Qualys Cloud Platform (VM, PC) v10.x

API Release Notes

Version 10.8

January 21, 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

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.

New SAP HANA Authentication API

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

SAP HANA authentication is now supported for compliance scans (using PC or SCA). The new SAP HANA API (api/2.0/fo/auth/sap_hana/) lets you list, create, update and delete SAP HANA 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/ with action=list) to list authentication records for all types. You'll see <AUTH_SAP_HANA_IDS> in the output when you have SAP HANA records in your account.

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>2020-12-11T10:14:40Z</DATETIME>
    <AUTH_RECORDS>
      <AUTH_UNIX_IDS>
        <ID_SET>
          <ID>2657490</ID>
          <ID>2657492</ID>
          <ID>2667500</ID>
          <ID>3731824</ID>
        </ID_SET>
      </AUTH_UNIX_IDS>
      <AUTH_WINDOWS_IDS>
        <ID_SET>
          <ID>2657491</ID>
          <ID>3862021</ID>
```

```
        </ID_SET>
      </AUTH_WINDOWS_IDS>
      <AUTH_SAP_HANA_IDS>
        <ID_SET>
          <ID_RANGE>185001-185002</ID_RANGE>
        </ID_SET>
      </AUTH_SAP_HANA_IDS>
    </AUTH_RECORDS>
  </RESPONSE>
</AUTH_RECORDS_OUTPUT>
```

Updated DTD

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

The element AUTH_SAP_HANA_IDS has been added to identify SAP HANA record IDs.

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

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

<!ELEMENT RESPONSE (DATETIME, AUTH_RECORDS?, WARNING_LIST?)>

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

<!ELEMENT AUTH_UNIX_IDS (ID_SET)>
<!ELEMENT AUTH_WINDOWS_IDS (ID_SET)>
<!ELEMENT AUTH_ORACLE_IDS (ID_SET)>
```

```
<!ELEMENT AUTH_ORACLE_LISTENER_IDS (ID_SET)>
<!ELEMENT AUTH_SNMP_IDS (ID_SET)>
<!ELEMENT AUTH_MS_SQL_IDS (ID_SET)>
<!ELEMENT AUTH_IBM_DB2_IDS (ID_SET)>
<!ELEMENT AUTH_VMWARE_IDS (ID_SET)>
<!ELEMENT AUTH_MS_IIS_IDS (ID_SET)>
<!ELEMENT AUTH_APACHE_IDS (ID_SET)>
<!ELEMENT AUTH_IBM_WEBSPHERE_IDS (ID_SET)>
<!ELEMENT AUTH_HTTP_IDS (ID_SET)>
<!ELEMENT AUTH_SYBASE_IDS (ID_SET)>
<!ELEMENT AUTH_MYSQL_IDS (ID_SET)>
<!ELEMENT AUTH_TOMCAT_IDS (ID_SET)>
<!ELEMENT AUTH_ORACLE_WEBLOGIC_IDS (ID_SET)>
<!ELEMENT AUTH_DOCKER_IDS (ID_SET)>
<!ELEMENT AUTH_POSTGRESQL_IDS (ID_SET)>
<!ELEMENT AUTH_MONGODB_IDS (ID_SET)>
<!ELEMENT AUTH_PALO_ALTO_FIREWALL_IDS (ID_SET)>
<!ELEMENT AUTH_VCENTER_IDS (ID_SET)>
<!ELEMENT AUTH_JBOSS_IDS (ID_SET)>
<!ELEMENT AUTH_MARIADB_IDS (ID_SET)>
<!ELEMENT AUTH_INFORMIXDB_IDS (ID_SET)>
<!ELEMENT AUTH_MS_EXCHANGE_IDS (ID_SET)>
<!ELEMENT AUTH_ORACLE_HTTP_SERVER_IDS (ID_SET)>
<!ELEMENT AUTH_GREENPLUM_IDS (ID_SET)>
<!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 WARNING_LIST (WARNING+)>
<!ELEMENT WARNING (CODE?, TEXT, URL?, ID_SET?)>
<!ELEMENT CODE (#PCDATA)>
<!ELEMENT TEXT (#PCDATA)>
<!ELEMENT URL (#PCDATA)>

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

<!-- EOF -->
```

List SAP HANA records

Use these parameters to list SAP HANA authentication records.

Parameter	Description
action={action}	(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) SAP HANA auth record IDs to list. Specify record IDs and/or ID ranges (for example, 1359-1407). Multiple entries are comma separated.

Sample - List SAP HANA Records with All Details

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_SAP_HANA_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/sap_hana/auth_sap_hana_list
_output.dtd">
<AUTH_SAP_HANA_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2021-01-12T14:34:42Z</DATETIME>
    <AUTH_SAP_HANA_LIST>
      <AUTH_SAP_HANA>
        <ID>4474042</ID>
        <TITLE><![CDATA[SAP_HANA]]></TITLE>
        <USERNAME><![CDATA[SYSTEM]]></USERNAME>
        <DATABASE><![CDATA[SYSTEMDB]]></DATABASE>
        <PORT>39013</PORT>
        <SSL_VERIFY><![CDATA[1]]></SSL_VERIFY>
        <HOSTS>
          <HOST><![CDATA[host.domain1]]></HOST>
        </HOSTS>
        <IP_SET>
          <IP>10.11.70.185</IP>
        </IP_SET>
        <UNIX_CONF_PATH><![CDATA[/etc/saphana.conf]]></UNIX_CONF_PATH>
        <PASSWORD_ENCRYPTION><![CDATA[1]]></PASSWORD_ENCRYPTION>
        <LOGIN_TYPE><![CDATA[basic]]></LOGIN_TYPE>
        <CREATED>
          <DATETIME>2021-01-12T14:28:16Z</DATETIME>
```

```
            <BY>joe_user</BY>
          </CREATED>
          <LAST_MODIFIED>
            <DATETIME>2021-01-12T14:33:05Z</DATETIME>
          </LAST_MODIFIED>
          <COMMENTS><![CDATA[created successfully]]></COMMENTS>
        </AUTH_SAP_HANA>
      </AUTH_SAP_HANA_LIST>
      <GLOSSARY>
        <USER_LIST>
          <USER>
            <USER_LOGIN>joe_user</USER_LOGIN>
            <FIRST_NAME>Joe</FIRST_NAME>
            <LAST_NAME>User</LAST_NAME>
          </USER>
        </USER_LIST>
      </GLOSSARY>
    </RESPONSE>
</AUTH_SAP_HANA_LIST_OUTPUT>
```

New DTD

<base_url>/api/2.0/fo/auth/sap_hana/auth_sap_hana_list_output.dtd

```
<!-- QUALYS AUTH_SAP_HANA_LIST_OUTPUT DTD -->
<!-- $Revision$ -->
<!ELEMENT AUTH_SAP_HANA_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_SAP_HANA_LIST|ID_SET)?,
WARNING_LIST?, GLOSSARY?)>
<!ELEMENT AUTH_SAP_HANA_LIST (AUTH_SAP_HANA+)>

<!ELEMENT AUTH_SAP_HANA (ID, TITLE,
USERNAME,DATABASE,PORT,SSL_VERIFY?,HOSTS?,IP_SET?,UNIX_CONF_PATH?,PASSWOR
D_ENCRYPTION?,LOGIN_TYPE?,DIGITAL_VAULT?,NETWORK_ID?,CREATED,LAST_MODIFIE
D,COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
```

```
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT DATABASE (#PCDATA)>
<!ELEMENT PORT (#PCDATA)>
<!ELEMENT SSL_VERIFY (#PCDATA)>
<!ELEMENT HOSTS (HOST+)>
<!ELEMENT HOST (#PCDATA)>
<!ELEMENT IP_SET (IP|IP_RANGE)+>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT IP_RANGE (#PCDATA)>
<!ELEMENT UNIX_CONF_PATH (#PCDATA)>
<!ELEMENT PASSWORD_ENCRYPTION (#PCDATA)>

<!ELEMENT LOGIN_TYPE (#PCDATA)>
<!ELEMENT DIGITAL_VAULT (DIGITAL_VAULT_ID, DIGITAL_VAULT_TYPE,
DIGITAL_VAULT_TITLE, VAULT_USERNAME?, VAULT_FOLDER?, VAULT_FILE?,
VAULT_SECRET_NAME?, VAULT_SYSTEM_NAME?, VAULT_NS_TYPE?, VAULT_NS_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_NS_TYPE (#PCDATA)>
<!ELEMENT VAULT_NS_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 SAP HANA Authentication Records

Use these parameters to create or update SAP HANA authentication records.

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.
SAP HANA	
database={value}	(Required for create request) The name of the database you want to authenticate to.
port={value}	(Required for create request) The port the database is on.
unix_conf_path={value}	(Required for create request when this record will be used for scanning Unix hosts) The SAP HANA configuration path on Unix hosts (up to 255 multi-byte characters).
ssl_verify={**0**|1}	(Optional to create or update record) SSL verification is skipped by default. Set to 1 if you want to verify the server's certificate is valid and trusted.
hosts={value}	(Required if ssl_verify=1) A list of FQDNs for 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
Login credentials	
username={value}	(Required for create request) The username of the account to be used for authentication. If password is specified this is the username of a SAP HANA account. If login_type=vault is specified, this is the username of a vault account. Maximum 255 characters (ascii).
password={value}	(For create request, password or login_type=vault is required) The password of the SAP HANA account to be used for authentication. Maximum 100 characters (ascii).
password_encryption={**0**|1}	(Optional to create or update record) Enable this option when your database instance requires an encrypted password for successful login. If password encryption is required and you do not enable this option then authentication will fail. When set to 1, password encryption is enabled in the record. When set to 0 (the default), password encryption is not enabled.
login_type={value}	(For create request, password or login_type=vault is required) Login type can be basic (default) or vault. Set to vault if a third party vault will be used to retrieve the password. Vault parameters need to be provided in the record. See "Vault Definition" in the API user guide.
vault_id={value}	(Required if login_type=vault) The ID of the vault to be used to retrieve the password for login.
vault_type={value}	(Required if login_type=vault) The third party vault to be used to retrieve the password for login. Certain vaults support this capability. See "Vault Support Matrix" in the API user guide.
Target Hosts	
ips={value}	(Required to create record) The IP address(es) for the targets you want to authenticate to. Multiple entries are comma separated. (Optional to update record) IPs specified will overwrite existing IPs in the record, and existing IPs will be removed. This parameter and the add_ips parameter or the remove_ips parameter cannot be specified in the same request.
add_ips={value}	(Optional to update record) Add IPs and/or ranges to the IPs list for this record. Multiple IPs/ranges are comma separated. This parameter and the ips parameter cannot be specified in the same request.

Parameter	Description
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 to create or update record, and valid only when the networks feature is enabled) The network ID for the record.

Sample - Create SAP HANA Record

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=create&title=sap_hana_API&username=root&password=root&database=sa
pDb&port=39013&ips=1.1.1.1&ssl_verify=1&hosts=test.domain.com&unix_conf_p
ath=/etc/saphana.conf&password_encryption=1"
"https://qualysapi.qualys.com/api/2.0/fo/auth/sap_hana/"
```

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>2021-01-12T14:39:46Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>4474043</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Sample - Update SAP HANA Record

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=update&ids=4474043&comments=update1"
"https://qualysapi.qualys.com/api/2.0/fo/auth/sap_hana/"
```

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>2021-01-12T14:45:58Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Updated</TEXT>
        <ID_SET>
          <ID>4474043</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Delete SAP HANA Records

Use the following parameter to delete one or more SAP HANA authentication records.

Parameter	Description
ids={value}	(Required to delete record) SAP HANA auth record IDs to delete. Specify record IDs and/or ID ranges (for example, 1359-1407). Multiple entries are comma separated.

Sample - Delete SAP HANA Records

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=delete&ids=4474043"
"https://qualysapi.qualys.com/api/2.0/fo/auth/sap_hana/"
```

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>2021-01-12T14:48:56Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Deleted</TEXT>
        <ID_SET>
          <ID>4474043</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

IBM WebSphere Authentication - Unix Directory Mode Input

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

When creating and updating IBM WebSphere App Server authentication records, you can now specify the Unix directory mode (installation directory or server directory) using the new input parameter unix_dir_mode. This allows you to create custom authentication records for instances at the installation directory level or the server directory level, similar to system-created authentication records for IBM WebSphere App Server. When listing IBM WebSphere App Server authentication records, you'll see the Unix directory mode setting in the XML output for each record.

Create/Update IBM WebSphere Authentication Record

The following input parameter is new in this release. Refer to the Qualys API (VM,PC) User Guide for details on all the input parameters for authentication records. There are several additional inputs which are not listed here.

Parameter	Description
unix_dir_mode={value}	(Optional) Specify the Unix directory mode. Valid values are installation_dir (for installation directory) and server_dir (for server directory). When not specified, installation_dir is used.

Sample - Create IBM WebSphere Record (installation dir)

In this example, we'll create an IBM WebSphere authentication record with the Unix directory mode set to installation directory.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=create&title=ibm_was_installation_dir&ips=10.11.71.33&unix_instal
lation_dir=/usr/local/ibm/&unix_dir_mode=installation_dir"
"https://qualysapi.qualys.com/api/2.0/fo/auth/ibm_websphere/"
```

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>2021-01-05T13:38:15Z</DATETIME>
    <BATCH_LIST>
```

```
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>4467949</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Sample - Create IBM WebSphere Record (server dir)

In this example, we'll create an IBM WebSphere authentication record with the Unix directory mode set to server directory.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=create&title=ibm_was_server_dir&ips=10.11.71.33&unix_installation
_dir=/usr/local/ibm/&unix_dir_mode=server_dir"
"https://qualysapi.qualys.com/api/2.0/fo/auth/ibm_websphere/"
```

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>2021-01-05T13:38:15Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>4467947</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Sample - Update IBM WebSphere Record

In this example, we'll update a IBM WebSphere record to change the Unix directory mode and Unix installation directory.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl" -d
"action=update&ids=4467949&unix_dir_mode=server_dir&unix_installation_dir
=/update/dir/"
"https://qualysapi.qualys.com/api/2.0/fo/auth/ibm_websphere/"
```

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>2021-01-05T14:05:33Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Updated</TEXT>
        <ID_SET>
          <ID>4467949</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

List IBM WebSphere Authentication Records

When you list IBM WebSphere authentication records, you'll see <UNIX_DIR_MODE> in the XML output which identifies the setting for each record (installation_dir or server_dir).

Sample - List IBM WebSphere Records

In this example, you'll see different IBM WebSphere records listed with the Unix directory mode value for each record.

API request:

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

XML output:

```
<!DOCTYPE AUTH_IBM_WEBSPHERE_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/ibm_websphere/auth_ibm_webs
phere_list_output.dtd">
```

```
<AUTH_IBM_WEBSPHERE_LIST_OUTPUT>
 <RESPONSE>
    <DATETIME>2021-01-05T13:46:51Z</DATETIME>
    <AUTH_IBM_WEBSPHERE_LIST>
      <AUTH_IBM_WEBSPHERE>
        <ID>4467947</ID>
        <TITLE><![CDATA[ibm_was_server_dir]]></TITLE>
        <IP_SET>
          <IP>10.11.71.33</IP>
        </IP_SET>

<UNIX_INSTLLATION_DIRECTORY><![CDATA[/usr/local/ibm/]]></UNIX_INSTLLATION
_DIRECTORY>
        <UNIX_DIR_MODE>server_dir</UNIX_DIR_MODE>
        <NETWORK_ID>0</NETWORK_ID>
        <CREATED>
          <DATETIME>2021-01-05T13:33:06Z</DATETIME>
          <BY>joe_user</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2021-01-05T13:33:06Z</DATETIME>
        </LAST_MODIFIED>
        <IS_SYSTEM_CREATED>0</IS_SYSTEM_CREATED>
        <IS_ACTIVE>1</IS_ACTIVE>
      </AUTH_IBM_WEBSPHERE>
      <AUTH_IBM_WEBSPHERE>
        <ID>4467949</ID>
        <TITLE><![CDATA[ibm_was_installation_dir]]></TITLE>
        <IP_SET>
          <IP>10.11.71.33</IP>
        </IP_SET>

<UNIX_INSTLLATION_DIRECTORY><![CDATA[/usr/local/ibm/]]></UNIX_INSTLLATION
_DIRECTORY>
        <UNIX_DIR_MODE>installation_dir</UNIX_DIR_MODE>
        <NETWORK_ID>0</NETWORK_ID>
        <CREATED>
          <DATETIME>2021-01-05T13:38:15Z</DATETIME>
          <BY>joe_user</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2021-01-05T13:38:15Z</DATETIME>
        </LAST_MODIFIED>
        <IS_SYSTEM_CREATED>0</IS_SYSTEM_CREATED>
        <IS_ACTIVE>1</IS_ACTIVE>
      </AUTH_IBM_WEBSPHERE>
   </AUTH_IBM_WEBSPHERE_LIST>
  </RESPONSE>
</AUTH_IBM_WEBSPHERE_LIST_OUTPUT>
```

DTD update:

<platform>/api/2.0/fo/auth/ibm_websphere/auth_ibm_websphere_list_output.dtd

```
<!-- QUALYS AUTH_IBM_WEBSPHERE_LIST_OUTPUT DTD -->
<!-- $Revision: $ -->
<!ELEMENT AUTH_IBM_WEBSPHERE_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_IBM_WEBSPHERE_LIST|ID_SET)?,
WARNING_LIST?, GLOSSARY?)>
<!ELEMENT AUTH_IBM_WEBSPHERE_LIST (AUTH_IBM_WEBSPHERE+)>

<!ELEMENT AUTH_IBM_WEBSPHERE (ID, TITLE, IP_SET,
UNIX_INSTLLATION_DIRECTORY?,UNIX_DIR_MODE?,
WINDOWS_INSTLLATION_DIRECTORY?, NETWORK_ID?, CREATED, LAST_MODIFIED,
IS_SYSTEM_CREATED?, IS_ACTIVE?, COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT UNIX_INSTLLATION_DIRECTORY (#PCDATA)>
<!ELEMENT UNIX_DIR_MODE (#PCDATA)>
<!ELEMENT WINDOWS_INSTLLATION_DIRECTORY (#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 IS_SYSTEM_CREATED (#PCDATA)>
<!ELEMENT IS_ACTIVE (#PCDATA)>
<!ELEMENT COMMENTS (#PCDATA)>
...
```

VM Host List Detection API and Host List API - Change to DNS Data in Output

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	Yes

For the VM Host List Detection API and the Host List API, we changed the way DNS data is presented in the XML output. Previously, we had a single tag for <DNS> where we showed either the DNS hostname (e.g. xpsp2-64-24-84) or the DNS hostname with the FQDN (e.g. xpsp2-64-24-84.sample.qualys.com). When scanning Windows hosts with both a scanner and an agent, the DNS value could switch between hostname only (from agent) and hostname with FQDN (from scanner).

Starting in this release, we will continue to show the <DNS> tag, but we've also added <DNS_DATA> where we'll show separate values for hostname, domain and FQDN so you can more easily parse this data and avoid confusion. See the example below:

```
<DNS>
    <![CDATA[xpsp2-64-24-84.sample.qualys.com]]>
</DNS>
<DNS_DATA>
    <HOSTNAME>
        <![CDATA[xpsp2-64-24-84]]>
    </HOSTNAME>
    <DOMAIN>
        <![CDATA[sample.qualys.com]]>
    </DOMAIN>
    <FQDN>
        <![CDATA[xpsp2-64-24-84.sample.qualys.com]]>
    </FQDN>
</DNS_DATA>
```

Host List API

Here's a sample API call and output for Host List.

API request:

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

XML output:

```
<!DOCTYPE HOST_LIST_OUTPUT SYSTEM
```

```
"http://qualysapi.qualys.com/api/2.0/fo/asset/host/dtd/output.dtd">
<HOST_LIST_OUTPUT>
    <RESPONSE>
        <DATETIME>2021-01-21T08:23:54Z</DATETIME>
        <HOST_LIST>
            <HOST>
                <ID>7933740</ID>
                <IP>10.4.4.1</IP>
                <TRACKING_METHOD>IP</TRACKING_METHOD>
                <NETWORK_ID>63010</NETWORK_ID>
                <DNS>
                    <![CDATA[10-4-4-1.sample.qualys.com]]>
                </DNS>
                <DNS_DATA>
                    <HOSTNAME>
                        <![CDATA[10-4-4-1]]>
                    </HOSTNAME>
                    <DOMAIN>
                        <![CDATA[sample.qualys.com]]>
                    </DOMAIN>
                    <FQDN>
                        <![CDATA[10-4-4-1.sample.qualys.com]]>
                    </FQDN>
                </DNS_DATA>
                <OS>
                   <![CDATA[Linux 2.4-2.6 / Embedded Device / F5 Networks
Big-IP]]>
                </OS>
            </HOST>
        </HOST_LIST>
    </RESPONSE>
</HOST_LIST_OUTPUT>
```

DTD update:

You'll now see DNS_DATA in the DTD for the Host List API. Please also note that the DTD has been renamed to follow a new naming convention.The new DTD name is:

<platform>/api/2.0/fo/asset/host/dtd/output.dtd

```
<!-- QUALYS HOST_OUTPUT DTD -->
<!-- $Revision$ -->
<!ELEMENT HOST_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, (HOST_LIST|ID_SET)?, WARNING?, GLOSSARY?)>
<!ELEMENT HOST_LIST (HOST+)>
<!ELEMENT HOST (ID, ASSET_ID?, IP?, TRACKING_METHOD?, NETWORK_ID?,
                DNS?, DNS_DATA?, CLOUD_PROVIDER?, CLOUD_SERVICE?,
                CLOUD_RESOURCE_ID?, EC2_INSTANCE_ID?, NETBIOS?, OS?,
                QG_HOSTID?, TAGS?, METADATA?,
                CLOUD_PROVIDER_TAGS?, LAST_VULN_SCAN_DATETIME?,
                LAST_VM_SCANNED_DATE?, LAST_VM_SCANNED_DURATION?,
                LAST_VM_AUTH_SCANNED_DATE?,
                LAST_VM_AUTH_SCANNED_DURATION?,
                LAST_COMPLIANCE_SCAN_DATETIME?,
                LAST_SCAP_SCAN_DATETIME?, OWNER?, COMMENTS?, USER_DEF?,
                ASSET_GROUP_IDS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT ASSET_ID (#PCDATA)>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT TRACKING_METHOD (#PCDATA)>
<!ELEMENT NETWORK_ID (#PCDATA)>
<!ELEMENT DNS (#PCDATA)>
<!ELEMENT DNS_DATA (HOSTNAME?, DOMAIN?, FQDN?)>
<!ELEMENT HOSTNAME (#PCDATA)>
<!ELEMENT DOMAIN (#PCDATA)>
<!ELEMENT FQDN (#PCDATA)>
<!ELEMENT EC2_INSTANCE_ID (#PCDATA)>
<!ELEMENT CLOUD_PROVIDER (#PCDATA)>
<!ELEMENT CLOUD_SERVICE (#PCDATA)>
<!ELEMENT CLOUD_RESOURCE_ID (#PCDATA)>
...
```

VM Host List Detection API

Here's a sample API call and output for VM Host List Detection.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl"
"https://qualysapi.qualys.com/api/2.0/fo/asset/host/vm/detection/?action=
list&ips=10.4.4.1&output_format=xml"
```

XML output:

```
<!DOCTYPE HOST_LIST_VM_DETECTION_OUTPUT SYSTEM
"http://qualysapi.qualys.com/api/2.0/fo/asset/host/vm/detection/dtd/outpu
```

```
t.dtd">
<HOST_LIST_VM_DETECTION_OUTPUT>
    <RESPONSE>
        <DATETIME>2021-01-21T08:30:43Z</DATETIME>
        <HOST_LIST>
            <HOST>
                <ID>7933740</ID>
                <IP>10.4.4.1</IP>
                <TRACKING_METHOD>IP</TRACKING_METHOD>
                <NETWORK_ID>63010</NETWORK_ID>
                <OS>
                   <![CDATA[Linux 2.4-2.6 / Embedded Device / F5 Networks
Big-IP]]>
                </OS>
                <DNS>
                    <![CDATA[10-4-4-1.sample.qualys.com]]>
                </DNS>
                <DNS_DATA>
                    <HOSTNAME>
                        <![CDATA[10-4-4-1]]>
                    </HOSTNAME>
                    <DOMAIN>
                        <![CDATA[sample.qualys.com]]>
                    </DOMAIN>
                    <FQDN>
                        <![CDATA[10-4-4-1.sample.qualys.com]]>
                    </FQDN>
                </DNS_DATA>
                <LAST_SCAN_DATETIME>2020-11-
06T06:26:36Z</LAST_SCAN_DATETIME>
                <LAST_VM_SCANNED_DATE>2020-11-
06T06:27:27Z</LAST_VM_SCANNED_DATE>
                <LAST_VM_SCANNED_DURATION>10</LAST_VM_SCANNED_DURATION>
                <DETECTION_LIST>
...
```

DTD update:

You'll now see DNS_DATA in the DTD for the VM Host List Detection API. Please also note that the DTD has been renamed to follow a new naming convention. New DTD name:

<platform>/api/2.0/fo/asset/host/vm/detection/dtd/output.dtd

```
<!-- QUALYS HOST_LIST_VM_DETECTION_OUTPUT DTD -->
<!-- $Revision$ -->
<!ELEMENT HOST_LIST_VM_DETECTION_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, HOST_LIST?, WARNING?)>
<!ELEMENT HOST_LIST (HOST+)>
<!ELEMENT HOST (ID, ASSET_ID?, IP?, IPV6?, TRACKING_METHOD?, NETWORK_ID?,
                OS?, OS_CPE?, DNS?, DNS_DATA?, CLOUD_PROVIDER?,
                CLOUD_SERVICE?, CLOUD_RESOURCE_ID?, EC2_INSTANCE_ID?,
                NETBIOS?, QG_HOSTID?,
                LAST_SCAN_DATETIME?, LAST_VM_SCANNED_DATE?,
                LAST_VM_SCANNED_DURATION?, LAST_VM_AUTH_SCANNED_DATE?,
                LAST_VM_AUTH_SCANNED_DURATION?,
                LAST_PC_SCANNED_DATE?, TAGS?, METADATA?,
                CLOUD_PROVIDER_TAGS?, DETECTION_LIST)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT ASSET_ID (#PCDATA)>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT IPV6 (#PCDATA)>
<!ELEMENT TRACKING_METHOD (#PCDATA)>
<!ELEMENT NETWORK_ID (#PCDATA)>
<!ELEMENT OS (#PCDATA)>
<!ELEMENT OS_CPE (#PCDATA)>
<!ELEMENT DNS (#PCDATA)>
<!ELEMENT DNS_DATA (HOSTNAME?, DOMAIN?, FQDN?)>
<!ELEMENT HOSTNAME (#PCDATA)>
<!ELEMENT DOMAIN (#PCDATA)>
<!ELEMENT FQDN (#PCDATA)>
<!ELEMENT CLOUD_PROVIDER (#PCDATA)>
<!ELEMENT CLOUD_SERVICE (#PCDATA)>
<!ELEMENT CLOUD_RESOURCE_ID (#PCDATA)>
...
```

VM Host List Detection API - New Superseded QIDs Filter

APIs affected	/api/2.0/fo/asset/host/vm/detection/?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 API that will allow you to filter out any QID that has been superseded by another QID detected on the same host. When a QID is filtered out of the results because it has been superseded, we'll remove the entire <DETECTION></DETECTION> block for that QID. This way the results are more streamlined and will show only the QIDs you need to fix. (Please note that there are DTD changes for VM Host List Detection API for the feature VM Host List Detection API and Host List API - Change to DNS Data in Output described earlier in this document.)

Input Parameters

One new input parameter has been added to VM Detection API to filter superseded QIDs.

Parameter	Description
filter_superseded_qids={0|1}	(Optional) When unspecified or set to 0, the XML output includes all QIDs even if they've been superseded. Specify 1 to filter out QIDs that have been superseded by another QID in the results.

Sample - Filter superseded QIDs (filter_superseded_qids=1)

In this example any QID superseded by another QID has been filtered out of the results. The XML output includes QID 370584 and QID 370613. QID 370610 was filtered out because it was superseded by QID 370613.

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&filter_superseded_qids=1"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE HOST_LIST_VM_DETECTION_OUTPUT SYSTEM
"http://qualysapi.qualys.com/api/2.0/fo/asset/host/vm/detection/dtd/outpu
t.dtd">
<HOST_LIST_VM_DETECTION_OUTPUT>
  <RESPONSE>
    <DATETIME>2020-06-03T10:22:34Z</DATETIME>
    <HOST_LIST>
      <HOST>
```

```
        <ID>1145</ID>
        <IP>10.10.10.9</IP>
        <TRACKING_METHOD>IP</TRACKING_METHOD>
        <OS><![CDATA[Windows 2003 Service Pack 2]]></OS>
        <DNS><![CDATA[win2003.sample.qualys.com]]></DNS>
        <DNS_DATA>
            <HOSTNAME>
                <![CDATA[win2003]]>
            </HOSTNAME>
            <DOMAIN>
                <![CDATA[sample.qualys.com]]>
            </DOMAIN>
            <FQDN>
                <![CDATA[win2003.sample.qualys.com]]>
            </FQDN>
        </DNS_DATA>
        <NETBIOS><![CDATA[LWIN2003HP1]]></NETBIOS>
        <LAST_SCAN_DATETIME>2018-01-08T19:50:18Z</LAST_SCAN_DATETIME>
        <LAST_VM_SCANNED_DATE>2018-01-08T19:36:29Z</LAST_VM_SCANNED_DATE>
        <LAST_VM_SCANNED_DURATION>619</LAST_VM_SCANNED_DURATION>
        <LAST_PC_SCANNED_DATE>2017-11-15T16:58:16Z</LAST_PC_SCANNED_DATE>
        <DETECTION_LIST>
<DETECTION>
            <QID>370584</QID>
            <TYPE>Confirmed</TYPE>
            <SEVERITY>5</SEVERITY>
            <SSL>0</SSL>
          <RESULTS><![CDATA[C:\Program Files\Mozilla Firefox\firefox.exe
Version is  42.0.0.0]]></RESULTS>
            <STATUS>Active</STATUS>
            <FIRST_FOUND_DATETIME>2017-10-
10T10:30:48Z</FIRST_FOUND_DATETIME>
            <LAST_FOUND_DATETIME>2020-04-
27T23:04:10Z</LAST_FOUND_DATETIME>
            <TIMES_FOUND>183</TIMES_FOUND>
            <LAST_TEST_DATETIME>2020-04-27T23:04:10Z</LAST_TEST_DATETIME>
            <LAST_UPDATE_DATETIME>2020-04-
27T23:05:41Z</LAST_UPDATE_DATETIME>
            <LAST_FIXED_DATETIME>2019-08-
27T22:48:04Z</LAST_FIXED_DATETIME>
            <IS_IGNORED>0</IS_IGNORED>
            <IS_DISABLED>0</IS_DISABLED>
            <LAST_PROCESSED_DATETIME>2020-04-
27T23:05:41Z</LAST_PROCESSED_DATETIME>
          </DETECTION>
<DETECTION>
            <QID>370613</QID>
            <TYPE>Confirmed</TYPE>
            <SEVERITY>5</SEVERITY>
```

```
            <SSL>0</SSL>
            <RESULTS><![CDATA[C:\Program
Files\Google\Chrome\Application\33.0.1750.149\chrome.dll file version is
33.0.1750.149
%ProgramFiles%\Google\Chrome\Application\33.0.1750.149\chrome.dll file
version is 33.0.1750.149]]></RESULTS>
            <STATUS>Active</STATUS>
            <FIRST_FOUND_DATETIME>2017-11-
12T20:11:32Z</FIRST_FOUND_DATETIME>
            <LAST_FOUND_DATETIME>2020-04-
27T23:04:10Z</LAST_FOUND_DATETIME>
            <TIMES_FOUND>162</TIMES_FOUND>
            <LAST_TEST_DATETIME>2020-04-27T23:04:10Z</LAST_TEST_DATETIME>
            <LAST_UPDATE_DATETIME>2020-04-
27T23:05:41Z</LAST_UPDATE_DATETIME>
            <LAST_FIXED_DATETIME>2019-10-
30T22:28:59Z</LAST_FIXED_DATETIME>
            <IS_IGNORED>0</IS_IGNORED>
            <IS_DISABLED>0</IS_DISABLED>
            <LAST_PROCESSED_DATETIME>2020-04-
27T23:05:41Z</LAST_PROCESSED_DATETIME>
          </DETECTION>
</DETECTION_LIST>
      </HOST>
    </HOST_LIST>
  </RESPONSE>
</HOST_LIST_VM_DETECTION_OUTPUT>
```

Sample - Do not filter superseded QIDs (filter_superseded_qids not specified)

In this example all the QIDs are included in the output even if they have been superseded. The XML output includes QID 370584, QID 370610 and QID 370613. QID 370610 appears in the results even though it is superseded by QID 370613.

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"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE HOST_LIST_VM_DETECTION_OUTPUT SYSTEM
"http://qualysapi.qualys.com/api/2.0/fo/asset/host/vm/detection/dtd/outpu
t.dtd">
<HOST_LIST_VM_DETECTION_OUTPUT>
  <RESPONSE>
    <DATETIME>2020-06-03T10:22:03Z</DATETIME>
```

```
    <HOST_LIST>
      <HOST>
        <ID>1145</ID>
        <IP>10.10.10.9</IP>
        <TRACKING_METHOD>IP</TRACKING_METHOD>
        <OS><![CDATA[Windows 2003 Service Pack 2]]></OS>
        <DNS><![CDATA[win2003.sample.qualys.com]]></DNS>
        <DNS_DATA>
            <HOSTNAME>
                <![CDATA[win2003]]>
            </HOSTNAME>
            <DOMAIN>
                <![CDATA[sample.qualys.com]]>
            </DOMAIN>
            <FQDN>
                <![CDATA[win2003.sample.qualys.com]]>
            </FQDN>
        </DNS_DATA>
        <NETBIOS><![CDATA[LWIN2003HP1]]></NETBIOS>
        <LAST_SCAN_DATETIME>2018-01-08T19:50:18Z</LAST_SCAN_DATETIME>
        <LAST_VM_SCANNED_DATE>2018-01-08T19:36:29Z</LAST_VM_SCANNED_DATE>
        <LAST_VM_SCANNED_DURATION>619</LAST_VM_SCANNED_DURATION>
        <LAST_PC_SCANNED_DATE>2017-11-15T16:58:16Z</LAST_PC_SCANNED_DATE>
        <DETECTION_LIST>
<DETECTION>
            <QID>370584</QID>
            <TYPE>Confirmed</TYPE>
            <SEVERITY>5</SEVERITY>
            <SSL>0</SSL>
          <RESULTS><![CDATA[C:\Program Files\Mozilla Firefox\firefox.exe
Version is  42.0.0.0]]></RESULTS>
            <STATUS>Active</STATUS>
            <FIRST_FOUND_DATETIME>2017-10-
10T10:30:48Z</FIRST_FOUND_DATETIME>
            <LAST_FOUND_DATETIME>2020-04-
27T23:04:10Z</LAST_FOUND_DATETIME>
            <TIMES_FOUND>183</TIMES_FOUND>
            <LAST_TEST_DATETIME>2020-04-27T23:04:10Z</LAST_TEST_DATETIME>
            <LAST_UPDATE_DATETIME>2020-04-
27T23:05:41Z</LAST_UPDATE_DATETIME>
            <LAST_FIXED_DATETIME>2019-08-
27T22:48:04Z</LAST_FIXED_DATETIME>
            <IS_IGNORED>0</IS_IGNORED>
            <IS_DISABLED>0</IS_DISABLED>
            <LAST_PROCESSED_DATETIME>2020-04-
27T23:05:41Z</LAST_PROCESSED_DATETIME>
          </DETECTION>
<DETECTION>
            <QID>370610</QID>
```

```
            <TYPE>Confirmed</TYPE>
            <SEVERITY>5</SEVERITY>
            <SSL>0</SSL>
           <RESULTS><![CDATA[C:\Program Files\Java\jre7\bin\wsdetect.dll
file version is 7.0.510.13]]></RESULTS>
            <STATUS>Active</STATUS>
            <FIRST_FOUND_DATETIME>2017-11-
12T20:11:32Z</FIRST_FOUND_DATETIME>
            <LAST_FOUND_DATETIME>2020-04-
27T23:04:10Z</LAST_FOUND_DATETIME>
            <TIMES_FOUND>175</TIMES_FOUND>
            <LAST_TEST_DATETIME>2020-04-27T23:04:10Z</LAST_TEST_DATETIME>
            <LAST_UPDATE_DATETIME>2020-04-
27T23:05:41Z</LAST_UPDATE_DATETIME>
            <LAST_FIXED_DATETIME>2020-04-
10T22:10:18Z</LAST_FIXED_DATETIME>
            <IS_IGNORED>0</IS_IGNORED>
            <IS_DISABLED>0</IS_DISABLED>
            <LAST_PROCESSED_DATETIME>2020-04-
27T23:05:41Z</LAST_PROCESSED_DATETIME>
          </DETECTION>
<DETECTION>
            <QID>370613</QID>
            <TYPE>Confirmed</TYPE>
            <SEVERITY>5</SEVERITY>
            <SSL>0</SSL>
            <RESULTS><![CDATA[C:\Program
Files\Google\Chrome\Application\33.0.1750.149\chrome.dll file version is
33.0.1750.149
%ProgramFiles%\Google\Chrome\Application\33.0.1750.149\chrome.dll file
version is 33.0.1750.149]]></RESULTS>
            <STATUS>Active</STATUS>
            <FIRST_FOUND_DATETIME>2017-11-
12T20:11:32Z</FIRST_FOUND_DATETIME>
            <LAST_FOUND_DATETIME>2020-04-
27T23:04:10Z</LAST_FOUND_DATETIME>
            <TIMES_FOUND>162</TIMES_FOUND>
            <LAST_TEST_DATETIME>2020-04-27T23:04:10Z</LAST_TEST_DATETIME>
            <LAST_UPDATE_DATETIME>2020-04-
27T23:05:41Z</LAST_UPDATE_DATETIME>
            <LAST_FIXED_DATETIME>2019-10-
30T22:28:59Z</LAST_FIXED_DATETIME>
            <IS_IGNORED>0</IS_IGNORED>
            <IS_DISABLED>0</IS_DISABLED>
            <LAST_PROCESSED_DATETIME>2020-04-
27T23:05:41Z</LAST_PROCESSED_DATETIME>
          </DETECTION>
</DETECTION_LIST>
      </HOST>
```

```
      </HOST_LIST>
    </RESPONSE>
</HOST_LIST_VM_DETECTION_OUTPUT>
```