify this option and define a custom ports list in the Unix authentication record. Note: The actual ports scanned also depends on the Ports setting in the Unix authentication record. For Windows hosts, the service scans a fixed set of required Windows ports (a service defined, internal list).
Additional	
additional_tcp_ports={0|**1**}	(Optional) Specify 1 to enable host discovery on additional TCP ports. Default setting is 1.
additional_tcp_ports_standard_scan={0|**1**}	(Optional) Specify 1 to enable standard scan of additional TCP ports. Standard Scan includes 13 ports: 21-23, 25, 53, 80, 88, 110-111, 135, 139, 443, 445. Default setting is 1.
additional_tcp_ports_additional={value1,value2}	(Optional) Specify additional TCP ports to scan. You can specify up to 20 ports including the standard scan ports.
additional_udp_ports={0|**1**}	(Optional) Specify 1 to enable host discovery on additional UDP ports. Default setting is 1.

Parameter	Description
additional_udp_ports_type={**standard**|custom}	(Optional) Specify "standard" to enable standard scan of additional UDP ports. Standard Scan includes 6 ports: 53, 111, 135, 137, 161, 500. Default is "standard". Specify "custom" to provide a custom list of ports using additional_udp_ports_custom.
additional_udp_ports_custom={value1,value2}	(Optional) Specify additional UDP ports to scan. You can specify up to 10 ports including the standard scan ports.
icmp={0|**1**}	(Optional) Specify 1 to only discover live hosts that respond to an ICMP ping. Default setting is 1.
blocked_resources={**0**|1}	(Optional) Specify 1 in order to add ports protected by your firewall/IDS to prevent them from being scanned.
protected_ports={**default**|custom}	(Optional) Ports protected by your firewall/IDS. Specify "default" to provide a list of default blocked ports: 0-1, 111, 513-514, 2049, 4100, 6000-6005, 7100, 8000. Default setting is "default". Specify custom to provide a custom list of protected ports using protected_ports_custom.
protected_ports_custom={value1,value2}	(Optional) Specify a custom list of protected ports.
protected_ips={**all**|custom}	(Optional) IP addresses and ranges protected by your firewall/IDS. Default is "all".
protected_ips_custom={value1,value2}	(Optional) Specify a custom list of IP addresses and ranges protected by your firewall/IDS.
ignore_rst_packets={**0**|1}	(Optional) Specify 1 to ignore all TCP RESET packets - firewall-generated and live-host-generated.
ignore_firewall_generated_syn_ack_packets={**0**|1}	(Optional) Specify 1 to determine if TCP SYN-ACK packets are generated by a filtering device and ignore packets that appear to originate from such devices.
not_send_ack_or_syn_ack_packets_during_host_discovery={**0**|1}	(Optional) Specify 1 if you do not want to send TCP ACK or SYN-ACK packets. Out of state TCP packets are not SYN packets and do not belong to an existing TCP session.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X POST
"action=create&title=pcjp&global=1&scan_parallel_scaling=1&scan_overall_p
erformance=high&scan_by_policy=1&policy_names=jp2&auto_update_expected_va
lue=1&scan_ports=standard&additional_tcp_ports=1&not_send_ack_or_syn_ack_
packets_during_host_discovery=1"
"http://qualysapi.qualys.com/api/2.0/fo/subscription/option_profile/pc/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"http://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-04-10T11:10:36Z</DATETIME>
    <TEXT>Compliance Option profile successfully added.</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>39044</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```