Qualys Cloud Platform (VM, PC) v10.x

API Release Notes

Version 10.10

April 16, 2021

This new version of the Qualys Cloud Platform (VM, 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

Compliance Policy API - Added Remediation Information for Control Technologies

Evaluate scan results as a string for Unix File Content Custom Controls

Posture Info API - Filter Output by Last Evaluation Date

VM Host List Detection and VM Host List API - New IPv6 Filter

OS-Authentication-based Data Collection Support for IBM WebSphere Liberty

New Azure MS SQL Record

Qualys API Server URL

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

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

This documentation uses the API server URL for Qualys US Platform 1 (https://qualysapi.qualys.com) in sample API requests. If you're on another platform, please replace this URL with the appropriate server URL for your account.

Compliance Policy API - Added Remediation Information for Control Technologies

APIs affected	/api/2.0/fo/compliance/policy/?action=list
New or Updated API	Updated
DTD or XSD changes	Yes
APIs affected	/api/2.0/fo/compliance/policy/?action=export /api/2.0/fo/compliance/policy/?action=import
New or Updated API	Updated
DTD or XSD changes	No

Starting with this release, users have the ability to customize remediation details for controls from within the Policy Editor UI. Remediation values can be defined for both Service-Defined Controls (SDCs) and User-Defined Controls (UDCs). Each technology for a control can have a different, custom remediation value. To save time, users have the option in the UI to copy control settings, including the remediation value, from one technology to all the other technologies for the same control.

We made the following API changes to support this feature.

Policy List - When you list policies from the API and include details=All, the XML output will include the remediation tag <REMEDIATION> for each technology for each control (SDCs and UDCs). For SDCs, you'll either see the default remediation value set for the control or a custom value defined in the policy. For UDCs, you'll see the custom remediation value whether this was set in the control or in the policy.

Policy Export - When you export policies to XML, we'll now include remediation values for SDCs. We already included remediation values for UDCs. The XML output will show values like below:

- <REMEDIATION></REMEDIATION> appears in the XML output for a SDC that uses the default remediation value, meaning the value has not been customized in the policy. Note that the Policy Export output does not show default remediation values for SDCs.

- <REMEDIATION><![CDATA[user's custom value in policy]]></REMEDIATION> appears in the XML output for a SDC when the remediation value has been customized in the policy and the control no longer uses the default value. You'll see your own custom text.

- <REMEDIATION>user's custom value for UDC</REMEDIATION> appears in the XML output for a UDC when the UDC has a remediation value defined. The remediation value for a UDC can be defined in the control or in the policy. You'll see your own custom text.

- <REMEDIATION><![CDATA[]]></REMEDIATION> appears in the XML output for a UDC when there is no remediation value defined for the UDC.

Policy Import - When you import a policy from XML into your account, we'll include the remediation value for each control technology as defined in the XML. For a SDC with a value of <REMEDIATION></REMEDIATION> we'll use the default value for the control.

Compliance Policy List

When you list compliance policies, the XML output will include the remediation value for each control technology. You'll either see the default remediation value (for SDC) or a custom remediation value (for SDC or UDC).

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"https://qualysapi.qualys.com/api/2.0/fo/compliance/policy/?action=list&i
ds=3748600&details=All"
```

XML Output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE POLICY_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/compliance/policy/policy_list_ou
tput.dtd">
<POLICY_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2021-04-08T23:29:04Z</DATETIME>
    <POLICY_LIST>
      <POLICY>
        <ID>3748600</ID>
        <TITLE><![CDATA[MyPolicy]]></TITLE>
...
          <CONTROL>
            <ID>1071</ID>
            <STATEMENT><![CDATA[Status of the 'Minimum Password Length'
setting]]></STATEMENT>
            <CRITICALITY>
              <LABEL><![CDATA[CRITICAL1]]></LABEL>
              <VALUE>4</VALUE>
            </CRITICALITY>
            <TECHNOLOGY_LIST>
              <TECHNOLOGY>
                <ID>25</ID>
                <NAME>CentOS 4.x</NAME>
                 <RATIONALE><![CDATA[Among the several characteristics that
make 'user identification' via password a secure and workable solution is
setting a 'minimum password length' requirement.  Each character that is
added to the password length squares the difficulty of breaking the
password via 'brute force,' which attempts using every combination
possible within the password symbol set-space, in order to discover a
user's password.  While no 'minimum length' can be guaranteed secure,
```

```
eight (8) is commonly considered to be the minimum for most application
access, along with requiring other password security factors, such as
increasing the size of the symbol set-space by requiring mixed-cases,
along with other forms of password variability creation, increases the
difficulty of breaking any password by brute-force attack.  PASS-MIN-
LEN]]></RATIONALE>
                <CUSTOMIZED>1</CUSTOMIZED>
                <REMEDIATION><![CDATA[user's custom value in
policy]]></REMEDIATION>
              </TECHNOLOGY>
            </TECHNOLOGY_LIST>
          </CONTROL>
          <CONTROL>
            <ID>1072</ID>
            <STATEMENT><![CDATA[Status of the 'Minimum Password Age'
setting]]></STATEMENT>
            <CRITICALITY>
              <LABEL><![CDATA[URGENT1]]></LABEL>
              <VALUE>5</VALUE>
            </CRITICALITY>
            <TECHNOLOGY_LIST>
              <TECHNOLOGY>
                <ID>25</ID>
                <NAME>CentOS 4.x</NAME>
                <RATIONALE><![CDATA[Among the characteristics that make
'user identification' via password a workable security solution is setting
a 'minimum password age.'  Without this minimum age requirement, any
user(s) who wish to re-use the same password can merely cycle through a
number of previously used passwords until returning to the preferred one
(this is determined by the 'Password History' setting).  While no specific
'minimum password age' can guarantee password security, one (1) day is
generally considered to be the shortest length of time permissible, along
with requiring other password security factors, such as increasing the
variability of the symbol set-space by requiring mixed-cases, special
characters, further increases the difficulty of breaking any password
using brute-force methods.  Consider implementing this control for all
account passwords in conjunction with CID 1318 (Password History) and CID
1071 (Minimum Password Length) and CID 1073 (Maximum Password
Age).]]></RATIONALE>
                <CUSTOMIZED>1</CUSTOMIZED>
                <REMEDIATION><![CDATA[To set the value for this setting
edit the '/etc/login.defs' file:
Add or edit the value of 'PASS_MIN_DAYS' setting according to the needs of
business.

Example:
PASS_MIN_DAYS 7

Modify user parameters for all users with a password set to match, with
```

```
the following command:
# chage --mindays 7 <user>]]></REMEDIATION>
              </TECHNOLOGY>
            </TECHNOLOGY_LIST>
          </CONTROL>
          <CONTROL>
            <ID>101554</ID>
            <STATEMENT><![CDATA[DS_Broken symlink]]></STATEMENT>
            <CRITICALITY>
              <LABEL><![CDATA[UNDEFINED]]></LABEL>
              <VALUE>0</VALUE>
            </CRITICALITY>
            <TECHNOLOGY_LIST>
              <TECHNOLOGY>
                <ID>80</ID>
                <NAME>CentOS 7.x</NAME>
                <RATIONALE><![CDATA[rationale statement]]></RATIONALE>
                <CUSTOMIZED>0</CUSTOMIZED>
                <REMEDIATION><![CDATA[]]></REMEDIATION>
              </TECHNOLOGY>
            </TECHNOLOGY_LIST>
          </CONTROL>
          <CONTROL>
            <ID>101569</ID>
            <STATEMENT><![CDATA[File/Directory Existence
Unix]]></STATEMENT>
            <CRITICALITY>
              <LABEL><![CDATA[MINIMUM1]]></LABEL>
              <VALUE>1</VALUE>
            </CRITICALITY>
            <TECHNOLOGY_LIST>
              <TECHNOLOGY>
                <ID>43</ID>
                <NAME>CentOS 6.x</NAME>
                <RATIONALE><![CDATA[.*]]></RATIONALE>
                <CUSTOMIZED>1</CUSTOMIZED>
                <REMEDIATION><![CDATA[user's custom value in
policy]]></REMEDIATION>
              </TECHNOLOGY>
            </TECHNOLOGY_LIST>
          </CONTROL>
          <CONTROL>
            <ID>101574</ID>
            <STATEMENT><![CDATA[Unix Directory Search Check]]></STATEMENT>
            <CRITICALITY>
              <LABEL><![CDATA[URGENT1]]></LABEL>
              <VALUE>5</VALUE>
            </CRITICALITY>
            <TECHNOLOGY_LIST>
```

```
                <TECHNOLOGY>
                  <ID>80</ID>
                  <NAME>CentOS 7.x</NAME>
                  <RATIONALE><![CDATA[.*]]></RATIONALE>
                  <CUSTOMIZED>0</CUSTOMIZED>
                  <REMEDIATION><![CDATA[user's custom value for
UDC]]></REMEDIATION>
                </TECHNOLOGY>
              </TECHNOLOGY_LIST>
            </CONTROL>
          </CONTROL_LIST>
        </POLICY>
      </POLICY_LIST>
...
```

DTD update:

DTD: <platform>/api/2.0/fo/compliance/policy/policy_list_output.dtd

We updated the Policy List Output DTD (policy_list_output.dtd) to include REMEDIATION.

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

...

<!ELEMENT CONTROL_LIST (CONTROL+)>
<!ELEMENT CONTROL (ID, STATEMENT, CRITICALITY?, DEPRECATED?,
TECHNOLOGY_LIST?)>
<!ELEMENT STATEMENT (#PCDATA)>
<!ELEMENT CRITICALITY (LABEL, VALUE)>
<!ELEMENT LABEL (#PCDATA)>
<!ELEMENT DEPRECATED (#PCDATA)>

<!ELEMENT TECHNOLOGY_LIST (TECHNOLOGY+)>
<!ELEMENT TECHNOLOGY (ID, NAME, RATIONALE, CUSTOMIZED, REMEDIATION?)>
<!ELEMENT NAME (#PCDATA)>
<!ELEMENT RATIONALE (#PCDATA)>
<!ELEMENT CUSTOMIZED (#PCDATA)>
<!ELEMENT REMEDIATION (#PCDATA)>

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

Policy Export to XML

When you export compliance policies, the XML output will include the remediation value for each control technology, as described earlier.

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -X "POST" -d
"action=export&id=221469&show_user_controls=1"
"https://qualysapi.qualys.com/api/2.0/fo/compliance/policy/"
```

XML Output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE POLICY_EXPORT_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/compliance/policy/policy_export_
output.dtd">
<POLICY_EXPORT_OUTPUT>
  <RESPONSE>
    <DATETIME>2021-04-02T23:58:52Z</DATETIME>
<POLICY>
    <TITLE><![CDATA[MyPolicy]]></TITLE>
    <EXPORTED><![CDATA[2021-04-02T23:58:52Z]]></EXPORTED>
    <COVER_PAGE><![CDATA[]]></COVER_PAGE>
    <STATUS><![CDATA[active]]></STATUS>
    <TECHNOLOGIES total="16">
        <TECHNOLOGY>
            <ID>2</ID>
            <NAME>Windows 2003 Server</NAME>
        </TECHNOLOGY>
...
    </TECHNOLOGIES>
    <SECTIONS total="1">
        <SECTION>
            <NUMBER>1</NUMBER>
            <HEADING><![CDATA[Untitled]]></HEADING>
            <CONTROLS total="4">
                <CONTROL>
                    <ID>1226</ID>
                    <CRITICALITY>
                        <LABEL><![CDATA[SERIOUS1]]></LABEL>
                        <VALUE>3</VALUE>
                    </CRITICALITY>
                   <IS_CONTROL_DISABLE><![CDATA[0]]></IS_CONTROL_DISABLE>
                    <TECHNOLOGIES total="1">
                        <TECHNOLOGY>
                            <ID>25</ID>
                            <NAME>CentOS 4.x</NAME>

<EVALUATE><CTRL><AND><DP><K>auth.useraccount.etc-csh-login</K><CD>match
```

```
all</CD><OP>re</OP><V><![CDATA[umask\s*077]]></V></DP><DP><K>auth.useracc
ount.ownerships-permissions-etc-csh-
login</K><OP>re</OP><V><![CDATA[.*:.*:[-r][-w][-x][-r][-w][-x][-r][-w][-
x]:/etc/csh.login]]></V><FV
set="1">314159265358979</FV></DP></AND></CTRL></EVALUATE>
                            <REMEDIATION>user's custom value for
UDC</REMEDIATION>
                        </TECHNOLOGY>
                    </TECHNOLOGIES>
                </CONTROL>
                <CONTROL>
                    <ID>1072</ID>
                    <CRITICALITY>
                        <LABEL><![CDATA[URGENT1]]></LABEL>
                        <VALUE>5</VALUE>
                    </CRITICALITY>
                   <IS_CONTROL_DISABLE><![CDATA[0]]></IS_CONTROL_DISABLE>
                    <TECHNOLOGIES total="2">
                        <TECHNOLOGY>
                            <ID>2</ID>
                            <NAME>Windows 2003 Server</NAME>

<EVALUATE><CTRL><DP><K>win.auth.passwords.minage</K><OP>eq</OP><V>1</V><F
V set="0">161803399999999</FV></DP></CTRL></EVALUATE>
                            <REMEDIATION></REMEDIATION>
                        </TECHNOLOGY>
                        <TECHNOLOGY>
                            <ID>12</ID>
                            <NAME>Windows 2000</NAME>

<EVALUATE><CTRL><DP><K>win.auth.passwords.minage</K><OP>eq</OP><V>1</V><F
V set="0">161803399999999</FV></DP></CTRL></EVALUATE>
                            <REMEDIATION></REMEDIATION>
                        </TECHNOLOGY>
                    </TECHNOLOGIES>
                </CONTROL>
                <USER_DEFINED_CONTROL>
                    <ID>101570</ID>
                    <UDC_ID>7293732a-b78a-75d0-80b9-e9df114679d2</UDC_ID>
                    <CHECK_TYPE>Unix File Content Check</CHECK_TYPE>
                   <IS_CONTROL_DISABLE><![CDATA[0]]></IS_CONTROL_DISABLE>
                    <CATEGORY>
                        <ID>9</ID>
                        <NAME><![CDATA[Encryption]]></NAME>
                    </CATEGORY>
                    <SUB_CATEGORY>
                        <ID>1008</ID>
                        <NAME><![CDATA[Email Encryption]]></NAME>
                    </SUB_CATEGORY>
```

```
                    <STATEMENT><![CDATA[File Content Check
unix]]></STATEMENT>
                    <CRITICALITY>
                        <LABEL><![CDATA[MEDIUM1]]></LABEL>
                        <VALUE>2</VALUE>
                    </CRITICALITY>
                    <COMMENT><![CDATA[]]></COMMENT>
                    <USE_AGENT_ONLY>0</USE_AGENT_ONLY>
                    <AUTO_UPDATE>0</AUTO_UPDATE>
                    <IGNORE_ERROR>1</IGNORE_ERROR>
                    <IGNORE_ITEM_NOT_FOUND>1</IGNORE_ITEM_NOT_FOUND>
                    <SCAN_PARAMETERS>
                        <FILE_PATH><![CDATA[/etc/profile]]></FILE_PATH>
                        <FILE_QUERY><![CDATA[^Jun]]></FILE_QUERY>
                        <DATA_TYPE>Line List</DATA_TYPE>
                        <EVALUATE_AS_STRING>0</EVALUATE_AS_STRING>
                        <DESCRIPTION><![CDATA[des]]></DESCRIPTION>
                    </SCAN_PARAMETERS>
                    <TECHNOLOGIES total="1">
                        <TECHNOLOGY>
                            <ID>43</ID>
                            <NAME>CentOS 6.x</NAME>

<EVALUATE><CTRL><DP><K>custom.file_content_check.2917429</K><L>0</L><CD>m
atch any</CD><OP>re</OP><V><![CDATA[true]]></V><FV set="1">item not
found:2</FV></DP></CTRL></EVALUATE>
                            <RATIONALE><![CDATA[.*]]></RATIONALE>
                            <REMEDIATION><![CDATA[]]></REMEDIATION>
                            <DATAPOINT>
                                <CARDINALITY>match any</CARDINALITY>
                                <OPERATOR>re</OPERATOR>
                                <DEFAULT_VALUES total="1">

<DEFAULT_VALUE><![CDATA[true]]></DEFAULT_VALUE>
                                </DEFAULT_VALUES>
                            </DATAPOINT>
                        </TECHNOLOGY>
                    </TECHNOLOGIES>
                    <REFERENCE_LIST/>
                </USER_DEFINED_CONTROL>
                <USER_DEFINED_CONTROL>
                    <ID>101304</ID>
                    <UDC_ID>95ea5cc3-e0da-5f9e-80c2-332eee1adb42</UDC_ID>
                    <CHECK_TYPE>Window File/Directory
Permission</CHECK_TYPE>
                  <IS_CONTROL_DISABLE><![CDATA[0]]></IS_CONTROL_DISABLE>
                    <CATEGORY>
                        <ID>3</ID>
                      <NAME><![CDATA[Access Control Requirements]]></NAME>
```

```
                    </CATEGORY>
                    <SUB_CATEGORY>
                        <ID>1007</ID>
                        <NAME><![CDATA[Authentication/Passwords]]></NAME>
                    </SUB_CATEGORY>
                    <STATEMENT><![CDATA[Basic File/Directory Permission -
WIN]]></STATEMENT>
                    <CRITICALITY>
                        <LABEL><![CDATA[SERIOUS1]]></LABEL>
                        <VALUE>3</VALUE>
                    </CRITICALITY>
                    <COMMENT><![CDATA[File/Directory
Permission]]></COMMENT>
                    <USE_AGENT_ONLY>0</USE_AGENT_ONLY>
                    <AUTO_UPDATE>0</AUTO_UPDATE>
                    <IGNORE_ERROR>0</IGNORE_ERROR>
                    <IGNORE_ITEM_NOT_FOUND>0</IGNORE_ITEM_NOT_FOUND>
                    <SCAN_PARAMETERS>
                        <FILE_PATH><![CDATA[c:\windows]]></FILE_PATH>
                        <DATA_TYPE>String List</DATA_TYPE>
                        <DESCRIPTION><![CDATA[File/Directory
Permission]]></DESCRIPTION>
                    </SCAN_PARAMETERS>
                    <TECHNOLOGIES total="2">
                        <TECHNOLOGY>
                            <ID>75</ID>
                            <NAME>Windows Server 2012 R2</NAME>

<EVALUATE><CTRL><DP><K>custom.file_permission.1921089</K><CD>matches</CD>
<OP>xre</OP><V><![CDATA[.*]]></V></DP></CTRL></EVALUATE>
                            <RATIONALE><![CDATA[File/Directory
Permission]]></RATIONALE>
                            <REMEDIATION><![CDATA[user's custom value in
policy]]></REMEDIATION>
                            <DATAPOINT>
                                <CARDINALITY>matches</CARDINALITY>
                                <OPERATOR>xre</OPERATOR>
                                <DEFAULT_VALUES total="1">

<DEFAULT_VALUE><![CDATA[.*]]></DEFAULT_VALUE>
                                </DEFAULT_VALUES>
                            </DATAPOINT>
                        </TECHNOLOGY>
                        <TECHNOLOGY>
                            <ID>91</ID>
                            <NAME>Windows 10</NAME>

<EVALUATE><CTRL><DP><K>custom.file_permission.1921089</K><CD>matches</CD>
<OP>xre</OP><V><![CDATA[.*]]></V></DP></CTRL></EVALUATE>
```

```
                              <RATIONALE><![CDATA[File/Directory
Permission]]></RATIONALE>
                              <REMEDIATION><![CDATA[user's custom value in
policy]]></REMEDIATION>
                              <DATAPOINT>
                                  <CARDINALITY>matches</CARDINALITY>
                                  <OPERATOR>xre</OPERATOR>
                                  <DEFAULT_VALUES total="1">

<DEFAULT_VALUE><![CDATA[.*]]></DEFAULT_VALUE>
                                  </DEFAULT_VALUES>
                              </DATAPOINT>
                          </TECHNOLOGY>
                      </TECHNOLOGIES>
                      <REFERENCE_LIST/>
                  </USER_DEFINED_CONTROL>
              </CONTROLS>
          </SECTION>
      </SECTIONS>
</POLICY>
  </RESPONSE>
</POLICY_EXPORT_OUTPUT>
```

Policy Import from XML

When you import a policy from XML into your account, we'll include the remediation value for each control technology.

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -H
ContentType:text/xml --data-binary "@policy.xml"
"https://qualysapi.qualys.com/api/2.0/fo/compliance/policy/?action=import
&title=ImportedPolicy&create_user_controls=1"
```

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>2021-04-08T22:51:16Z</DATETIME>
 <TEXT>Successfully imported compliance policy</TEXT>
 <ITEM_LIST>
 <ITEM>
<KEY>ID</KEY>
 <VALUE>1867541</VALUE>
 </ITEM>
```

```
 <ITEM>
 <KEY>TITLE</KEY>
 <VALUE>ImportedPolicy1</VALUE>
 </ITEM>
 </ITEM_LIST>
 </RESPONSE>
</SIMPLE_RETURN>
```

Evaluate scan results as a string for Unix File Content Custom Controls

APIs affected	/api/2.0/fo/compliance/control/ /api/2.0/fo/compliance/policy/
New or Updated API	Updated
DTD or XSD changes	Yes

We added an option to Unix File Content custom controls to evaluate scan results as a string instead string list. Once the <EVALUATE_AS_STRING> parameter is enabled (1), the scan result is evaluated as a single string. By default the option is disabled (0).

Sample: List FC UDC when Evaluate as string is enabled

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X POST -d
"action=list&ids=102090&details=All"
"https://qualysapi.qualys.com/api/2.0/fo/compliance/control/"
```

XML Output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE CONTROL_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/compliance/control/control_list_
output.dtd">
<CONTROL_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2021-04-06T11:14:08Z</DATETIME>
    <CONTROL_LIST>
      <CONTROL>
        <ID>102090</ID>
        <UPDATE_DATE>2021-04-01T11:59:40Z</UPDATE_DATE>
        <CREATED_DATE>2021-04-01T11:59:40Z</CREATED_DATE>
        <CATEGORY>Web Application Services</CATEGORY>
        <SUB_CATEGORY><![CDATA[Web Server/Tier Settings]]></SUB_CATEGORY>
        <STATEMENT><![CDATA[FC_New Option Enabled _With String
list]]></STATEMENT>
        <CRITICALITY>
          <LABEL><![CDATA[URGENT]]></LABEL>
          <VALUE>5</VALUE>
        </CRITICALITY>
        <CHECK_TYPE><![CDATA[Unix File Content Check]]></CHECK_TYPE>
        <COMMENT><![CDATA[String list]]></COMMENT>
        <IGNORE_ERROR>1</IGNORE_ERROR>
        <IGNORE_ITEM_NOT_FOUND>1</IGNORE_ITEM_NOT_FOUND>
        <SCAN_PARAMETERS>
```

```
            <FILE_PATH><![CDATA[/home/testscan/samram]]></FILE_PATH>
            <FILE_QUERY><![CDATA[.*]]></FILE_QUERY>
            <DATA_TYPE>String List</DATA_TYPE>
            <EVALUATE_AS_STRING>1</EVALUATE_AS_STRING>
            <DESCRIPTION><![CDATA[with string list]]></DESCRIPTION>
        </SCAN_PARAMETERS>
        <TECHNOLOGY_LIST>
        ...
```

Sample: List FC UDC when Evaluate as string is disabled

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X POST -d
"action=list&ids=102091&details=All"
"https://qualysapi.qualys.com/api/2.0/fo/compliance/control/"
```

XML Output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE CONTROL_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/compliance/control/control_list_
output.dtd">
<CONTROL_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2021-04-06T11:16:08Z</DATETIME>
    <CONTROL_LIST>
      <CONTROL>
        <ID>102091</ID>
        <UPDATE_DATE>2021-04-01T12:07:02Z</UPDATE_DATE>
        <CREATED_DATE>2021-04-01T12:07:02Z</CREATED_DATE>
        <CATEGORY>Anti-Virus/Malware</CATEGORY>
        <SUB_CATEGORY><![CDATA[Virus/Malware Prevention]]></SUB_CATEGORY>
        <STATEMENT><![CDATA[FC_New Option Disabled_With String
list]]></STATEMENT>
        <CRITICALITY>
          <LABEL><![CDATA[UNDEFINED]]></LABEL>
          <VALUE>0</VALUE>
        </CRITICALITY>
        <CHECK_TYPE><![CDATA[Unix File Content Check]]></CHECK_TYPE>
        <COMMENT><![CDATA[FC_New Option disabld_With String
list]]></COMMENT>
        <IGNORE_ERROR>1</IGNORE_ERROR>
        <IGNORE_ITEM_NOT_FOUND>1</IGNORE_ITEM_NOT_FOUND>
        <SCAN_PARAMETERS>
            <FILE_PATH><![CDATA[/home/testscan/samram]]></FILE_PATH>
            <FILE_QUERY><![CDATA[.*]]></FILE_QUERY>
            <DATA_TYPE>String List</DATA_TYPE>
            <EVALUATE_AS_STRING>0</EVALUATE_AS_STRING>
```

```
            <DESCRIPTION><![CDATA[FC_New Option Disabled_With String
list]]></DESCRIPTION>
        </SCAN_PARAMETERS>
        <TECHNOLOGY_LIST>
       ...
```

Sample: Export Policy where Evaluate as string is enabled

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X POST -d
"action=export&id=3721621&show_user_controls=1"
"https://qualysapi.qualys.com/api/2.0/fo/compliance/policy/"
```

XML Output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE POLICY_EXPORT_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/compliance/policy/policy_export_
output.dtd">
<POLICY_EXPORT_OUTPUT>
  <RESPONSE>
    <DATETIME>2021-04-06T11:56:11Z</DATETIME>
<POLICY>
    <TITLE><![CDATA[Multiline CHeck Oracle asset]]></TITLE>
    <EXPORTED><![CDATA[2021-04-06T11:56:11Z]]></EXPORTED>
    <COVER_PAGE><![CDATA[]]></COVER_PAGE>
    <STATUS><![CDATA[active]]></STATUS>
    <TECHNOLOGIES total="2">
        <TECHNOLOGY>
            <ID>79</ID>
            <NAME>Oracle Enterprise Linux 7.x</NAME>
            ...
                    <CRITICALITY>
                        <LABEL><![CDATA[URGENT]]></LABEL>
                        <VALUE>5</VALUE>
                    </CRITICALITY>
                    <COMMENT><![CDATA[FC UDC]]></COMMENT>
                    <USE_AGENT_ONLY>0</USE_AGENT_ONLY>
                    <AUTO_UPDATE>0</AUTO_UPDATE>
                    <IGNORE_ERROR>1</IGNORE_ERROR>
                    <IGNORE_ITEM_NOT_FOUND>1</IGNORE_ITEM_NOT_FOUND>
                    <SCAN_PARAMETERS>
<FILE_PATH><![CDATA[/home/testscan/samram]]></FILE_PATH>
                        <FILE_QUERY><![CDATA[.*]]></FILE_QUERY>
                        <DATA_TYPE>Line List</DATA_TYPE>
                        <EVALUATE_AS_STRING>1</EVALUATE_AS_STRING>
                       <DESCRIPTION><![CDATA[New option enabled with line
list]]></DESCRIPTION>
```

```
</SCAN_PARAMETERS>
<TECHNOLOGIES total="2">
    <TECHNOLOGY>
        <ID>79</ID>
        <NAME>Oracle Enterprise Linux 7.x</NAME>
       ...
```

DTD update:

The newly added option in the Control List Output (control_list_output.dtd) and Policy Export DTD (policy_export_output.dtd) is highlighted in bold for your reference.

Control List Output

DTD: <platform>/api/2.0/fo/compliance/control/control_list_output.dtd

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

<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
POST_DATA?)>
...
<!ELEMENT SCAN_PARAMETERS (PATH_TYPE?, REG_HIVE?, REG_KEY?,
REG_VALUE_NAME?, FILE_PATH?, FILE_QUERY?, HASH_TYPE?, WMI_NS?,
WMI_QUERY?, SHARE_USER?, PATH_USER?, GROUP_NAME?, GROUP_NAME_LIMIT?,
BASE_DIR?, SHOULD_DESCEND?, DEPTH_LIMIT?, INTEGRITY_CHECK_DEPTH_LIMIT?,
FOLLOW_SYMLINK?, FILE_NAME_MATCH?, FILE_NAME_SKIP?, DIR_NAME_MATCH?,
DIR_NAME_SKIP?, WIN_FILE_SYS_OBJECT_TYPES?,
MATCH_WELL_KNOWN_USERS_FOR_ANY_DOMAIN?, WIN_PERMISSION_USERS?,
WIN_PERMISSION_MATCH?, WIN_PERMISSIONS?, PERMISSIONS?, PERM_COND?,
TYPE_MATCH?, USER_OWNER?, GROUP_OWNER?,
        TIME_LIMIT?, MATCH_LIMIT?, INTEGRITY_CHECK_TIME_LIMIT?,
INTEGRITY_CHECK_MATCH_LIMIT?, INTEGRITY_CHECK_OBJECT_TYPES?,
DIGEST_HASH?, PERMISSION_MONITOR?, DATA_TYPE, EVALUATE_AS_STRING?,
DESCRIPTION)>
<!ELEMENT PATH_TYPE (#PCDATA)>
<!ELEMENT REG_HIVE (#PCDATA)>
<!ELEMENT REG_KEY (#PCDATA)>
<!ELEMENT REG_VALUE_NAME (#PCDATA)>
<!ELEMENT FILE_PATH (#PCDATA)>
<!ELEMENT FILE_QUERY (#PCDATA)>
<!ELEMENT HASH_TYPE (#PCDATA)>
...

<!ELEMENT DATA_TYPE (#PCDATA)>
<!ELEMENT EVALUATE_AS_STRING (#PCDATA)>
<!ELEMENT DESCRIPTION (#PCDATA)>
```

Policy Export Output

DTD: <platform>/api/2.0/fo/compliance/policy/policy_export_output.dtd

```
<!-- QUALYS POLICY_EXPORT_OUTPUT DTD -->
<!-- $Revision: 62328 $ -->
<!ELEMENT POLICY_EXPORT_OUTPUT (REQUEST?, RESPONSE)>

...
<!ELEMENT SCAN_PARAMETERS (PATH_TYPE?, REG_HIVE?, REG_KEY?,
REG_VALUE_NAME?, FILE_PATH?, FILE_QUERY?, HASH_TYPE?, WMI_NS?,
WMI_QUERY?, SHARE_USER?,
PATH_USER?, BASE_DIR?, SHOULD_DESCEND?, DEPTH_LIMIT?,
INTEGRITY_CHECK_DEPTH_LIMIT?, FOLLOW_SYMLINK?, FILE_NAME_MATCH?,
FILE_NAME_SKIP?, DIR_NAME_MATCH?,
DIR_NAME_SKIP?, PERMISSIONS?, PERM_COND?, TYPE_MATCH?, USER_OWNER?,
GROUP_OWNER?, TIME_LIMIT?, MATCH_LIMIT?, INTEGRITY_CHECK_TIME_LIMIT?,
INTEGRITY_CHECK_MATCH_LIMIT?, INTEGRITY_CHECK_OBJECT_TYPES?,
WIN_FILE_SYS_OBJECT_TYPES?,
MATCH_WELL_KNOWN_USERS_FOR_ANY_DOMAIN?, WIN_PERMISSION_USERS?,
WIN_PERMISSION_MATCH?, WIN_PERMISSIONS?, GROUP_NAME?,
GROUP_NAME_LIMIT?, DIGEST_HASH?, PERMISSION_MONITOR?, DATA_TYPE,
EVALUATE_AS_STRING?, DESCRIPTION)>
<!ELEMENT PATH_TYPE (#PCDATA)>
<!ELEMENT REG_HIVE (#PCDATA)>
<!ELEMENT REG_KEY (#PCDATA)>
<!ELEMENT REG_VALUE_NAME (#PCDATA)>
<!ELEMENT FILE_PATH (#PCDATA)>
<!ELEMENT FILE_QUERY (#PCDATA)>
<!ELEMENT HASH_TYPE (#PCDATA)>
<!ELEMENT WMI_NS (#PCDATA)>
<!ELEMENT WMI_QUERY (#PCDATA)>
<!ELEMENT SHARE_USER (#PCDATA)>
...
<!ELEMENT INTEGRITY_CHECK_TIME_LIMIT (#PCDATA)>
<!ELEMENT INTEGRITY_CHECK_MATCH_LIMIT (#PCDATA)>
<!ELEMENT INTEGRITY_CHECK_OBJECT_TYPES (#PCDATA)>
<!ELEMENT DIGEST_HASH (#PCDATA)>
<!ELEMENT PERMISSION_MONITOR (#PCDATA)>
<!ELEMENT WIN_PERMISSION_MATCH (#PCDATA)>
<!ELEMENT MATCH_WELL_KNOWN_USERS_FOR_ANY_DOMAIN (#PCDATA)>
<!ELEMENT WIN_PERMISSION_USERS (#PCDATA)>
<!ELEMENT GROUP_NAME (#PCDATA)>
<!ELEMENT GROUP_NAME_LIMIT (#PCDATA)>
<!ELEMENT DATA_TYPE (#PCDATA)>
<!ELEMENT EVALUATE_AS_STRING (#PCDATA)>
<!ELEMENT DB_QUERY (#PCDATA)>
...
```

Schema update (ImportableControl.xsd):

To support the option to evaluate scan results as a string, we have added the element EVALUATE_AS_STRING to the XSD file.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
...
 <xs:element name="SCAN_PARAMETERS">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="PATH_TYPE" minOccurs="0" maxOccurs="1" />
                 <xs:element ref="REG_HIVE" minOccurs="0" maxOccurs="1" />
                 <xs:element ref="REG_KEY" minOccurs="0" maxOccurs="1" />
...
                 <xs:element ref="PERMISSION_MONITOR" minOccurs="0"
maxOccurs="1" />

                 <xs:element ref="DATA_TYPE" maxOccurs="1" />
                 <xs:element ref="EVALUATE_AS_STRING" minOccurs="0"
maxOccurs="1" />
                 <xs:element ref="DESCRIPTION" maxOccurs="1" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>
...
    <xs:element name="DESCRIPTION">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:minLength value="1"/>
                <xs:maxLength value="1000"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>

    <xs:element name="EVALUATE_AS_STRING" default="0">
        <xs:simpleType>
            <xs:restriction base="xs:integer">
                <xs:enumeration value="0" />
                <xs:enumeration value="1" />
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
...
</xs:schema>
```

Posture Info API - Filter Output by Last Evaluation Date

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

Users already have the ability to filter the posture information list output by the date parameter status_changes_since, which filters results based on the posture's status modified date. Now we're introducing a new date parameter called evaluation_date that allows users to filter the output by the last posture evaluation date/time. The XML output will show the evaluation date. We also updated the Posture Info Output DTD to support this feature. There is no change to CSV output.

Input Parameters

We added the evaluation_date input parameter. Use this input to filter the output based on when the posture was last evaluated. Refer to the Qualys API (VM,PC) User Guide for details on all the possible input parameters for posture information.

Parameter	Description
evaluation_date={date}	(Optional) Show compliance posture info records when the posture evaluation date is equal to or greater than a certain date and time (optional). The date/time is specified in YYYY-MM-DD[THH:MM:SSZ] format (UTC/GMT), like "2021-04-01" or "2021-04-01T23:12:00Z".

Sample - Filtering by evaluation date

In this example, we're filtering the output by an evaluation date of 2021-03-05. The XML output will only include info records with an evaluation date equal to or greater than March 5, 2021.

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=list&policy_id=3318470&details=Basic&output_format=xml&evaluation
_date=2021-03-05"
"https://qualysapi.qualys.com/api/2.0/fo/compliance/posture/info/"
```

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>2021-04-07T21:59:40Z</DATETIME>
    <INFO_LIST>
      <INFO>
        <ID>10911451</ID>
        <HOST_ID>3077710</HOST_ID>
        <CONTROL_ID>1071</CONTROL_ID>
        <TECHNOLOGY_ID>43</TECHNOLOGY_ID>
        <INSTANCE></INSTANCE>
        <STATUS>Passed</STATUS>
        <POSTURE_MODIFIED_DATE>2020-11-
03T07:12:32Z</POSTURE_MODIFIED_DATE>
        <EVALUATION_DATE>2021-04-05T20:36:21Z</EVALUATION_DATE>
        <PREVIOUS_STATUS>Passed</PREVIOUS_STATUS>
        <FIRST_FAIL_DATE>N/A</FIRST_FAIL_DATE>
        <LAST_FAIL_DATE>N/A</LAST_FAIL_DATE>
        <FIRST_PASS_DATE>2020-11-03T07:12:32Z</FIRST_PASS_DATE>
        <LAST_PASS_DATE>2021-04-05T20:36:22Z</LAST_PASS_DATE>
      </INFO>
      <INFO>
        <ID>10911452</ID>
        <HOST_ID>3077710</HOST_ID>
        <CONTROL_ID>1113</CONTROL_ID>
        <TECHNOLOGY_ID>43</TECHNOLOGY_ID>
        <INSTANCE></INSTANCE>
        <STATUS>Failed</STATUS>
        <POSTURE_MODIFIED_DATE>2020-11-
03T07:12:32Z</POSTURE_MODIFIED_DATE>
        <EVALUATION_DATE>2021-04-05T20:36:21Z</EVALUATION_DATE>
        <PREVIOUS_STATUS>Failed</PREVIOUS_STATUS>
        <FIRST_FAIL_DATE>2020-11-03T07:12:32Z</FIRST_FAIL_DATE>
        <LAST_FAIL_DATE>2021-04-05T20:36:22Z</LAST_FAIL_DATE>
        <FIRST_PASS_DATE>N/A</FIRST_PASS_DATE>
        <LAST_PASS_DATE>N/A</LAST_PASS_DATE>
      </INFO>
      <INFO>
        <ID>10911479</ID>
        <HOST_ID>4640713</HOST_ID>
        <CONTROL_ID>1048</CONTROL_ID>
        <TECHNOLOGY_ID>21</TECHNOLOGY_ID>
        <INSTANCE></INSTANCE>
        <STATUS>Passed</STATUS>
        <POSTURE_MODIFIED_DATE>2020-11-
03T07:12:33Z</POSTURE_MODIFIED_DATE>
        <EVALUATION_DATE>2021-03-05T21:35:00Z</EVALUATION_DATE>
        <PREVIOUS_STATUS>Passed</PREVIOUS_STATUS>
        <FIRST_FAIL_DATE>N/A</FIRST_FAIL_DATE>
        <LAST_FAIL_DATE>N/A</LAST_FAIL_DATE>
        <FIRST_PASS_DATE>2020-11-03T07:12:33Z</FIRST_PASS_DATE>
```

```
        <LAST_PASS_DATE>2021-03-05T21:35:00Z</LAST_PASS_DATE>
      </INFO>
      <INFO>
        <ID>10911480</ID>
        <HOST_ID>4640713</HOST_ID>
        <CONTROL_ID>1071</CONTROL_ID>
        <TECHNOLOGY_ID>21</TECHNOLOGY_ID>
        <INSTANCE></INSTANCE>
        <STATUS>Passed</STATUS>
        <POSTURE_MODIFIED_DATE>2020-11-
03T07:12:33Z</POSTURE_MODIFIED_DATE>
        <EVALUATION_DATE>2021-03-05T21:35:00Z</EVALUATION_DATE>
        <PREVIOUS_STATUS>Passed</PREVIOUS_STATUS>
        <FIRST_FAIL_DATE>N/A</FIRST_FAIL_DATE>
        <LAST_FAIL_DATE>N/A</LAST_FAIL_DATE>
        <FIRST_PASS_DATE>2020-11-03T07:12:33Z</FIRST_PASS_DATE>
        <LAST_PASS_DATE>2021-03-05T21:35:00Z</LAST_PASS_DATE>
      </INFO>
...
```

Sample - No filtering by evaluation date

In this sample, we're not filtering the output by evaluation date.

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=list&policy_id=1568053&details=All&output_format=xml&control_ids=
1164" "https://qualysapi.qualys.com/api/2.0/fo/compliance/posture/info/"
```

XML Output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM

<?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>2021-04-03T00:24:22Z</DATETIME>
    <INFO_LIST>
      <INFO>
        <ID>7306298</ID>
        <HOST_ID>2052193</HOST_ID>
        <CONTROL_ID>1164</CONTROL_ID>
        <TECHNOLOGY_ID>2</TECHNOLOGY_ID>
        <INSTANCE></INSTANCE>
        <STATUS>Passed</STATUS>
```

```
        <POSTURE_MODIFIED_DATE>2019-07-
23T16:55:17Z</POSTURE_MODIFIED_DATE>
        <EVALUATION_DATE>2019-08-19T21:13:54Z</EVALUATION_DATE>
        <PREVIOUS_STATUS>Passed</PREVIOUS_STATUS>
        <FIRST_FAIL_DATE>N/A</FIRST_FAIL_DATE>
        <LAST_FAIL_DATE>N/A</LAST_FAIL_DATE>
        <FIRST_PASS_DATE>2019-08-19T21:13:54Z</FIRST_PASS_DATE>
        <LAST_PASS_DATE>2019-08-19T21:13:54Z</LAST_PASS_DATE>
        <EVIDENCE>
          <BOOLEAN_EXPR><![CDATA[(:dp_1 in #fv_1 )]]></BOOLEAN_EXPR>
          <DPV_LIST>
            <DPV lastUpdated="2019-08-16T04:42:54Z">
              <LABEL>:dp_1</LABEL>
              <V><![CDATA[0]]></V>
            </DPV>
          </DPV_LIST>
        </EVIDENCE>
      </INFO>
      <INFO>
        <ID>7306958</ID>
        <HOST_ID>2277184</HOST_ID>
        <CONTROL_ID>1164</CONTROL_ID>
        <TECHNOLOGY_ID>2</TECHNOLOGY_ID>
        <INSTANCE></INSTANCE>
<STATUS>Passed</STATUS>
        <POSTURE_MODIFIED_DATE>2019-07-
23T16:55:30Z</POSTURE_MODIFIED_DATE>
        <EVALUATION_DATE>2019-08-16T05:00:19Z</EVALUATION_DATE>
        <PREVIOUS_STATUS>Passed</PREVIOUS_STATUS>
        <FIRST_FAIL_DATE>N/A</FIRST_FAIL_DATE>
        <LAST_FAIL_DATE>N/A</LAST_FAIL_DATE>
        <FIRST_PASS_DATE>2019-08-16T05:00:19Z</FIRST_PASS_DATE>
        <LAST_PASS_DATE>2019-08-16T05:00:19Z</LAST_PASS_DATE>
        <EVIDENCE>
          <BOOLEAN_EXPR><![CDATA[(:dp_1 in #fv_1 )]]></BOOLEAN_EXPR>
          <DPV_LIST>
            <DPV lastUpdated="2019-08-16T04:42:54Z">
              <LABEL>:dp_1</LABEL>
              <V><![CDATA[0]]></V>
            </DPV>
          </DPV_LIST>
        </EVIDENCE>
      </INFO>
    </INFO_LIST>
```

DTD update:

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

We added EVALUATION_DATE to the Posture Info List Output DTD.

```
<!-- QUALYS POSTURE_INFO_LIST_OUTPUT DTD -->
<!-- $Revision$ -->
<!ELEMENT POSTURE_INFO_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, ((INFO_LIST?, SUMMARY?, WARNING_LIST?,
GLOSSARY?) | POLICY+))>

<!ELEMENT POLICY (ID, DATETIME, INFO_LIST?, SUMMARY?, WARNING_LIST?,
GLOSSARY?)>

<!ELEMENT INFO_LIST (INFO+)>
<!ELEMENT INFO (ID, HOST_ID, CONTROL_ID, TECHNOLOGY_ID, INSTANCE?, STATUS,
REMEDIATION?, POSTURE_MODIFIED_DATE?, EVALUATION_DATE?, PREVIOUS_STATUS?,
FIRST_FAIL_DATE?, LAST_FAIL_DATE?, FIRST_PASS_DATE?, LAST_PASS_DATE?,
EXCEPTION?, EVIDENCE?, CAUSE_OF_FAILURE?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT HOST_ID (#PCDATA)>
<!ELEMENT CONTROL_ID (#PCDATA)>
<!ELEMENT TECHNOLOGY_ID (#PCDATA)>
<!ELEMENT INSTANCE (#PCDATA)>
<!ELEMENT STATUS (#PCDATA)>
<!ELEMENT REMEDIATION (#PCDATA)>
<!ELEMENT POSTURE_MODIFIED_DATE (#PCDATA)>
<!ELEMENT EVALUATION_DATE (#PCDATA)>
...
```

VM Host List Detection and VM Host List API - New IPv6 Filter

APIs affected	/api/2.0/fo/asset/host/vm/detection/?action=list /api/2.0/fo/asset/host/?action=list
New or Updated API	Updated
DTD or XSD changes	No

This release introduces a new input parameter for the VM Host List Detection and Host List API that will allow you to filter hosts based on IPv6 address.

Input Parameters

One new input parameter has been added to VM Host List Detection and VM Host List API.

Parameter	Description
ipv6={value}	(Optional) A valid IPv6 address. Multiple entries are comma separated. If ipv6 is used as filter parameter then other target input filter parameters are not accepted.

Sample - Host List Detection Using IPv6

In this example we have used ipv6 as a input parameter to list hosts with the hosts latest vulnerability data, based on the host based scan data available in the user's account.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"https://qualysapi.qualys.com/api/2.0/fo/asset/host/vm/detection/?action=
list&ipv6=fe80::250:56ff:fe90:aaa0-
fe80::250:56ff:fe90:aaa2&no_vm_scan_since=2015-08-
01&show_cloud_tags=1&show_asset_id=1"
```

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/dtd/outp
ut.dtd">
<HOST_LIST_VM_DETECTION_OUTPUT>
    <RESPONSE>
        <DATETIME>2021-04-08T10:30:01Z</DATETIME>
        <HOST_LIST>
            <HOST>
                <ID>138060</ID>
                <ASSET_ID>159951</ASSET_ID>
                <IP>fe80::250:56ff:fe90:aaa2</IP>
```

```
                <TRACKING_METHOD>IP</TRACKING_METHOD>
                <NETWORK_ID>0</NETWORK_ID>
                <OS>
                    <![CDATA[Cisco VPN 3000 Concentrator]]>
                </OS>
                <LAST_SCAN_DATETIME>2018-01-
12T09:31:39Z</LAST_SCAN_DATETIME>
                <LAST_VM_SCANNED_DATE>2018-01-
12T08:17:55Z</LAST_VM_SCANNED_DATE>
                <LAST_VM_SCANNED_DURATION>139</LAST_VM_SCANNED_DURATION>
                <DETECTION_LIST>
                    <DETECTION>
                        <QID>27288</QID>
                        <TYPE>Confirmed</TYPE>
                        <SEVERITY>3</SEVERITY>
                        <PORT>21</PORT>
                        <PROTOCOL>tcp</PROTOCOL>
                        <SSL>0</SSL>
                        <RESULTS>
                            <![CDATA[220 Session will be terminated after
600 seconds of inactivity.
257-User  access denied.
257 MKD command successful.
550-User  access denied.
550 Not a regular file
250-User  access denied.
250 RMD command successful.]]>
                        </RESULTS>
                        <STATUS>Active</STATUS>
                        <FIRST_FOUND_DATETIME>2017-04-
25T07:03:17Z</FIRST_FOUND_DATETIME>
                        <LAST_FOUND_DATETIME>2018-01-
12T08:17:55Z</LAST_FOUND_DATETIME>
                        <TIMES_FOUND>6</TIMES_FOUND>
                        <LAST_TEST_DATETIME>2018-01-
12T08:17:55Z</LAST_TEST_DATETIME>
                        <LAST_UPDATE_DATETIME>2018-01-
12T09:31:39Z</LAST_UPDATE_DATETIME>
                        <IS_IGNORED>0</IS_IGNORED>
                        <IS_DISABLED>0</IS_DISABLED>
                        <LAST_PROCESSED_DATETIME>2018-01-
12T09:31:39Z</LAST_PROCESSED_DATETIME>
                    </DETECTION>
                    <DETECTION>
                        <QID>38304</QID>
                        <TYPE>Potential</TYPE>
                        <SEVERITY>4</SEVERITY>
                        <PORT>22</PORT>
                        <PROTOCOL>tcp</PROTOCOL>
```

```
                        <SSL>0</SSL>
                        <RESULTS>
                            <![CDATA[SSH-1.5-X]]>
                        </RESULTS>
                        <STATUS>Active</STATUS>
                        <FIRST_FOUND_DATETIME>2017-04-
25T07:03:17Z</FIRST_FOUND_DATETIME>
                        <LAST_FOUND_DATETIME>2018-01-
12T08:17:55Z</LAST_FOUND_DATETIME>
                        <TIMES_FOUND>6</TIMES_FOUND>
                        <LAST_TEST_DATETIME>2018-01-
12T08:17:55Z</LAST_TEST_DATETIME>
                        <LAST_UPDATE_DATETIME>2018-01-
12T09:31:39Z</LAST_UPDATE_DATETIME>
                        <IS_IGNORED>0</IS_IGNORED>
                        <IS_DISABLED>0</IS_DISABLED>
                        <LAST_PROCESSED_DATETIME>2018-01-
12T09:31:39Z</LAST_PROCESSED_DATETIME>
                    </DETECTION>
                    <DETECTION>
                        <QID>82054</QID>
                        <TYPE>Confirmed</TYPE>
                        <SEVERITY>2</SEVERITY>
                        <SSL>0</SSL>
                        <RESULTS>
                            <![CDATA[Tested on port 21 with an injected
SYN/RST offset by 16 bytes.
Tested on port 22 with an injected SYN/RST offset by 16 bytes.]]>
                        </RESULTS>
                        <STATUS>Active</STATUS>
                        <FIRST_FOUND_DATETIME>2017-04-
25T07:03:17Z</FIRST_FOUND_DATETIME>
                        <LAST_FOUND_DATETIME>2018-01-
12T08:17:55Z</LAST_FOUND_DATETIME>
                        <TIMES_FOUND>6</TIMES_FOUND>
                        <LAST_TEST_DATETIME>2018-01-
12T08:17:55Z</LAST_TEST_DATETIME>
                        <LAST_UPDATE_DATETIME>2018-01-
12T09:31:39Z</LAST_UPDATE_DATETIME>
                        <IS_IGNORED>0</IS_IGNORED>
                        <IS_DISABLED>0</IS_DISABLED>
                        <LAST_PROCESSED_DATETIME>2018-01-
12T09:31:39Z</LAST_PROCESSED_DATETIME>
                    </DETECTION>
                </DETECTION_LIST>
            </HOST>
        </HOST_LIST>
    </RESPONSE>
</HOST_LIST_VM_DETECTION_OUTPUT>
```

Sample - Host List Using IPv6

In this example we have used ipv6 as a input parameter to list scanned hosts in the user's account.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"https://qualysapi.qualys.com/api/2.0/fo/asset/host/?action=list&details=
All&ipv6=fe80::250:56ff:fe90:aaa1,2001:558:70:188::1,2001:558:30::433,fe8
0::250:56ff:fe90:aaa0&no_vm_scan_since=2000-08-
01&show_asset_id=1&show_cloud_tags=1&host_metadata_fields=subnetId"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE HOST_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/asset/host/dtd/list/output.dtd">
<HOST_LIST_OUTPUT>
    <RESPONSE>
        <DATETIME>2021-04-12T06:03:44Z</DATETIME>
        <HOST_LIST>
            <HOST>
                <ID>2387203</ID>
                <ASSET_ID></ASSET_ID>
                <IP>fe80::250:56ff:fe90:aaa1</IP>
                <TRACKING_METHOD>IP</TRACKING_METHOD>
                <NETWORK_ID>0</NETWORK_ID>
                <CLOUD_PROVIDER_TAGS>
                    <CLOUD_TAG>
                        <NAME>
                            <![CDATA[Lifecycle]]>
                        </NAME>
                        <VALUE>
                            <![CDATA[20201231]]>
                        </VALUE>
                        <LAST_SUCCESS_DATE>2020-09-
11T00:00:00Z</LAST_SUCCESS_DATE>
                    </CLOUD_TAG>
                    <CLOUD_TAG>
                        <NAME>
                            <![CDATA[JIRA]]>
                        </NAME>
                        <VALUE>
                            <![CDATA[QWEB-1025]]>
                        </VALUE>
                        <LAST_SUCCESS_DATE>2020-09-
11T00:00:00Z</LAST_SUCCESS_DATE>
                    </CLOUD_TAG>
                </CLOUD_PROVIDER_TAGS>
```

```
                <LAST_VULN_SCAN_DATETIME>2018-10-
24T06:55:31Z</LAST_VULN_SCAN_DATETIME>
                <LAST_VM_SCANNED_DATE>2018-10-
24T06:55:31Z</LAST_VM_SCANNED_DATE>
               <LAST_VM_SCANNED_DURATION>70964</LAST_VM_SCANNED_DURATION>
            </HOST>
         </HOST_LIST>
     </RESPONSE>
</HOST_LIST_OUTPUT>
```

OS-Authentication-based Data Collection Support for IBM WebSphere Liberty

APIs affected	/api/2.0/fo/subscription/option_profile/pc/?action=update|create /api/2.0/fo/subscription/option_profile/pc/?action=list api/2.0/fo/subscription/option_profile/?action=export api/2.0/fo/subscription/option_profile/?action=import
New or Updated API	Updated
DTD or XSD changes	Yes

Now you can enable data collection on the IBM WebSphere Liberty instances by using the underlying OS authentication records without creating an authentication record for IBM WebSphere Liberty. Currently, we support the following IBM WebSphere Liberty versions.

- IBM WebSphere Liberty 19.x
- IBM WebSphere Liberty 20.x

For IBM WebSphere Liberty instances running on UNIX host assets, you must create a Unix authentication record with Sudo as root delegation.

We've updated the Option Profile APIs to incorporate this change.

Create/Update Compliance Option Profile

To enable this data collection, while creating or updating a compliance option profile, set the value of the input parameters as mentioned in the following table. Refer to the Qualys API (VM,PC) User Guide for details on all the supported input parameters.

Input Parameters

Parameter	Description
enable_os_based_instance_discovery={0|1}	(Optional) Set the value to 1 to enable data collection on the IBM WebSphere Liberty instances by using underlying OS authentication record. By default, this option is disabled.
os_based_instance_disc_technologies	(Optional) Specify IBM WebSphere Liberty in the comma-separated list of supported technologies to enable OS authentication-based data collection. You can use this parameter only if you set the value of the enable_os_based_instance_discovery parameter to 1.

Sample create compliance option profile

In this sample, we are creating an option profile for data collection on IBM WebSphere Liberty technology instances.

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X POST -d
"action=create&title=OP-
IBM&enable_os_based_instance_discovery=1&os_based_instance_disc_technolog
ies=IBM WebSphere Liberty"&scan_ports=standard
"https://qualysapi.qualys.com/api/2.0/fo/subscription/option_profile/pc/"
```

XML Ouput:

```
<?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>2021-04-09T15:40:06Z</DATETIME>
        <TEXT>Compliance Option profile successfully added.</TEXT>
        <ITEM_LIST>
            <ITEM>
                <KEY>ID</KEY>
                <VALUE>4853557</VALUE>
            </ITEM>
        </ITEM_LIST>
    </RESPONSE>
</SIMPLE_RETURN>
```

Sample update compliance option profile

In this sample, we are updating an existing option profile (ID 4805661) to enable data collection on IBM WebSphere Liberty instances by using the underlying OS authentication record.

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X POST -d
"action=update&id=4805661&enable_os_based_instance_discovery=1&os_based_i
nstance_disc_technologies=IBM WebSphere Liberty"
"https://qualysapi.qualys.com/api/2.0/fo/subscription/option_profile/pc/"
```

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>2021-04-09T15:39:20Z</DATETIME>
        <TEXT>Compliance Option profile successfully updated.</TEXT>
        <ITEM_LIST>
            <ITEM>
                <KEY>ID</KEY>
                <VALUE>4805661</VALUE>
            </ITEM>
        </ITEM_LIST>
    </RESPONSE>
</SIMPLE_RETURN>
```

List Compliance Option Profile

In this sample, we are listing a single option profile specified by profile ID (4847531). In the XML output, you see IBM WebSphere Liberty in the <TECHNOLOGIES> tag inside the <OS_BASED_INSTANCE_DISC_COLLECTION> parent tag.

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X POST -d
"action=list&id=4847531"
"https://qualysapi.qualys.com/api/2.0/fo/subscription/option_profile/pc/"
```

XML Output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE OPTION_PROFILES SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/subscription/option_profile/opti
on_profile_info.dtd">
<OPTION_PROFILES>
    <OPTION_PROFILE>
        <BASIC_INFO>
            <ID>4847531</ID>
            <GROUP_NAME>
                <![CDATA[IBM-Lib-Data-Collection-OS-based-Auth]]>
            </GROUP_NAME>
            <GROUP_TYPE>compliance</GROUP_TYPE>
            <USER_ID>
                <![CDATA[Joe User (joe_user)]]>
            </USER_ID>
            <UNIT_ID>0</UNIT_ID>
            <SUBSCRIPTION_ID>232602</SUBSCRIPTION_ID>
            <IS_GLOBAL>0</IS_GLOBAL>
            <UPDATE_DATE>2021-04-09T07:41:08Z</UPDATE_DATE>
        </BASIC_INFO>
        <SCAN>
            <PORTS>
                <TARGETED_SCAN>1</TARGETED_SCAN>
```

```
            </PORTS>
            <PERFORMANCE>
                <PARALLEL_SCALING>0</PARALLEL_SCALING>
                <OVERALL_PERFORMANCE>Normal</OVERALL_PERFORMANCE>
                <HOSTS_TO_SCAN>
                    <EXTERNAL_SCANNERS>15</EXTERNAL_SCANNERS>
                    <SCANNER_APPLIANCES>30</SCANNER_APPLIANCES>
                </HOSTS_TO_SCAN>
                ...
        <OS_BASED_INSTANCE_DISC_COLLECTION>
            <TECHNOLOGIES>
                <TECHNOLOGY>IBM WebSphere Liberty</TECHNOLOGY>
            </TECHNOLOGIES>
        </OS_BASED_INSTANCE_DISC_COLLECTION>
    </OPTION_PROFILE>
</OPTION_PROFILES>
```

Export Compliance Option Profile:

In this sample, we are exporting a single option profile specified by ID (4847531). In the XML output, you see IBM WebSphere Liberty in the <TECHNOLOGIES> tag inside the <OS_BASED_INSTANCE_DISC_COLLECTION> parent tag.

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -X GET
"https://qualysapi.qualys.com/api/2.0/fo/subscription/option_profile/?act
ion=export&output_format=xml&option_profile_type=compliance&option_profil
e_id=4847531"
```

XML Output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE OPTION_PROFILES SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/subscription/option_profile/opti
on_profile_info.dtd">
<OPTION_PROFILES>
    <OPTION_PROFILE>
        <BASIC_INFO>
            <ID>4847531</ID>
            <GROUP_NAME>
                <![CDATA[IBM-Lib-Data-Collection-OS-based-Auth]]>
            </GROUP_NAME>
            <GROUP_TYPE>compliance</GROUP_TYPE>
            <USER_ID>
                <![CDATA[Joe User (joe_user)]]>
            </USER_ID>
            <UNIT_ID>0</UNIT_ID>
            <SUBSCRIPTION_ID>232602</SUBSCRIPTION_ID>
```

```
            <IS_GLOBAL>0</IS_GLOBAL>
            <UPDATE_DATE>2021-04-09T07:41:08Z</UPDATE_DATE>
        </BASIC_INFO>
        <SCAN>
            ...
        </INSTANCE_DATA_COLLECTION>
        <OS_BASED_INSTANCE_DISC_COLLECTION>
            <TECHNOLOGIES>
                <TECHNOLOGY>Oracle JRE</TECHNOLOGY>
                <TECHNOLOGY>IBM WebSphere Liberty</TECHNOLOGY>
            </TECHNOLOGIES>
        </OS_BASED_INSTANCE_DISC_COLLECTION>
    </OPTION_PROFILE>
</OPTION_PROFILES>
```

Import Compliance Option Profile

In this sample, we are importing an option profile in an input XML file to the user's account.

API Request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With:curl" -H "content-type:
text/xml" -X POST --data-binary @IBM_OP.xml
"https://qualysapi.qualys.com/api/2.0/fo/subscription/option_profile/?act
ion=import"
```

Note: The IBM_OP.xml file contains the request POST data.

Request POST Data:

```
<?xml version="1.0" encoding="UTF-8" ?>
<OPTION_PROFILES>
    <OPTION_PROFILE>
        <BASIC_INFO>
            <ID>4847531</ID>
            <GROUP_NAME>
                <![CDATA[IBM-Lib-Data-Collection-OS-based-Auth]]>
            </GROUP_NAME>
            <GROUP_TYPE>compliance</GROUP_TYPE>
            <USER_ID>
                <![CDATA[[Joe User (joe_user)]]>
            </USER_ID>
            <UNIT_ID>0</UNIT_ID>
            <SUBSCRIPTION_ID>232602</SUBSCRIPTION_ID>
            <IS_GLOBAL>0</IS_GLOBAL>
            <UPDATE_DATE>2021-04-09T07:41:08Z</UPDATE_DATE>
        </BASIC_INFO>
        <SCAN>
```

```
                <PORTS>
                    <TARGETED_SCAN>1</TARGETED_SCAN>
                </PORTS>
                <PERFORMANCE>
                    <PARALLEL_SCALING>0</PARALLEL_SCALING>
                    <OVERALL_PERFORMANCE>Normal</OVERALL_PERFORMANCE>
                    <HOSTS_TO_SCAN>
                        <EXTERNAL_SCANNERS>15</EXTERNAL_SCANNERS>
                        <SCANNER_APPLIANCES>30</SCANNER_APPLIANCES>
                    </HOSTS_TO_SCAN>
                    <PROCESSES_TO_RUN>
                        <TOTAL_PROCESSES>10</TOTAL_PROCESSES>
                        <HTTP_PROCESSES>10</HTTP_PROCESSES>
                    </PROCESSES_TO_RUN>
                    <PACKET_DELAY>Medium</PACKET_DELAY>

<PORT_SCANNING_AND_HOST_DISCOVERY>Normal</PORT_SCANNING_AND_HOST_DISCOVER
Y>
                </PERFORMANCE>
                <DISSOLVABLE_AGENT>
                    <DISSOLVABLE_AGENT_ENABLE>0</DISSOLVABLE_AGENT_ENABLE>
                    <PASSWORD_AUDITING_ENABLE>

<HAS_PASSWORD_AUDITING_ENABLE>0</HAS_PASSWORD_AUDITING_ENABLE>
                    </PASSWORD_AUDITING_ENABLE>

<WINDOWS_SHARE_ENUMERATION_ENABLE>0</WINDOWS_SHARE_ENUMERATION_ENABLE>

<WINDOWS_DIRECTORY_SEARCH_ENABLE>0</WINDOWS_DIRECTORY_SEARCH_ENABLE>
                </DISSOLVABLE_AGENT>
                <FILE_INTEGRITY_MONITORING>
                   <AUTO_UPDATE_EXPECTED_VALUE>0</AUTO_UPDATE_EXPECTED_VALUE>
                </FILE_INTEGRITY_MONITORING>
                <CONTROL_TYPES>
                    <FIM_CONTROLS_ENABLED>0</FIM_CONTROLS_ENABLED>
                    <CUSTOM_WMI_QUERY_CHECKS>0</CUSTOM_WMI_QUERY_CHECKS>
                </CONTROL_TYPES>
            </SCAN>
            <ADDITIONAL>
                <HOST_DISCOVERY>
                    <TCP_PORTS>
                        <STANDARD_SCAN>1</STANDARD_SCAN>
                    </TCP_PORTS>
                    <UDP_PORTS>
                        <STANDARD_SCAN>1</STANDARD_SCAN>
                    </UDP_PORTS>
                    <ICMP>1</ICMP>
                </HOST_DISCOVERY>
                <PACKET_OPTIONS>
```

```
<IGNORE_FIREWALL_GENERATED_TCP_RST>0</IGNORE_FIREWALL_GENERATED_TCP_RST>

<IGNORE_FIREWALL_GENERATED_TCP_SYN_ACK>0</IGNORE_FIREWALL_GENERATED_TCP_S
YN_ACK>

<NOT_SEND_TCP_ACK_OR_SYN_ACK_DURING_HOST_DISCOVERY>0</NOT_SEND_TCP_ACK_OR
_SYN_ACK_DURING_HOST_DISCOVERY>
            </PACKET_OPTIONS>
        </ADDITIONAL>
        <INSTANCE_DATA_COLLECTION>
            <DATABASES>
                <AUTHENTICATION_TYPES_LIST>
                    <AUTHENTICATION_TYPE>MongoDB</AUTHENTICATION_TYPE>
                    <AUTHENTICATION_TYPE>Oracle</AUTHENTICATION_TYPE>
                </AUTHENTICATION_TYPES_LIST>
            </DATABASES>
        </INSTANCE_DATA_COLLECTION>
        <OS_BASED_INSTANCE_DISC_COLLECTION>
            <TECHNOLOGIES>
                <TECHNOLOGY>Oracle JRE</TECHNOLOGY>
                <TECHNOLOGY>IBM WebSphere Liberty</TECHNOLOGY>
            </TECHNOLOGIES>
        </OS_BASED_INSTANCE_DISC_COLLECTION>
    </OPTION_PROFILE>
</OPTION_PROFILES>
```

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>2021-04-22T05:48:33Z</DATETIME>
    <TEXT> Successfully imported Option profile for the subscription Id
218748</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>108689</KEY>
        <VALUE>
                New Option Profile
        </VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Schema Update (option_profiles.xsd)

The option_profiles.xsd schema is used to validate a proper format and required elements of the option profile XML file when importing and exporting option profiles. The tags that we've added to support technology instance data collection by using OS-based authentication records are highlighted in the following option_profiles.xsd extract.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema attributeFormDefault="unqualified"
elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="OPTION_PROFILES" type="OPTION_PROFILESType"/>
    <xs:complexType name="CONTROL_TYPESType">
        <xs:sequence>
...
 <xs:complexType name="OPTION_PROFILEType">
        <xs:sequence>
            <xs:element type="BASIC_INFOType" name="BASIC_INFO"/>
            <xs:element type="SCANType" name="SCAN"/>
            <xs:element type="MAPType" name="MAP" minOccurs="0"/>
            <xs:element type="ADDITIONALType" name="ADDITIONAL"/>
            <xs:element type="INSTANCE_DATA_COLLECTIONType"
name="INSTANCE_DATA_COLLECTION"/>
            <xs:element type="OS_BASED_INSTANCE_DISC_COLLECTIONType"
name="OS_BASED_INSTANCE_DISC_COLLECTION"/>
        </xs:sequence>
    </xs:complexType>
...
<xs:complexType name="OS_BASED_INSTANCE_DISC_COLLECTIONType">
        <xs:sequence>
            <xs:element type="TECHNOLOGIESSType" name="TECHNOLOGIES"
minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="TECHNOLOGIESSType">
        <xs:sequence>
            <xs:element name="TECHNOLOGY" maxOccurs="unbounded">
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:enumeration value="Oracle JRE"/>
                        <xs:enumeration value="IBM WebSphere Liberty"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:schema>
```

New Azure MS SQL Record

APIs affected	/api/2.0/fo/auth/
New or Updated API	No
DTD or XSD changes	Yes
APIs affected	/api/2.0/fo/auth/azure_ms_sql/
New or Updated API	New
DTD or XSD changes	Yes

Azure MS SQL authentication is now supported for only compliance scans. The new Azure MS SQL Authentication API (api/2.0/fo/auth/azure_ms_sql/) lets you list, create, update and delete Azure MS SQL authentication records. User permissions for this API are the same as other authentication record APIs. Note that the API supports only Database authentication. You can use the Auto discover parameter and we'll automatically find databases on your target hosts, so you don't have to provide database information in your record.

API Sample - List All Record Types

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

API request:

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

Response:

```
<?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>2021-04-01T14:15:11Z</DATETIME>
    <AUTH_RECORDS>
      <AUTH_UNIX_IDS>
        <ID_SET>
          <ID>4493178</ID>
          <ID_RANGE>4494671-4494672</ID_RANGE>
          <ID>4494681</ID>
          <ID>4499552</ID>
          <ID>4581443</ID>
          <ID>4588275</ID>
          <ID>4588277</ID>
```

```
            <ID>4595255</ID>
          </ID_SET>
        </AUTH_UNIX_IDS>
        ...
        <AUTH_AZURE_MS_SQL_IDS>
          <ID_SET>
            <ID>4620753</ID>
            <ID_RANGE>4620756-4620757</ID_RANGE>
            <ID>4620763</ID>
          </ID_SET>
        </AUTH_AZURE_MS_SQL_IDS>
      </AUTH_RECORDS>
    </RESPONSE>
</AUTH_RECORDS_OUTPUT>
```

Updated DTD

DTD: <platform>/api/2.0/fo/auth/auth_records.dtd

The element AUTH_AZURE_MS_SQL_IDS was added to identify Azure MS SQL 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_JBOSS_IDS?, AUTH_MARIADB_IDS?, AUTH_INFORMIXDB_IDS?,
AUTH_MS_EXCHANGE_IDS?, AUTH_ORACLE_HTTP_SERVER_IDS?, AUTH_GREENPLUM_IDS?,
AUTH_MICROSOFT_SHAREPOINT_IDS?, AUTH_KUBERNETES_IDS?,
AUTH_SAPIQ_IDS?,AUTH_SAP_HANA_IDS?, AUTH_AZURE_MS_SQL_IDS? )>
...
<!ELEMENT AUTH_MICROSOFT_SHAREPOINT_IDS (ID_SET)>
<!ELEMENT AUTH_KUBERNETES_IDS (ID_SET)>
<!ELEMENT AUTH_SAPIQ_IDS (ID_SET)>
<!ELEMENT AUTH_SAP_HANA_IDS (ID_SET)>
<!ELEMENT AUTH_AZURE_MS_SQL_IDS (ID_SET)>
...
```

List Azure MS SQL Records

Use the new Azure MS SQL Authentication Record List API

(/api/2.0/fo/auth/azure_ms_sql/?action=list) to list Azure MS SQL records.

Input Parameters

Parameter	Description
action=list	(Required) Specify list (using GET or POST) to list records.
details={value}	(Optional) Default value is Basic. You can choose from None, Basic, and All.
ids={value}	(Optional) Azure MS SQL record IDs to list. Specify record IDs and/or ID ranges (for example, 1359-1407). Multiple entries are comma separated.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=list&ids=4620763"
```

Response:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_AZURE_MS_SQL_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/azure_ms_sql/dtd/auth_list_
output.dtd">
<AUTH_AZURE_MS_SQL_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2021-04-01T13:53:08Z</DATETIME>
    <AUTH_AZURE_MS_SQL_LIST>
      <AUTH_AZURE_MS_SQL>
        <ID>4620763</ID>
        <TITLE><![CDATA[AzureMSSQL_Auth_API]]></TITLE>
        <PROVIDER_NAME><![CDATA[Azure]]></PROVIDER_NAME>
        <USERNAME><![CDATA[john_user@qualys.com]]></USERNAME>
        <INSTANCE><![CDATA[MSSQLSERVER]]></INSTANCE>
        <DATABASE><![CDATA[testdb]]></DATABASE>
        <PORT>42</PORT>
        <IP_SET>
          <IP>1.1.1.4</IP>
        </IP_SET>
        <LOGIN_TYPE><![CDATA[basic]]></LOGIN_TYPE>
        <CREATED>
          <DATETIME>2021-04-01T11:47:51Z</DATETIME>
          <BY>up_at</BY>
        </CREATED>
        <LAST_MODIFIED>
```

```
            <DATETIME>2021-04-01T11:47:51Z</DATETIME>
          </LAST_MODIFIED>
        </AUTH_AZURE_MS_SQL>
      </AUTH_AZURE_MS_SQL_LIST>
    </RESPONSE>
</AUTH_AZURE_MS_SQL_LIST_OUTPUT>
```

New DTD:

DTD: <platform>/api/2.0/fo/auth/azure_ms_sql/dtd/auth_list_output.dtd

```
<!ELEMENT AUTH_AZURE_MS_SQL_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_AZURE_MS_SQL_LIST|ID_SET)?,
WARNING_LIST?, GLOSSARY?)>
<!ELEMENT AUTH_AZURE_MS_SQL_LIST (AUTH_AZURE_MS_SQL+)>

<!ELEMENT AUTH_AZURE_MS_SQL (ID, TITLE, PROVIDER_NAME, USERNAME,
INSTANCE, (DATABASE | AUTO_DISCOVER_DATABASES), PORT, IP_SET, LOGIN_TYPE?,
DIGITAL_VAULT?, NETWORK_ID?, CREATED, LAST_MODIFIED, COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT PROVIDER_NAME (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT INSTANCE (#PCDATA)>
<!ELEMENT DATABASE (#PCDATA)>
<!ELEMENT PORT (#PCDATA)>
<!ELEMENT AUTO_DISCOVER_DATABASES (#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?,
VAULT_SECRET_KV_PATH?, VAULT_SECRET_KV_NAME?, VAULT_SECRET_KV_KEY?,
VAULT_SERVICE_TYPE?)>
        <!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 VAULT_SECRET_KV_PATH (#PCDATA)>
        <!ELEMENT VAULT_SECRET_KV_NAME (#PCDATA)>
        <!ELEMENT VAULT_SECRET_KV_KEY (#PCDATA)>
        <!ELEMENT VAULT_SERVICE_TYPE (#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 Azure MS SQLRecord

Use these parameters to create or update a Azure MS SQL record. For an update request, all parameters are optional except “ids” which is required.

Input Parameters

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).
network_id={value}	(Optional and valid when the networks feature is enabled) The network ID for the record.
comments={value}	(Optional to create or update record) User defined comments.Maximum of 1999 characters.
instance_name	(Optional to create or update record) The name of the database instance to be scanned. The default value is MSSQLSERVER. We support only MSSQLSERVER value for this parameter and do not support named instances.
database_name={value}	(Optional to create or update record) The database name of the Azure MS SQL database to be scanned. The database name may contain a maximum of 128 characters. These parameters are mutually exclusive: database_name and auto_discover_databases=1.
auto_discover_databases={**0**|1}	Set auto_discover_databases=1 and we’ll find all Azure MS SQL Server databases on each host. These parameters are mutually exclusive: database_name and auto_discover_databases=1.
port={value}	(Required to create record, optional to update record) The port number assigned to the database instance to be scanned.
Login credentials	
provider_name	(Optional) Name of the cloud service provider. The only value supported is azure.

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 Azure MS SQL. The username must contain '@'.
password={value}	(Required to create record, optional to update record) when login_type=basic, specify the password to be used for authentication to Azure MS SQL.
Vaults	
vault_type={value}	(Required to create record when login_type=vault) The vault type to be used for authentication. Azure MS SQL supports same vault types that are supported by MS SQL. See Vault Support Matrix in the VM/PC API User Guide.
vault_id={value}	(Required only when action=create and login_type=vault) The ID of the vault you want to use.
{vault parameters}	(Required only when action=create and login_type=vault) The vault parameters required depend on the vault type selected. See Vault Definition in the VM/PC API User Guide.
Target Hosts	
ips={value}	Enter a combination of IPs and IP ranges to identify compliance hosts. Multiple entries are comma separated. (Optional to update record) Overwrites (replaces) the IP list for the authentication record. The IPs you specify are added and any existing IPs are removed. For create request, it is required to specify either this parameter or member_domain parameter. For update request, this parameter and the add_ips or remove_ips or member_domain parameter cannot be specified in the same request.
add_ips={value}	(Optional to update record) You may enter a combination of IPs and IP ranges to identify compliance hosts. Multiple entries are comma separated. (Optional to update record) IPs specified will overwrite existing IPs in the record, and existing IPs will be removed.
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.

Example: Create Azure MS SQL Record (with basic login)

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=create&title=my-azuremssql-record&ips=1.1.1.4&port=42
&database_name=dbname"
"https://qualysapi.qualys.com/api/2.0/fo/auth/azure_ms_sql/"
```

Response:

```
<?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>2021-04-01T11:47:51Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>4620763</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Example: Update Azure MS SQL Record (with auto_discover_databases=1)

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&&auto_discover_databases=1&ids=207024"
"https://qualysapi.qualys.com/api/2.0/fo/auth/azure_ms_sql/"
```

Response:

```
<?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>2021-03-26T22:22:41Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Updated</TEXT>
                <ID_SET>
                    <ID>207024</ID>
                </ID_SET>
```

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

Delete Azure MS SQLRecords

Use these parameters to delete records.

Input Parameters

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

Example: Delete Azure MS SQL Records

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=delete&ids=4620768"
"https://qualysapi.qualys.com/api/2.0/fo/auth/azure_ms_sql/"
```

Response:

```
<?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>2021-04-01T13:12:51Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Deleted</TEXT>
        <ID_SET>
          <ID>4620768</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```