Qualys Cloud Platform (VM, PC) v8.x

API Release Notes

Version 8.14

June 1, 2018

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

URL to the Qualys API Server

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

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

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

Vault Support API - Cyber-Ark changed to CyberArk

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

We have changed Cyber-Ark to CyberArk for improved integration of CyberArk vaults. The change affects vault-type input parameter during vault creation (CyberArk AIM and CyberArk PIM Suite). The response also reflects the change.

Input Parameters

Updated input parameter is described below.

Parameter	Description
action = list or action= create	(Required)
vault_type={value}	Required only when action=create and login_type=vault. The updated options are: CyberArk AIM CyberArk PIM Suite

Example: List Vault

API request:

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

XML output:

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

"https://qualysapi.qualys.com/api/2.0/fo/vault/vault_output.dtd">
<AUTH_VAULT_LIST_OUTPUT>
    <RESPONSE>
        <DATETIME>2017-09-12T13:55:57Z</DATETIME>
        <STATUS>Success</STATUS>
        <COUNT>13</COUNT>
        <AUTH_VAULTS>
          ...
            <AUTH_VAULT>
```

```
                    <TITLE>
                        <![CDATA[My CyberArk AIM vault]]>
                    </TITLE>
                    <VAULT_TYPE>
                        <![CDATA[CyberArk AIM]]>
                    </VAULT_TYPE>
                    <LAST_MODIFIED>
                        <DATETIME>2014-02-13T12:05:21Z</DATETIME>
                        <BY>user_john</BY>
                    </LAST_MODIFIED>
                    <ID>1421</ID>
                </AUTH_VAULT>
                <AUTH_VAULT>
                    <TITLE>
                        <![CDATA[My CyberArk PIM Suite vault1]]>
                    </TITLE>
                    <VAULT_TYPE>
                        <![CDATA[CyberArk PIM Suite]]>
                    </VAULT_TYPE>
                    <LAST_MODIFIED>
                        <DATETIME>2014-02-19T06:43:44Z</DATETIME>
                        <BY>user_john</BY>
                    </LAST_MODIFIED>
                    <ID>1441</ID>
                </AUTH_VAULT>
              ...
            </AUTH_VAULTS>
        </RESPONSE>
    </AUTH_VAULT_LIST_OUTPUT>
```

Example: Create CyberArk Vault

API request:

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml"
"action=create&title=CyberArk_AIM_Vault&type=CyberArk%20AIM&appid=Qualys_
Scanner_1522015219&safe=my_safe&url=http://host.domain/AIMWebService/v1.1
/AIM.asmx"
"https://qualysapi.qualys.com/api/2.0/fo/vault/index.php/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
 <RESPONSE>
   <DATETIME>2018-04-05T05:43:32Z</DATETIME>
   <TEXT>Success</TEXT>
```

```
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>237135</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Example: Create Windows Authentication Record using Vault (CyberArk)

API request:

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml"
"action=create&title=My%20Windows%20
Record&username=Qualys&ips=10.10.10.70&login_type=vault&vault_id=236320&v
ault_type=CyberArk%20AIM&file=vaultfile&folder=abc\folder"
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE BATCH_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
 <RESPONSE>
   <DATETIME>2018-04-05T10:25:26Z</DATETIME>
   <BATCH_LIST>
     <BATCH>
       <TEXT>Successfully Created</TEXT>
       <ID_SET>
         <ID>218385</ID>
       </ID_SET>
     </BATCH>
   </BATCH_LIST>
 </RESPONSE>
</BATCH_RETURN>
```

Support for Client Id and Name in Multiple APIs

APIs affected	/api/2.0/fo/scan/?action=list /api/2.0/fo/scan/?action=launch /api/2.0/fo/scan/compliance/?action=list /api/2.0/fo/scan/compliance/?action=launch /api/2.0/fo/schedule/scan/?action=list /api/2.0/fo/schedule/scan/?action=create /api/2.0/fo/schedule/scan/?action=update /api/2.0/fo/report/?action=list /api/2.0/fo/scan/?action=fetch
New or Updated API	Updated
DTD or XSD changes	Yes

We now support for client element (id and name) for Consultant type subscriptions in Scan API, Scheduled Scan API, Compliance Scan API, and Report API.

Input Parameters

New input parameters are added for Consultant type subscriptions as described below.

Parameter	Description
client_id= {value}	(Optional) Id assigned to the client.
client_name={value}	(Optional) Name of the client. Note: The client_id and client_name parameters are mutually exclusive and cannot be specified together in the same request.

Example: List Scan

API request:

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml"
"https://qualysapi.qualys.com/api/2.0/fo/scan/
?action=list&client_name=ABC Company"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SCAN_LIST_OUTPUT SYSTEM
"http://qualysapi.qualys.com/api/2.0/fo/scan/scan_list_output.dtd">
<SCAN_LIST_OUTPUT>
    <RESPONSE>
        <DATETIME>2018-04-06T11:51:38Z</DATETIME>
        <SCAN_LIST>
```

```
            <SCAN>
                <REF>scan/1523015280.40054</REF>
                <TYPE>On-Demand</TYPE>
                <TITLE>
                    <![CDATA[Relaunched scan 20180406]]>
                </TITLE>
                <CLIENT>
                   <ID>1008</ID>
                     <NAME><![CDATA[ABC Company]]></NAME>
                </CLIENT>
                <USER_LOGIN>user_john</USER_LOGIN>
                <LAUNCH_DATETIME>2018-04-06T11:48:00Z</LAUNCH_DATETIME>
                <DURATION>00:06:28</DURATION>
               <PROCESSING_PRIORITY>0 - No Priority</PROCESSING_PRIORITY>
                <PROCESSED>1</PROCESSED>
                <STATUS>
                    <STATE>Finished</STATE>
                </STATUS>
                <TARGET>
                    <![CDATA[10.10.10.1-10.10.10.10]]>
                </TARGET>
            </SCAN>
        </SCAN_LIST>
    </RESPONSE>
</SCAN_LIST_OUTPUT>
```

Example: List Scheduled Scan

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl"
"https://qualysapi.qualys.com/api/2.0/fo/schedule/scan/?action=list&clien
t_name=ABC Company"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SCHEDULE_SCAN_LIST_OUTPUT SYSTEM
"http://qualysapi.qualys.com/api/2.0/fo/schedule/scan/schedule_scan_list_
output.dtd">
<SCHEDULE_SCAN_LIST_OUTPUT>
    <RESPONSE>
        <DATETIME>2018-04-06T12:03:28Z</DATETIME>
        <SCHEDULE_SCAN_LIST>
            <SCAN>
                <ID>22340</ID>
                <ACTIVE>1</ACTIVE>
                <TITLE>
                    <![CDATA[Scheduled Scan]]>
```

```
                </TITLE>
                <CLIENT>
                    <ID>1008</ID>
                      <NAME><![CDATA[ABC Company]]></NAME>
                </CLIENT>
                <USER_LOGIN>user_john</USER_LOGIN>
                <TARGET>
                    <![CDATA[10.10.10.1]]>
                </TARGET>
                ...
                </SCHEDULE>
            </SCAN>
        </SCHEDULE_SCAN_LIST>
    </RESPONSE>
</SCHEDULE_SCAN_LIST_OUTPUT>
```

Example: List Compliance Scan

API request:

```
curl -u "USERNAME:PASSWD" -H "X-Requested-With: curl" -d
"action=list&client_name=ABC Company"
"https://qualysapi.qualys.com/api/2.0/fo/scan/compliance/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SCAN_LIST_OUTPUT SYSTEM
"http://qualysapi.qualys.com/api/2.0/fo/scan/scan_list_output.dtd">
<SCAN_LIST_OUTPUT>
    <RESPONSE>
        <DATETIME>2018-04-06T12:01:08Z</DATETIME>
        <SCAN_LIST>
            <SCAN>
                <ID>40055</ID>
                <REF>compliance/1523015788.40055</REF>
                <TYPE>On-Demand</TYPE>
                <TITLE>
                    <![CDATA[Compliance Scan]]>
                </TITLE>
                 <CLIENT>
                   <ID>1008</ID>
                     <NAME><![CDATA[ABC Company]]></NAME>
                </CLIENT>
                <USER_LOGIN>user_john</USER_LOGIN>
                <LAUNCH_DATETIME>2018-04-06T11:56:28Z</LAUNCH_DATETIME>
                <DURATION>00:00:05</DURATION>
                <PROCESSED>1</PROCESSED>
                <STATUS>
```

```
                    <STATE>Finished</STATE>
                </STATUS>
                <TARGET>
                    <![CDATA[10.10.10.1]]>
                </TARGET>
            </SCAN>
        </SCAN_LIST>
    </RESPONSE>
</SCAN_LIST_OUTPUT>
```

Example: List Scan Report

API request:

```
curl -u "USERNAME:PASSWD" -H "X-Requested-With: curl" -d
"action=list&client_name=ABC Company"
"https://qualysapi.qualys.com/api/2.0/fo/report/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE REPORT_LIST_OUTPUT SYSTEM
"http://qualysapi.qualys.com/api/2.0/fo/report/report_list_output.dtd">
<REPORT_LIST_OUTPUT>
    <RESPONSE>
        <DATETIME>2018-04-06T12:11:19Z</DATETIME>
        <REPORT_LIST>
            <REPORT>
                <ID>34130</ID>
                <TITLE>
                    <![CDATA[Scan Report for Client]]>
                </TITLE>
                 <CLIENT>
                   <ID>1008</ID>
                     <NAME><![CDATA[ABC Company]]></NAME>
                </CLIENT>
                <TYPE>Consultant</TYPE>
                <USER_LOGIN>user_john</USER_LOGIN>
                <LAUNCH_DATETIME>2018-04-06T12:16:37Z</LAUNCH_DATETIME>
                ...
            </REPORT>
        </REPORT_LIST>
    </RESPONSE>
</REPORT_LIST_OUTPUT>
```

Example: Fetch Scan Results

Let us fetch the scan results in CSV and JSON format to view the client details.

API request (CSV):

```
curl -u "USERNAME:PASSWD" -H "X-Requested-With: curl" -d
"action=fetch&scan_ref=scan/1524812710.88997&output_format=csv_extended"
"https://qualysapi.qualys.com/api/2.0/fo/scan/"
```

CSV Report output:

```
"Scan Results","05/21/2018 11:27:19"
"Qualys, Inc.","919 E Hillsdale Blvd","Floor 4","Foster
City","California","United States of America","94404"
"Patrick Slimmer","qualys_ps","Manager"

"Launch Date","Client", "Active Hosts","Total
Hosts","Type","Status","Reference","Scanner Appliance","Duration","Scan
Title","Asset Groups","IPs","Excluded IPs","Option Profile","Network"
"04/27/2018 07:05:09","ABC Company","4","5","On
Demand","Finished","scan/1524812710.88997","10.113.244.56 (Scanner
10.0.14-1, Vulnerability Signatures 2.4.319-1)","00:18:26","My-Client-
Scan",,"10.113.197.129-10.113.197.133",,"Initial Options","Global Default
Network"
```

API request (JSON):

```
curl -u "USERNAME:PASSWD" -H "X-Requested-With: curl" -d
"action=fetch&scan_ref=scan/1524812710.88997&output_format=json_extended"
"https://qualysapi.qualys.com/api/2.0/fo/scan/"
```

JSON Report output:

```
[{"scan_report_template_title":"Scan Results","result_date":"05\/21\/2018
11:31:38","company":"Qualys, Inc.","add1":"919 E Hillsdale
Blvd","add2":"Floor 4","city":"Foster
City","state":"California","country":"United States of
America","zip":"94404","name":"Patrick
Slimmer","username":"qualys_ps","role":"Manager"},
{"launch_date":"04\/27\/2018 07:05:09","Client":"ABC
Company","active_hosts":"4","total_hosts":"5","type":"On
Demand","status":"Finished","reference":"scan\/1524812710.88997","scanner
_appliance":"10.113.244.56 (Scanner 10.0.14-1, Vulnerability Signatures
2.4.319-1)","duration":"00:18:26","scan_title":"My-Client-
Scan","asset_groups":null,"ips":"10.113.197.129-
10.113.197.133","excluded_ips":"","option_profile":"Initial
Options","network":"Global Default Network"},
```

DTD Update

We updated the DTD to include the client name and client id element (in bold).

Scan List DTD (scan_list_output.dtd)

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

```
<!-- QUALYS SCAN_LIST_OUTPUT DTD -->
<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
POST_DATA?)>
...
<!ELEMENT TYPE (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT CLIENT (ID,NAME)>
...
<!ELEMENT OPTION_PROFILE (TITLE, DEFAULT_FLAG?)>
<!ELEMENT DEFAULT_FLAG (#PCDATA)>
<!-- EOF -->
```

Scheduled Scan List DTD (schedule_scan_list_output.dtd)

DTD: <base_url>/api/2.0/fo/schedule/scan/schedule_scan_list_output.dtd

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

<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
POST_DATA?)>
....
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT CLIENT (ID,NAME)>
...
<!ELEMENT DISTRIBUTION_GROUP (ID, TITLE)>
<!-- EOF -->
```

Report List DTD (report_list_output.dtd)

DTD: <base_url>/api/2.0/fo/report/report_list_output.dtd

```
<!-- QUALYS SCHEDULE_SCAN_LIST_OUTPUT DTD -->
<!ELEMENT SCHEDULE_SCAN_LIST_OUTPUT (REQUEST?,RESPONSE)>
<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
POST_DATA?)>
....
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT CLIENT (ID,NAME)>
...
<!ELEMENT PERCENT (#PCDATA)>
<!ELEMENT EXPIRATION_DATETIME (#PCDATA)>
<!-- EOF -->
```

New Scan Summary API for Hosts Not Scanned

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

This new Summary API lets you identify hosts that were not scanned and why.

How it works - First we'll find all the scans launched since the date (or within the date range) that you specify. Then we'll identify hosts that were included in the scan target but not scanned for some reason. For each host you'll see the category/reason it was not scanned and the host's tracking method.

Categories for hosts not scanned:

Excluded - The hosts were excluded. Hosts may be excluded on a per scan basis (by the user launching or scheduling the scan) or globally for all scans. Managers and Unit Managers have privileges to edit the global excluded hosts list for the subscription.

Cancelled - Hosts were not scanned because the scan was cancelled. Scans may be cancelled by a user, by an administrator or automatically by the service as specified in scheduled scan settings.

Dead - The hosts were not "alive" at the time of the scan, meaning that they did not respond to probes sent by the scanning engine, and the option to Scan Dead Hosts was not enabled.

Unresolved - Hosts were scanned but they could not be reported because the NetBIOS or DNS hostname, whichever tracking method is specified for each host, could not be resolved.

Duplicate - The hosts were duplicated within a single segment/slice of the scan job. For example, two different hostnames resolving to the same IP with tracking by IP.

Not Vulnerable - Hosts were found to be not vulnerable during host discovery without having to run a full scan. This could happen for example if the list of QIDs to be scanned are limited to certain ports and those ports are found to be closed.

Aborted - The scan was abruptly discontinued. This is a rare occurrence that may be caused for various reasons. Contact Support for assistance.

Blocked - Hosts were blocked from scanning for some reason.

Input Parameters

Use input parameters to filter hosts in the output as described below.

Parameter	Description
action=list	(Required)
scan_date_since={value}	(Required) Include scans started since a certain date. Specify the date in YYYY-MM-DD format. The date must be less than or equal to today's date.
scan_date_to={value}	(Optional) Include scans started up to a certain date. Specify the date in YYYY-MM-DD format. The date must be more than or equal to scan_date_since, and less than or equal to today's date.
output_format={value}	(Optional) The output format: XML (the default), CSV or JSON.
tracking_method={value}	(Optional) By default hosts with any tracking method will be returned in the output. Use this option to only include hosts with a certain tracking method. Valid values are: IP, DNS, NETBIOS, EC2.
include_dead={0|**1**}	(Optional) Set to 0 if you do not want to include dead hosts in the output. Dead hosts are included by default.
include_excluded={**0**|1}	(Optional) Set to 1 to include hosts that were excluded from a scan in the output. Excluded hosts are not included by default.
include_unresolved={**0**|1}	(Optional) Set to 1 to include unresolved hosts in the output. Unresolved hosts are not included by default.
include_cancelled={**0**|1}	(Optional) Set to 1 to include cancelled hosts in the output. Cancelled hosts are not included by default.
include_notvuln={**0**|1}	(Optional) Set to 1 to include hosts that are not vulnerable in the output. Not vulnerable hosts are not included by default.
include_blocked={**0**|1}	(Optional) Set to 1 to include blocked hosts in the output. Blocked hosts are not included by default.
include_duplicate={**0**|1}	(Optional) Set to 1 to include duplicate hosts in the output. Duplicate hosts are not included by default.
include_aborted={**0**|1}	(Optional) Set to 1 to include aborted hosts in the output. Aborted hosts are not included by default.

Example - XML Output

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl"
"https://qualysapi.qualys.com/api/2.0/fo/scan/summary/?action=list
&scan_date_since=2018-04-27&include_excluded=1&include_unresolved=1
&include_cancelled=1&include_notvuln=1&include_duplicate=1"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SCAN_SUMMARY_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/scan/summary/scan_summary_output
.dtd">
<SCAN_SUMMARY_OUTPUT>
 <RESPONSE>
   <DATETIME>2018-05-02T10:45:40Z</DATETIME>
   <SCAN_SUMMARY_LIST>
     <SCAN_SUMMARY>
       <SCAN_REF>scan/1525251885.92469</SCAN_REF>
       <SCAN_DATE>2018-05-02T09:04:34Z</SCAN_DATE>
       <HOST_SUMMARY category="notvuln" tracking="IP">10.10.10.10-
10.10.10.15,10.10.10.17</HOST_SUMMARY>
       <HOST_SUMMARY category="notvuln" tracking="DNS">gfi-31-
1.caac125.qualys.com,gfi-31-2.caac125.qualys.com</HOST_SUMMARY>
       <HOST_SUMMARY category="notvuln" tracking="NETBIOS">gfi-31-3,gfi-
31-4</HOST_SUMMARY>
       <HOST_SUMMARY category="cancelled"
tracking="IP">10.10.10.20,10.10.10.22</HOST_SUMMARY>
       <HOST_SUMMARY category="cancelled" tracking="DNS">gfi-31-
5.caac125.qualys.com,gfi-31-6.caac125.qualys.com</HOST_SUMMARY>
      <HOST_SUMMARY category="cancelled" tracking="NETBIOS">gfi-31-7,gfi-
31-8</HOST_SUMMARY>
       <HOST_SUMMARY category="dead"
tracking="IP">10.10.10.25</HOST_SUMMARY>
       <HOST_SUMMARY category="dead" tracking="DNS">gfi-31-
9.caac125.qualys.com</HOST_SUMMARY>
       <HOST_SUMMARY category="dead" tracking="NETBIOS">gfi-31-10,gfi-31-
11</HOST_SUMMARY>
       <HOST_SUMMARY category="excluded"
tracking="IP">10.10.10.26</HOST_SUMMARY>
       <HOST_SUMMARY category="unresolved" tracking="DNS">gfi-31-
12.caac125.qualys.com</HOST_SUMMARY>
       <HOST_SUMMARY category="unresolved" tracking="NETBIOS">gfi-31-
13</HOST_SUMMARY>
       <HOST_SUMMARY category="duplicate"
tracking="IP">10.10.10.27</HOST_SUMMARY>
       <HOST_SUMMARY category="duplicate" tracking="DNS">gfi-31-
14.caac125.qualys.com</HOST_SUMMARY>
       <HOST_SUMMARY category="duplicate" tracking="NETBIOS">gfi-31-
15</HOST_SUMMARY>
     </SCAN_SUMMARY>
   </SCAN_SUMMARY_LIST>
 </RESPONSE>
</SCAN_SUMMARY_OUTPUT>
```

Example - CSV Output

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl"
"https://qualysapi.qualys.com/api/2.0/fo/scan/summary/?action=list
&scan_date_since=2018-04-27&include_excluded=1&include_unresolved=1
&include_cancelled=1&include_notvuln=1&include_duplicate=1&output_format=
CSV"
```

CSV output:

```
"SCAN_REF","SCAN_DATE","CATEGORY","TRACKING","HOSTS"
"scan/1525251885.92469","2018-05-
02T09:04:34Z","notvuln","IP","10.10.10.10-10.10.10.15,10.10.10.17"
"scan/1525251885.92469","2018-05-02T09:04:34Z","notvuln","DNS","gfi-31-
1.caac125.qualys.com,gfi-31-2.caac125.qualys.com"
"scan/1525251885.92469","2018-05-02T09:04:34Z","notvuln","NETBIOS","gfi-
31-3,gfi-31-4"
"scan/1525251885.92469","2018-05-
02T09:04:34Z","cancelled","IP","10.10.10.20,10.10.10.22"
"scan/1525251885.92469","2018-05-02T09:04:34Z","cancelled","DNS","gfi-31-
5.caac125.qualys.com,gfi-31-6.caac125.qualys.com"
"scan/1525251885.92469","2018-05-
02T09:04:34Z","cancelled","NETBIOS","gfi-31-7,gfi-31-8"
"scan/1525251885.92469","2018-05-02T09:04:34Z","dead","IP","10.10.10.25"
"scan/1525251885.92469","2018-05-02T09:04:34Z","dead","DNS","gfi-31-
9.caac125.qualys.com"
"scan/1525251885.92469","2018-05-02T09:04:34Z","dead","NETBIOS","gfi-31-
10,gfi-31-11"
"scan/1525251885.92469","2018-05-
02T09:04:34Z","excluded","IP","10.10.10.26"
"scan/1525251885.92469","2018-05-02T09:04:34Z","unresolved","DNS","gfi-
31-12.caac125.qualys.com"
"scan/1525251885.92469","2018-05-
02T09:04:34Z","unresolved","NETBIOS","gfi-31-13"
"scan/1525251885.92469","2018-05-
02T09:04:34Z","duplicate","IP","10.10.10.27"
"scan/1525251885.92469","2018-05-02T09:04:34Z","duplicate","DNS","gfi-31-
14.caac125.qualys.com"
"scan/1525251885.92469","2018-05-
02T09:04:34Z","duplicate","NETBIOS","gfi-31-15"
```

Example - JSON Output

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl"
"https://qualysapi.qualys.com/api/2.0/fo/scan/summary/?action=list
&scan_date_since=2018-04-27&include_excluded=1&include_unresolved=1
&include_cancelled=1&include_notvuln=1&include_duplicate=1&output_format=
JSON"
```

JSON output:

```
[
{
    "scanRef": "scan\/1525251885.92469",
    "scanDate": "2018-05-02T09:04:34Z",
    "hostSummary": [
        {
            "category": "notvuln",
            "tracking": "IP",
            "hosts": "10.10.10.10-10.10.10.15,10.10.10.17"
        },
        {
            "category": "notvuln",
            "tracking": "DNS",
            "hosts": "gfi-31-1.caac125.qualys.com,gfi-31-
2.caac125.qualys.com"
        },
        {
            "category": "notvuln",
            "tracking": "NETBIOS",
            "hosts": "gfi-31-3,gfi-31-4"
        },
        {
            "category": "cancelled",
            "tracking": "IP",
            "hosts": "10.10.10.20,10.10.10.22"
        },
        {
            "category": "cancelled",
            "tracking": "DNS",
            "hosts": "gfi-31-5.caac125.qualys.com,gfi-31-
6.caac125.qualys.com"
        },
        {
            "category": "cancelled",
            "tracking": "NETBIOS",
            "hosts": "gfi-31-7,gfi-31-8"
        },
        {
```

```
            "category": "dead",
            "tracking": "IP",
            "hosts": "10.10.10.25"
        },
        {
            "category": "dead",
            "tracking": "DNS",
            "hosts": "gfi-31-9.caac125.qualys.com"
        },
        {
            "category": "dead",
            "tracking": "NETBIOS",
            "hosts": "gfi-31-10,gfi-31-11"
        },
        {
            "category": "excluded",
            "tracking": "IP",
            "hosts": "10.10.10.26"
        },
        {
            "category": "unresolved",
            "tracking": "DNS",
            "hosts": "gfi-31-12.caac125.qualys.com"
        },
        {
            "category": "unresolved",
            "tracking": "NETBIOS",
            "hosts": "gfi-31-13"
        },
        {
            "category": "duplicate",
            "tracking": "IP",
            "hosts": "10.10.10.27"
        },
        {
            "category": "duplicate",
            "tracking": "DNS",
            "hosts": "gfi-31-14.caac125.qualys.com"
        },
        {
            "category": "duplicate",
            "tracking": "NETBIOS",
            "hosts": "gfi-31-15"
        },
    ]
}
]
```

New DTD

This new DTD (scan_summary_output.dtd) is used for the Scan Summary API.

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

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

<!ELEMENT RESPONSE (DATETIME, SCAN_SUMMARY_LIST?)>
<!ELEMENT SCAN_SUMMARY_LIST (SCAN_SUMMARY*)>
<!ELEMENT SCAN_SUMMARY (SCAN_REF?, SCAN_DATE?, HOST_SUMMARY*)>
<!ELEMENT SCAN_REF (#PCDATA)>
<!ELEMENT SCAN_DATE (#PCDATA)>
<!ELEMENT HOST_SUMMARY (#PCDATA)>

<!ATTLIST HOST_SUMMARY category CDATA #IMPLIED>
<!ATTLIST HOST_SUMMARY tracking CDATA #IMPLIED>
<!-- EOF -->
```

New Support for Wallix AdminBastion (WAB) Vaults

API affected	/api/2.0/fo/vault/
New or Updated API	Updated
DTD or XSD changes	No
API affected	/api/2.0/fo/auth/windows/
New or Updated API	Updated
DTD or XSD changes	Yes
API affected	/api/2.0/fo/auth/unix/
New or Updated API	Updated
DTD or XSD changes	Yes

This new vault type can be used to retrieve authentication credentials from a Wallix AdminBastion (WAB) vault. We updated the authentication vault API (create, update, list, view) and the authentication record API (create, update, list) to support the new vault type. We updated the DTDs for listing Windows and Unix records.

Authentication Vault API

You can now create, update, list and view Wallix AdminBastion (WAB) vaults.

Create WAB Authentication Vault

Use the parameter "action=create" to create a new vault in your account.

Parameter	Description
action=create	(Required)
title={value}	(Required) The vault title.
type={value}	(Required) Specify type=Wallix AdminBastion (WAB)
comments={value}	(Optional) User defined comments.
url={value}	(Required) The HTTP or HTTPS URL to access the WAB web services API.
ssl_verify={0|**1**}	(Optional) When set to 1 (the default), our service will verify the SSL certificate of the web server to make sure the certificate is valid and trusted. When set to 0, our service will not verify the certificate of the web server.
username={value}	(Required) The user account that can call the WAB web services API.

Parameter	Description
password={value}	(Optional) The password for the user account that can call the WAB web services API. For a create request, you must specify password or appkey. Both parameters cannot be specified in the same request.
appkey={value}	(Optional) Your WAB REST API key (alpha-numeric value) for connecting to the WAB web services API. - Do not include leading or trailing periods or spaces. - These characters are not allowed: \\ / : * ? " < > | - UTF-8 multibyte characters are not allowed. For a create request, you must specify password or appkey. Both parameters cannot be specified in the same request.

Examples

Create vault with username & password

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=create&type=Wallix AdminBastion (WAB)&title=My WAB
Vault&url=https://wab.example.com/api&ssl_verify=1&username=user&password
=abc123&comments=creating wab vault from api"
"https://qualysapi.qualys.com/api/2.0/fo/vault/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-05-08T18:02:13Z</DATETIME>
    <TEXT>Success</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>246031</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Create vault with username & appkey

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=create&type=Wallix AdminBastion (WAB)&title=My WAB Vault2&url=
https://wab.example.com/api&ssl_verify=0&username=user&appkey=
AyZJNAyDIaPpzCBsT6ElNRO8o9Sfa6fsbZ2LHH8V6G&comments=creating wab vault
from api using appkey" "https://qualysapi.qualys.com/api/2.0/fo/vault/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-05-08T18:26:14Z</DATETIME>
    <TEXT>Success</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>246040</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Update WAB Authentication Vault

Use the parameter “action=update” to update a vault in your account. You must include the ID for the vault you’re updating.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&id=246031&title=New Vault
Title&ssl_verify=0&username=user2&password=xyz456&comments=Update vault"
"https://qualysapi.qualys.com/api/2.0/fo/vault/">update_wab_vault.xml
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-05-08T18:11:58Z</DATETIME>
    <TEXT>Success</TEXT>
    <ITEM_LIST>
```

```
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>246031</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

List Authentication Vaults

Use the parameter "action=list" to list the vault defined in your account. To view a specific vault, specify the ID or title of the vault.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=list&title=My WAB Vault"
"https://qualysapi.qualys.com/api/2.0/fo/vault/">vault_list2.xml
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_VAULT_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/vault/vault_output.dtd">
<!-- This report was generated with an evaluation version of Qualys //-->
<AUTH_VAULT_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-05-08T18:08:44Z</DATETIME>
    <STATUS>Success</STATUS>
    <COUNT>1</COUNT>
    <AUTH_VAULTS>
      <AUTH_VAULT>
        <TITLE><![CDATA[My WAB Vault]]></TITLE>
        <VAULT_TYPE><![CDATA[Wallix AdminBastion (WAB)]]></VAULT_TYPE>
        <LAST_MODIFIED>
          <DATETIME>2018-05-08T18:02:12Z</DATETIME>
          <BY>qualys_ps</BY>
        </LAST_MODIFIED>
        <ID>246031</ID>
      </AUTH_VAULT>
    </AUTH_VAULTS>
  </RESPONSE>
</AUTH_VAULT_LIST_OUTPUT>
```

View WAB Authentication Vault

Use the parameter “action=view” and specify the ID of the vault to view its details.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=view&id=246031"
"https://qualysapi.qualys.com/api/2.0/fo/vault/">vault2.xml
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE VAULT_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/vault/vault_view.dtd">
<!-- This report was generated with an evaluation version of Qualys //-->
<VAULT_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-05-08T18:05:20Z</DATETIME>
    <VAULT_QUEST>
      <TITLE><![CDATA[My WAB Vault]]></TITLE>
      <COMMENTS><![CDATA[creating wab vault from api]]></COMMENTS>
      <VAULT_TYPE><![CDATA[Wallix AdminBastion (WAB)]]></VAULT_TYPE>
      <CREATED_ON>2018-05-08T18:02:12Z</CREATED_ON>
      <OWNER>qualys_ps</OWNER>
      <LAST_MODIFIED>
        <DATETIME>2018-05-08T18:02:12Z</DATETIME>
        <BY>qualys_ps</BY>
      </LAST_MODIFIED>
      <APPKEY><![CDATA[]]></APPKEY>
      <USERNAME><![CDATA[user]]></USERNAME>
      <URL><![CDATA[https://wab.example.com/api]]></URL>
      <SSL_VERIFY><![CDATA[1]]></SSL_VERIFY>
      <ID>246031</ID>
    </VAULT_QUEST>
  </RESPONSE>
</VAULT_OUTPUT>
```

Authentication Record API

Create, update, list authentication records with Wallix AdminBastion (WAB) vaults. You can use WAB vaults with Windows and Unix records.

Create Authentication Record

Use these input parameters when creating/updating an authentication record and getting credentials from your Wallix AdminBastion (WAB) vault.

Parameter	Description
action=create|update	(Required)
login_type={value}	(Required to create/update vault information) Specify login_type=vault to add vault information. By default, the parameter is set to basic.
vault_id={value}	(Required when action=create and login_type=vault) A vault ID.
vault_type={value}	(Required when action=create and login_type=vault) Specify vault_type=Wallix AdminBastion (WAB)
authorization_name={value}	(Required when vault_type=Wallix AdminBastion (WAB)) The name of the authorization that enables secret retrieval from a group of targets.
target_name={value}	(Required when vault_type=Wallix AdminBastion (WAB)) Specify the name of the target device using one of these formats: user@global_WABdomain user@local_WABdomain@device where *user* is the user with access to the target, *global_WABdomain* is a domain name in a domain controller, *local_WABdomain* is a local domain, *device* is the device you want to scan You can use one or more variables to match several targets that use the same naming convention. See below.

Using Variables in the Target Name

You can use one or more variables when defining the target name in order to match several targets that use the same naming convention.

${ip} // The IP address of the target, i.e. 10.20.30.40.

${ip_dash} // The IP address of the target with dashes instead of dots, i.e. 10-20-30-40.

${dnshost} // The DNS host name of the target, i.e. host.domain.

${host} // The host name of the target, i.e. host before .domain.

${nbhost} // (Windows only) The NetBIOS name of the target in upper-case, i.e. HOST_ABC.

Sample Target Names

Let's say you have these 6 devices in WAB:

CentOS6
10.50.60.70
10.50.60.88
10.30.10.12
10-20-32-201
10-20-31-112_win81-x86.prod.qualys.com

You'll need to create 4 records with the following target names where the user is "qualys_scan" and the local_WABdomain is "local":

Record 1: qualys_scan@local@CentOS6

Record 2: qualys_scan@local@${ip} (matches 10.50.60.70, 10.50.60.88 and 10.30.10.12)

Record 3: qualys_scan@local@10-20-32-201 -or- qualys_scan@local@${ip_dash}

Record 4: qualys_scan@local@10-20-31-112_win81-x86.prod.qualys.com -or- qualys_scan@local@${ip_dash}_${dnshost}

Examples

Create Windows record with WAB vault

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=create&title=WINDOWS_AUTH_WAB&username=administrator&ips=10.20.31
.112&login_type=vault&vault_id=246044&vault_type=Wallix AdminBastion
(WAB)&authorization_name=my_authorization&target_name=administrator@TEMP-
WIN81-X86@Windows-8"
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE BATCH_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
  <RESPONSE>
    <DATETIME>2018-05-08T21:17:37Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
```

```
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>241064</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Update Windows record with WAB vault

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&ids=241064&title=WINDOWS_AUTH_WAB2&ips=10.20.31.112,10.10.
10.10&login_type=vault&vault_id=246044&vault_type=Wallix AdminBastion
(WAB)&authorization_name=my_authorization2&target_name=administrator@TEMP
-WIN81-X86@Windows-8"
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/">update_windows.xm
l
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE BATCH_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
  <RESPONSE>
    <DATETIME>2018-05-08T22:01:02Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Updated</TEXT>
        <ID_SET>
          <ID>241064</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

List Windows authentication records

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=list&ids=241064"
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/">windows_wab.xml
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_WINDOWS_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/auth_windows_list_o
utput.dtd">
<AUTH_WINDOWS_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-05-08T21:32:32Z</DATETIME>
    <AUTH_WINDOWS_LIST>
      <AUTH_WINDOWS>
        <ID>241064</ID>
        <TITLE><![CDATA[WINDOWS_AUTH_WAB]]></TITLE>
        <USERNAME><![CDATA[administrator]]></USERNAME>
        <NTLM_V2>1</NTLM_V2>
        <IP_SET>
          <IP>10.20.31.112</IP>
        </IP_SET>
        <LOGIN_TYPE><![CDATA[vault]]></LOGIN_TYPE>
        <DIGITAL_VAULT>
          <DIGITAL_VAULT_ID><![CDATA[246044]]></DIGITAL_VAULT_ID>
          <DIGITAL_VAULT_TYPE><![CDATA[Wallix AdminBastion
(WAB)]]></DIGITAL_VAULT_TYPE>
          <DIGITAL_VAULT_TITLE><![CDATA[My WAB
Vault]]></DIGITAL_VAULT_TITLE>

<VAULT_AUTHORIZATION_NAME><![CDATA[my_authorization]]></VAULT_AUTHORIZATI
ON_NAME>
          <VAULT_TARGET_NAME><![CDATA[administrator@TEMP-WIN81-
X86@Windows-8]]></VAULT_TARGET_NAME>
        </DIGITAL_VAULT>
        <NETWORK_ID>0</NETWORK_ID>
        <CREATED>
          <DATETIME>2018-05-08T21:17:36Z</DATETIME>
          <BY>qualys_ps</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2018-05-08T21:17:36Z</DATETIME>
        </LAST_MODIFIED>
      </AUTH_WINDOWS>
    </AUTH_WINDOWS_LIST>
  </RESPONSE>
</AUTH_WINDOWS_LIST_OUTPUT>
```

DTD Updates

We added VAULT_AUTHORIZATION_NAME and VAULT_TARGET_NAME to both the Windows and Unix Authentication Record List Output DTDs.

Windows Authentication Records List Output DTD

```
<!-- QUALYS AUTH_WINDOWS_LIST_OUTPUT DTD -->
...
<!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_AUTHORIZATION_NAME?, VAULT_TARGET_NAME?)>
<!ELEMENT DIGITAL_VAULT_ID (#PCDATA)>
<!ELEMENT DIGITAL_VAULT_TYPE (#PCDATA)>
<!ELEMENT DIGITAL_VAULT_TITLE (#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_AUTHORIZATION_NAME (#PCDATA)>
<!ELEMENT VAULT_TARGET_NAME (#PCDATA)>
...
```

Unix Authentication Records List Output DTD

```
<!-- QUALYS AUTH_UNIX_LIST_OUTPUT DTD -->
...
<!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_EP_NAME?, VAULT_EP_TYPE?,
VAULT_EP_CONT?, VAULT_NS_TYPE?, VAULT_NS_NAME?, VAULT_ACCOUNT_NAME?,
VAULT_AUTHORIZATION_NAME?, VAULT_TARGET_NAME?)>
<!ELEMENT DIGITAL_VAULT_ID (#PCDATA)>
<!ELEMENT DIGITAL_VAULT_TYPE (#PCDATA)>
<!ELEMENT DIGITAL_VAULT_TITLE (#PCDATA)>
<!ELEMENT VAULT_USERNAME (#PCDATA)>
<!ELEMENT VAULT_FOLDER (#PCDATA)>
<!ELEMENT VAULT_FILE (#PCDATA)>
<!ELEMENT VAULT_SECRET_NAME (#PCDATA)>
<!ELEMENT VAULT_SYSTEM_NAME (#PCDATA)>
<!ELEMENT VAULT_EP_NAME (#PCDATA)>
<!ELEMENT VAULT_EP_TYPE (#PCDATA)>
```

```
<!ELEMENT VAULT_EP_CONT (#PCDATA)>
<!ELEMENT VAULT_NS_TYPE (#PCDATA)>
<!ELEMENT VAULT_NS_NAME (#PCDATA)>
<!ELEMENT VAULT_ACCOUNT_NAME (#PCDATA)>
<!ELEMENT VAULT_AUTHORIZATION_NAME (#PCDATA)>
<!ELEMENT VAULT_TARGET_NAME (#PCDATA)>
...
```

Fix to Vault View API Output

API affected	/api/2.0/fo/vault/ with action=view
New or Updated API	Neither (output change only)
DTD or XSD changes	No

We fixed the XML output of the authentication vault view API to fix a DTD validation error. When echo_request=1 is specified as part of the API call, the REQUEST section now correctly appears before the RESPONSE section in the output.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=view&id=244084&echo_request=1"
"https://qualysapi.qualys.com/api/2.0/fo/vault/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE VAULT_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/vault/vault_view.dtd">
<VAULT_OUTPUT>
  <REQUEST>
    <DATETIME>2018-05-15T17:22:56Z</DATETIME>
    <USER_LOGIN>qualys_ps</USER_LOGIN>
    <RESOURCE>https://qualysapi.qualys.com/api/2.0/fo/vault/</RESOURCE>
    <PARAM_LIST>
      <PARAM>
        <KEY>action</KEY>
        <VALUE>view</VALUE>
      </PARAM>
      <PARAM>
        <KEY>id</KEY>
        <VALUE>244084</VALUE>
      </PARAM>
      <PARAM>
        <KEY>echo_request</KEY>
        <VALUE>1</VALUE>
      </PARAM>
    </PARAM_LIST>
  </REQUEST>
  <RESPONSE>
    <DATETIME>2018-05-15T17:22:56Z</DATETIME>
    <VAULT_QUEST>
      <TITLE><![CDATA[WAB Vault]]></TITLE>
      <COMMENTS><![CDATA[creating wab vault]]></COMMENTS>
      <VAULT_TYPE><![CDATA[Wallix AdminBastion (WAB)]]></VAULT_TYPE>
```

```
      <CREATED_ON>2018-05-02T22:25:30Z</CREATED_ON>
      <OWNER>qualys_ps</OWNER>
      <LAST_MODIFIED>
        <DATETIME>2018-05-02T22:41:26Z</DATETIME>
        <BY>qualys_ps</BY>
      </LAST_MODIFIED>
      <APPKEY><![CDATA[]]></APPKEY>
      <USERNAME><![CDATA[root6update]]></USERNAME>
      <URL><![CDATA[https://wab.example.com/api6update]]></URL>
      <SSL_VERIFY><![CDATA[0]]></SSL_VERIFY>
      <ID>244084</ID>
    </VAULT_QUEST>
  </RESPONSE>
</VAULT_OUTPUT>
```

Support for EC2 Scanning using only Instance ID

APIs affected	/api/2.0/fo/scan/ /api/2.0/fo/scan/compliance/
New or Updated API	Updated
DTD or XSD changes	No

We now support launch of on demand internal ec2 scans using only ec2 instance ids. You can use tags if needed. Using tags is now optional.

Input Parameters

Updated input parameter is described below.

Parameter	Description
action=launch	An action for the request.
ec2_instance_ids={value}	(Optional) The ID of the EC2 instance on which you want to launch the VM or compliance scan. Multiple ec2 instance ids are comma separated. You can add up to maximum 10 instance Ids.

Example: Launch VM Scan for EC2 Instance

Let us launch a VM scan on EC2 instances using the parameter ec2_instance_ids.

API request:

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml"
"action=launch&scan_title=Ec2InstanceScanScan_TAGS_1525653991&&option_tit
le=Initial+Options&iscanner_id=212711&connector_name=arn&ec2_endpoint=us-
east-1&ec2_instance_ids=i-0c9768f97a2816ad6, i-0211dfd18a6dff979"
"https://qualysapi.qualys.com/api/2.0/fo/scan/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-05-07T10:04:02Z</DATETIME>
    <TEXT>New VM scan is launched</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>295771</VALUE>
```

```
      </ITEM>
      <ITEM>
        <KEY>REFERENCE</KEY>
        <VALUE>scan/1525687440.95771</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
```

Example: Launch Compliance Scan for EC2 Instance

Let us launch a compliance scan on EC2 instances using the parameter ec2_instance_ids.

API request:

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml"
"action=launch&scan_title=Compliance_scan_EC2_instancesScan_TAGS_15256561
68&option_title=Initial+PC+Options&iscanner_name=netwr_nd_EC2_1&connector
_name=VA_Conn1_US_East&ec2_endpoint=vpc-1e37cd76&ec2_instance_ids=i-
09bf38681784b500f, i-08696271b868d8355"
"https://qualysapi.qualys.com/api/2.0/fo/scan/compliance/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-05-07T12:24:46Z</DATETIME>
    <TEXT>New Compliance Scan is launched</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>295847</VALUE>
      </ITEM>
      <ITEM>
        <KEY>REFERENCE</KEY>
        <VALUE>compliance/1525695884.95847</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Example: Scan List

The sample scan list API call (GET method) below returns scans within the user account. The target element now lists the ec2_instance_id in the response.

API request:

```
curl -u "USERNAME:PASSWD" -H "X-Requested-With: curl" -d "action=list"
"https://qualysapi.qualys.com/api/2.0/fo/scan/"
"action=list&scan_ref=scan/1525687440.95771"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_VAULT_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/scan/scan_list_output.dtd">
<SCAN_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2018-05-07T10:09:16Z</DATETIME>
    <SCAN_LIST>
      <SCAN>
        <REF>scan/1525687440.95771</REF>
        <TYPE>API</TYPE>
        <TITLE><![CDATA[testScan_TAGS_1525653991]]></TITLE>
        <USER_LOGIN>user_john</USER_LOGIN>
        <LAUNCH_DATETIME>2018-05-07T10:04:00Z</LAUNCH_DATETIME>
        <DURATION>Pending</DURATION>
        <PROCESSING_PRIORITY>0 - No Priority</PROCESSING_PRIORITY>
        <PROCESSED>0</PROCESSED>
        <STATUS>
          <STATE>Queued</STATE>
          <SUB_STATE>Launch_Requested</SUB_STATE>
        </STATUS>
        <TARGET><![CDATA[i-0c9768f97a2816ad6, i-
0211dfd18a6dff979]]></TARGET>
      </SCAN>
    </SCAN_LIST>
  </RESPONSE>
</SCAN_LIST_OUTPUT>
```

Update to CertView Scan Results to include FQDN

We added FQDN to the header section of CertView scan results where we'll now list the FQDNs in the scan target, if any. Previously we listed the target FQDNs with the target IPs. You can download scan results from the UI or fetch results from the API. These changes apply to CertView Scans only.

API updates: Scan Results (fetch from API) | Scan Results (download from UI)

Scan Results (fetch from API)

API affected	/api/2.0/fo/scan/?action=fetch
New or Updated API	Neither (output change only)
DTD or XSD changes	No

Example - CSV Extended Format

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=fetch&scan_ref=scan/1526427022.06085&mode=extended&output_format=
csv_extended" "https://qualysapi.qualys.com/api/2.0/fo/scan/"
```

Sample CSV output:

We added "FQDN" in the CSV Header. This change only applies to CertView scans.

```
"Scan Results","05/21/2018 at 15:10:35 (GMT-0700)"
"Qualys, Inc.","919 E Hillsdale Blvd, Floor 4",,"Foster
City","California","United States of America","94404"
"Patrick Slimmer","qualys_ps","Manager"

"Launch Date","Active Hosts","Total Hosts","Type","Status","Reference",
"Scanner Appliance","Duration","Scan Title","Asset Groups","IPs",
"Excluded IPs","Option Profile","FQDN"
"05/15/2018 23:30:21","6","8","Scheduled","Finished",
"scan/1526427022.06085",
"10.11.51.103 (Scanner 9.7.20-1, Vulnerability Signatures 2.4.182-
2)","00:18:49","My-CertView-Scan","My-Asset-Group","10.10.25.143,
10.11.65.212-10.11.65.214","10.11.65.213-10.11.65.214","My-Option-
Profile","2k3sp2-25.test.qualys.com, 2k8sp0-25.test.qualys.com,
s2016tp5dc.test.qualys.com"
```

Example - JSON Extended Format

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=fetch&scan_ref=scan/1526427022.06085&mode=extended&output_format=
json_extended" "https://qualysapi.qualys.com/api/2.0/fo/scan/"
```

Sample JSON output:

We added "FQDN". This change only applies to CertView scans.

```
[{"scan_report_template_title":"Scan Results","result_date":"05\/21\/2018
13:04:06","company":"Qualys,Inc.","add1":"919 E Hillsdale Blvd, Floor
4","add2":null,"city":"Foster
City","state":"California","country":"United States of
America","zip":"94404","name":"Patrick
Slimmer","username":"qualys_ps","role":"Manager"},
{"launch_date":"05\/15\/2018
23:30:21","active_hosts":"6","total_hosts":"8","type":"Scheduled","status
":"Finished","reference":"scan\/1526427022.06085","scanner_appliance":"10
.11.51.103 (Scanner 9.7.20-1, Vulnerability Signatures 2.4.182-
2)","duration":"00:18:49","scan_title":"My-CertView-
Scan","asset_groups":"My-Asset-Group","ips":"10.10.25.143, 10.11.65.212-
10.11.65.214","excluded_ips":"10.11.65.213-
10.11.65.214","option_profile":"My-Option-Profile","fqdn":"2k3sp2-
25.test.qualys.com, 2k8sp0-25.test.qualys.com,
s2016tp5dc.test.qualys.com"},
```

Scan Results (download from UI)

API affected	N/A (affects end report)
New or Updated API	Neither (output change only)
DTD or XSD changes	No

Scan Results in XML format

We added <KEY value="FQDN"> to the Header section of the XML output. This change only applies to CertView scans.

```
<HEADER>
  <KEY value="USERNAME">qualys_ps</KEY>
  <KEY value="COMPANY"><![CDATA[Qualys, Inc.]]></KEY>
  <KEY value="DATE">2018-05-15T21:26:21Z</KEY>
  <KEY value="TITLE"><![CDATA[My-CertView-Scan]]></KEY>
  <KEY value="TARGET"><![CDATA[10.10.31.58, 10.11.65.212-
```

```
10.11.65.214]]></KEY>
  <KEY value="FQDN"><![CDATA[2k3sp2-25.test.qualys.com, 2k8sp0-
25.test.qualys.com, s2016tp5dc.test.qualys.com]]></KEY>
  <KEY value="EXCLUDED_TARGET"><![CDATA[N/A]]></KEY>
...
</HEADER>
```

Scan Results in CSV format

We added "FQDN" in the CSV Header. This change only applies to CertView scans.

```
"Scan Results","05/15/2018 at 15:10:35 (GMT-0700)"
"Qualys, Inc.","919 E Hillsdale Blvd, Floor 4",,"Foster
City","California","United States of America","94404"
"Patrick Slimmer","qualys_ps","Manager"

"Launch Date","Active Hosts","Total Hosts","Type","Status",
"Reference","Scanner Appliance","Duration","Scan Title","Asset
Groups","IPs","Excluded IPs","Option Profile","FQDN"
"05/15/2018 at 14:26:21 (GMT-0700)","8","8","On-demand","Finished",
"scan/1526419582.05882","10.11.51.104 (Scanner 9.7.20-1, Vulnerability
Signatures 2.4.182-2)","00:13:01","My-CertView-Scan","My-Asset-
Group","10.10.31.58, 10.11.65.212-10.11.65.214",,"My-Option-
Profile","2k3sp2-25.test.qualys.com, 2k8sp0-25.test.qualys.com,
s2016tp5dc.test.qualys.com"
...
```

Patch Report is now available in XML format

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

You can now launch and download patch reports in XML format using the API and UI.

Example: Launch Patch Report

You can now specify output_format = xml when launching a patch report.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl"
"action=launch&report_type=Patch&output_format=xml&template_id=11102&repo
rt_title=APILaunchReport_patchn&ips=10.10.20.88"
"https://qualysapi.qualys.com/api/2.0/fo/report/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE SIMPLE_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/simple_return.dtd">
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2018-05-10T06:33:16Z</DATETIME>
    <TEXT>New report launched</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>34560</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Example: Download Patch Report

You can download patch reports in XML format from the UI or from the Report API (/api/2.0/fo/report/?action=fetch).

In this example, the patch report is grouped by host. The output will differ depending on the grouping method selected in the patch report template.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl"
"action=fetch&id=34560" "https://qualysapi.qualys.com/api/2.0/fo/report/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE PATCH_REPORT SYSTEM
"https://qualysapi.qualys.com/patch_report.dtd">
<PATCH_REPORT>
  <HEADER>
    <NAME><![CDATA[My Patch Report]]></NAME>
    <GENERATION_DATETIME>2018-06-01T06:23:14Z</GENERATION_DATETIME>
    <COMPANY_INFO>
      <NAME><![CDATA[Afco[Network]]]></NAME>
      <ADDRESS><![CDATA[500 First Street]]></ADDRESS>
      <CITY><![CDATA[Jersey City]]></CITY>
      <STATE><![CDATA[New Jersey]]></STATE>
      <COUNTRY><![CDATA[United States of America]]></COUNTRY>
      <ZIP_CODE><![CDATA[07303]]></ZIP_CODE>
    </COMPANY_INFO>
    <USER_INFO>
      <NAME><![CDATA[John Doe]]></NAME>
      <USERNAME>user_john</USERNAME>
      <ROLE>Manager</ROLE>
    </USER_INFO>
  </HEADER>
  <SUMMARY>
    <REPORT_SUMMARY>
      <TITLE><![CDATA[My Patch Report]]></TITLE>
      <ASSET_GROUPS><![CDATA[]]></ASSET_GROUPS>
      <IPS>N/A</IPS>
      <ASSET_TAGS><![CDATA[Included(all): PBPS-Targets;
Excluded(any);
]]></ASSET_TAGS>
      <GROUP_BY><![CDATA[Host]]></GROUP_BY>
      <CREATED_ON>06/01/2018</CREATED_ON>
      <NETWORK><![CDATA[Global Default Network]]></NETWORK>
    </REPORT_SUMMARY>
    <PATCH_SUMMARY>
      <TOTAL_PATCHES>646</TOTAL_PATCHES>
```

```
      <HOST_REQUIRING_PATCHES>8</HOST_REQUIRING_PATCHES>
      <VULN_ADDRESSED><![CDATA[761]]></VULN_ADDRESSED>
    </PATCH_SUMMARY>
  </SUMMARY>
  <PATCH_LIST_BY_HOST>
    <HOST_LIST>
      <HOST>
        <IP>10.20.31.244</IP>
        <DNS><![CDATA[]]></DNS>
        <NETBIOS><![CDATA[]]></NETBIOS>
        <OS><![CDATA[Linux 2.4-2.6 / Embedded Device / F5 Networks Big-IP
/ Linux 2.6]]></OS>

<OS_CPE><![CDATA[cpe:/o:oracle:oracle_linux:5.5::enterprise:]]></OS_CPE>
        <PATCH_COUNT><![CDATA[403]]></PATCH_COUNT>
        <NETWORK><![CDATA[Star Trek]]></NETWORK>
        <PATCH_LIST>
          <PATCH_INFO>
            <PATCH_QID>19589</PATCH_QID>
            <VENDOR_ID>CPUOCT2010</VENDOR_ID>
            <SEVERITY>5</SEVERITY>
            <PATCH_TITLE><![CDATA[Oracle Database October 2010 Security
Update Multiple Vulnerabilities]]></PATCH_TITLE>
            <VULN_COUNT>15</VULN_COUNT>
            <PATCH_PUBLISHED>10/12/2010</PATCH_PUBLISHED>
            <CVSS_BASE_SCORE>10</CVSS_BASE_SCORE>
            <CVSS3_BASE_SCORE>0</CVSS3_BASE_SCORE>
            <DETECTION_INFO>
              <DETECTION>
                <VULN_QID>19589</VULN_QID>
                <VULN_SEVERITY>5</VULN_SEVERITY>
                <VULN_TYPE><![CDATA[Practice]]></VULN_TYPE>
               <VULN_TITLE><![CDATA[Oracle Database October 2010 Security
Update Multiple Vulnerabilities]]></VULN_TITLE>

<DETECTION_INSTANCE><![CDATA[tcp/1521]]></DETECTION_INSTANCE>

<DETECTION_NORMALIZED_INSTANCE><![CDATA[]]></DETECTION_NORMALIZED_INSTANC
E>
                <DETECTION_DATE_LAST_FOUND><![CDATA[< 1 day
ago]]></DETECTION_DATE_LAST_FOUND>
                <CVSS_BASE_SCORE>10</CVSS_BASE_SCORE>
                <CVSS3_BASE_SCORE>0</CVSS3_BASE_SCORE>
              </DETECTION>
               .....
            </DETECTION_INFO>
          </PATCH_INFO>
           .....
        </PATCH_LIST>
```

```
      </HOST>
    </HOST_LIST>
    <PATCH_LINKS>
      <PATCH>
        <PATCH_QID>19589</PATCH_QID>
        <OS><![CDATA[]]></OS>

<LINK><![CDATA[https://support.oracle.com/CSP/main/article?cmd=show&type=
NOT&id=1159443.1]]></LINK>
      </PATCH>
      <PATCH>
        <PATCH_QID>19643</PATCH_QID>
        <OS><![CDATA[Oracle Database]]></OS>
        <LINK><![CDATA[https://support.oracle.com]]></LINK>
      </PATCH>
      .....
    </PATCH_LINKS>
  </PATCH_LIST_BY_HOST>
</PATCH_REPORT>
```

New DTD

The Patch Report DTD (patch_report.dtd) is used.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- QUALYS PATCH REPORT DTD -->

<!ELEMENT PATCH_REPORT (ERROR | (HEADER, (SUMMARY | (REPORT_SUMMARY,
PATCH_SUMMARY)), PATCH_LIST_BY_HOST?, PATCH_LIST_BY_AG?,
PATCH_LIST_BY_OS?, PATCH_LIST_BY_QID?, NON_RUNNING_KERNELS?))>
<!ELEMENT ERROR (#PCDATA)>
<!ATTLIST ERROR number CDATA #IMPLIED>

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

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

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

```
<!-- SUMMARY DETAILS -->
<!ELEMENT SUMMARY (REPORT_SUMMARY, PATCH_SUMMARY)>

<!ELEMENT REPORT_SUMMARY (TITLE, ASSET_GROUPS?, IPS?, ASSET_TAGS?,
GROUP_BY, CREATED_ON, NETWORK?)>
    <!ELEMENT ASSET_GROUPS (#PCDATA)>
    <!ELEMENT TITLE (#PCDATA)>
    <!ELEMENT IPS (#PCDATA)>
    <!ELEMENT ASSET_TAGS (#PCDATA)>
    <!ELEMENT GROUP_BY (#PCDATA)>
    <!ELEMENT CREATED_ON (#PCDATA)>

<!ELEMENT PATCH_SUMMARY (TOTAL_PATCHES, HOST_REQUIRING_PATCHES,
VULN_ADDRESSED)>
    <!ELEMENT TOTAL_PATCHES (#PCDATA)>
    <!ELEMENT HOST_REQUIRING_PATCHES (#PCDATA)>
    <!ELEMENT VULN_ADDRESSED (#PCDATA)>

<!-- PATCH_LIST_BY_HOST -->
<!ELEMENT PATCH_LIST_BY_HOST (HOST_LIST?, PATCH_LINKS?)>

<!-- PATCH_LIST_BY_ASSET GROUP -->
<!ELEMENT PATCH_LIST_BY_AG (ASSET_GROUPS_LIST, PATCH_LINKS?)>

<!ELEMENT ASSET_GROUPS_LIST (ASSET_GROUP*)>
<!ELEMENT ASSET_GROUP (NAME?, TOTAL_PATCHES?, HOST_NEEDING_PATCHES?,
TOTAL_DETECTION_FIXED?, HOST_LIST?)>
    <!ELEMENT HOST_NEEDING_PATCHES (#PCDATA)>
    <!ELEMENT TOTAL_DETECTION_FIXED (#PCDATA)>

<!-- PATCH_LIST_BY_QID -->
<!ELEMENT PATCH_LIST_BY_QID (PATCH_LIST, PATCH_LINKS?)>
        <!ELEMENT PATCH_LIST (PATCH_INFO*)>

<!-- PATCH_LIST_BY_OS -->
<!ELEMENT PATCH_LIST_BY_OS (OS_LIST?, PATCH_LINKS?)>

<!ELEMENT OS_LIST (OS_DETAILS*)>
<!ELEMENT OS_DETAILS (NAME?, TOTAL_PATCHES?,
SUMMARY_HOSTS_NEEDING_PATCHES?, SUMMARY_TOTAL_DETECTIONS_FIXED?,
PATCH_LIST)>
    <!ELEMENT SUMMARY_HOSTS_NEEDING_PATCHES (#PCDATA)>
    <!ELEMENT SUMMARY_TOTAL_DETECTIONS_FIXED (#PCDATA)>


<!ELEMENT HOST_LIST (HOST*)>


<!ELEMENT HOST (IP?,  DNS?, NETBIOS?, OS?, OS_CPE?, PATCH_COUNT?,
```

```
VULN_COUNT?, NETWORK?, PATCH_LIST?, DETECTION_INFO? )>
    <!ELEMENT IP (#PCDATA)>
    <!ELEMENT DNS (#PCDATA)>
    <!ELEMENT NETBIOS (#PCDATA)>
    <!ELEMENT OS (#PCDATA)>
    <!ELEMENT OS_CPE (#PCDATA)>
    <!ELEMENT PATCH_COUNT (#PCDATA)>
    <!ELEMENT NETWORK (#PCDATA)>

<!ELEMENT PATCH_INFO (PATCH_QID?, VENDOR_ID?, SEVERITY?, PATCH_TITLE?,
VULN_COUNT?,  HOST_COUNT?, PATCH_PUBLISHED?,  CVSS_BASE_SCORE?,
CVSS3_BASE_SCORE?, NETWORK?, DETECTION_INFO?, HOST_LIST?)>
    <!ELEMENT PATCH_QID (#PCDATA)>
    <!ELEMENT VENDOR_ID (#PCDATA)>
    <!ELEMENT SEVERITY (#PCDATA)>
    <!ELEMENT PATCH_TITLE (#PCDATA)>
    <!ELEMENT VULN_COUNT (#PCDATA)>
    <!ELEMENT HOST_COUNT (#PCDATA)>
    <!ELEMENT PATCH_PUBLISHED (#PCDATA)>
    <!ELEMENT CVSS_BASE_SCORE (#PCDATA)>
    <!ELEMENT CVSS3_BASE_SCORE (#PCDATA)>
    <!ELEMENT DETECTION_INFO (DETECTION*)>

        <!ELEMENT DETECTION (VULN_QID?, VULN_SEVERITY?, VULN_TYPE?,
VULN_TITLE?, DETECTION_INSTANCE?, DETECTION_NORMALIZED_INSTANCE?,
DETECTION_DATE_LAST_FOUND?,CVSS_BASE_SCORE?, CVSS3_BASE_SCORE?)>
            <!ELEMENT VULN_QID (#PCDATA)>
            <!ELEMENT VULN_SEVERITY (#PCDATA)>
            <!ELEMENT VULN_TYPE (#PCDATA)>
            <!ELEMENT VULN_TITLE (#PCDATA)>
            <!ELEMENT DETECTION_INSTANCE (#PCDATA)>
            <!ELEMENT DETECTION_NORMALIZED_INSTANCE (#PCDATA)>
            <!ELEMENT DETECTION_DATE_LAST_FOUND (#PCDATA)>

<!-- PATCH_LINKS -->
<!ELEMENT PATCH_LINKS (PATCH*)>

<!ELEMENT PATCH (PATCH_QID?, OS?, LINK?)>
<!ELEMENT NON_RUNNING_KERNELS (NON_RUNNING_KERNEL*)>
<!ELEMENT NON_RUNNING_KERNEL (QID?, IP?, SEVERITY?)>

<!ELEMENT LINK (#PCDATA)>
<!ELEMENT QID (#PCDATA)>
```

Option Profile - Import/Export Map Authentication

APIs affected	/api/2.0/fo/subscription/option_profile/
New or Updated API	Neither (schema change only)
DTD or XSD changes	Yes

We have added 2 new values for the tag <MAP_AUTHENTICATION> to support future capabilities: vCenter, none.

Also, the value VMware, available in previous release, is now renamed to VMware-ESXi.

XSD change

File: schemas/option_profile/option_profiles.xsd

```
<xs:element name="MAP_AUTHENTICATION" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="VMware-ESXi"/>
      <xs:enumeration value="vCenter"/>
      <xs:enumeration value="none"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```