



# Qualys Cloud Platform v3.x

## API Release Notes

Version 3.6

May 11, 2021

Qualys Cloud Suite API gives you many ways to integrate your programs and API calls with Qualys capabilities. 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

[AM API: VNet Details for Azure Assets](#)

[WAS API: Findings Output to Show the History of Findings](#)

[WAS API: Set "From Address" for Emails Sent on Scan Completion](#)

[WAS API: Assign System and Dynamic tags from the UI and API for the WAS module](#)

### 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 gateway URL for Qualys US Platform 1 (<https://gateway.qg1.apps.qualys.com>) in sample API requests. If you're on another platform, please replace this URL with the appropriate gateway URL for your account.

Qualys Cloud Platform 3.6 brings you many more API improvements and updates!

[Issues Addressed](#)

## AM API: VNet Details for Azure Assets

|                     |                                     |
|---------------------|-------------------------------------|
| API affected        | /qps/rest/2.0/get/am/hostasset/<id> |
| New or Updated APIs | Updated API                         |
| DTD or XSD changes  | Yes                                 |

We now fetch details and provide information about the virtual network (VNet) for your Microsoft Azure assets. Assets are returned when they are visible to the user (i.e. in the user's scope).

### Permissions

- You must have the AM module enabled.
- You must have the "API access" permission and Asset Management Permission "Read Asset"

### Sample: Get VNet Information for Azure assets

#### API Request

```
curl -n -u "USERNAME:PASSWORD"
"https://qualysapi.qualys.com/qps/rest/2.0/get/am/hostasset/2797037"
```

#### XML Output

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="https://qualysapi.qualys.com/xsd/2.0/am/ho
stasset.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <HostAsset>
      <id>2797037</id>
      <name>TestNullPublicIP</name>
      <created>2020-06-04T09:51:32Z</created>
      <modified>2021-02-19T18:13:41Z</modified>
      <type>HOST</type>
      <sourceInfo>
        <list>
          <AzureAssetSourceSimple>
            <assetId>2797037</assetId>
            <type>AZURE</type>
            <firstDiscovered>2020-06-
04T09:51:32Z</firstDiscovered>
            <lastUpdated>2021-02-19T18:13:41Z</lastUpdated>
            <azureVmTags>
```

```

        <tags>
          <list>
            <AzureTags>
              <key>Owner</key>
              <value>John Doe</value>
            </AzureTags>
            <AzureTags>
              <key>Department</key>
              <value>Engineering</value>
            </AzureTags>
          </list>
        </tags>
      </azureVmTags>
      <name>TestNullPublicIP</name>
      <location>westus2</location>
      <vmSize>Standard_B1ls</vmSize>
      <vmId>221ecf25-4bae-470f-89aa-58241eb874f2</vmId>
      <offer>CentOS</offer>
      <state>RUNNING</state>
      <publisher>OpenLogic</publisher>
      <version>latest</version>
      <osType>Linux</osType>
      <subnet>default</subnet>
      <subscriptionId>XXXXXXXX-XXXX-XXXX-XXXX-
XXXXXXXXXXXX</subscriptionId>
      <resourceGroupName>CV-VirtualMachines-
RG</resourceGroupName>
      <macAddress>00-0D-3A-F9-77-C1</macAddress>
      <privateIpAddress>172.16.3.4</privateIpAddress>
      <virtualNetwork>CV-VirtualMachines-RG-
vnet</virtualNetwork>
    </AzureAssetSourceSimple>
  </list>
</sourceInfo>
...
</HostAsset>
</data>
</ServiceResponse>

```

## XSD Updates

<platform API server>/qps/rest/2.0/hostasset.xsd

```

...
<element name="publisher" type="string" minOccurs="0" />
<element name="version" type="string" minOccurs="0" />
<element name="osType" type="string" minOccurs="0" />
<element name="subnet" type="string" minOccurs="0" />
<element name="ipv6" type="string" minOccurs="0" />

```

```
<element name="subscriptionId" type="string" minOccurs="0" />
<element name="resourceGroupName" type="string" minOccurs="0" />
<element name="macAddress" type="string" minOccurs="0" />
<element name="publicIpAddress" type="string" minOccurs="0" />
<element name="privateIpAddress" type="string" minOccurs="0" />
<element name="virtualNetwork" type="string" minOccurs="0" />

</sequence>
</extension>
</complexContent>
</complexType>
...
```

## WAS API: Findings Output to Show the History of Findings

|                     |                                                                        |
|---------------------|------------------------------------------------------------------------|
| API affected        | /qps/rest/3.0/get/was/finding/<id><br>/qps/rest/3.0/search/was/finding |
| New or Updated APIs | Updated                                                                |
| DTD or XSD changes  | Yes                                                                    |

With this release, we will now show in the Get and Search Findings API output the history of findings and each history will include the scan reference ID, scan ID and scan launch date.

### Permissions

- You must have the WAS module enabled.
- You must have the "API access" and "Access WAS module" permissions.

### Sample - Get details of a finding

The finding details show the history of finding.

#### API Request

```
curl -n -u "USERNAME:PASSWORD"  
"https://qualysapi.qualys.com/qps/rest/3.0/get/was/finding/701259"
```

#### XML Output

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="https://qualysapi.qualys.com/qps/xsd/3.0/w  
as/finding.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <data>  
    <Finding>  
      <id>701259</id>  
      <uniqueId>98a6ec79-98e4-4a19-988a-067c81b41dd2</uniqueId>  
      <qid>150210</qid>  
      <name><![CDATA[Information Disclosure via Response Header]]></name>  
      <type>INFORMATION_GATHERED</type>  
      <potential>>false</potential>  
      <findingType>QUALYS</findingType>  
      <group>IG_WEAK</group>  
      ...  
    </webApp>  
    <severityComment><![CDATA[Edit severity from 3 to 1 from WAS  
KB]]></severityComment>
```

```
<history>
  <set>
    <WebAppFindingHistory>
      <scanData>
        <id>4010576</id>
        <reference>was/1616077808388.1093760.4</reference>
        <launchedDate>2021-03-18T15:27:06Z</launchedDate>
      </scanData>
    </WebAppFindingHistory>
    <WebAppFindingHistory>
      <scanData>
        <id>4003754</id>
        <reference>was/1615905009787.1092030.4</reference>
        <launchedDate>2021-03-16T15:30:21Z</launchedDate>
      </scanData>
    <WebAppFindingHistory>
      <scanData>
        <id>3972937</id>
        <reference>was/1615041007684.1082193.4</reference>
        <launchedDate>2021-03-06T15:21:17Z</launchedDate>
      </scanData>
    </WebAppFindingHistory>
    ...
  </set>
</history>
</Finding>
</data>
</ServiceResponse>
```

## Sample - Search for a finding to view the finding history

### API Request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --
data-binary @-
"https://qualysapi.qualys.com/qps/rest/3.0/search/was/finding/" <
file.xml
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<ServiceRequest>
  <preferences>
    <verbose>true</verbose>
  </preferences>
  <filters>
    <Criteria field="uniqueId" operator="EQUALS">8c9c933c-e5c5-f77e-
e053-294f2c0ab892</Criteria>
  </filters>
</ServiceRequest>
```

## XML Output

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="https://qualysapi.qualys.com/qps/xsd/3.0/w
as/finding.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>50</count>
  <hasMoreRecords>true</hasMoreRecords>
  <lastId>2924970</lastId>
  <data>
    <Finding>
      <id>701259</id>
      <uniqueId>98a6ec79-98e4-4a19-988a-067c81b41dd2</uniqueId>
      <qid>150210</qid>
      <name><![CDATA[Information Disclosure via Response Header]]></name>
      <type>INFORMATION_GATHERED</type>
      <potential>>false</potential>
      <findingType>QUALYS</findingType>
      ...
    <history>
      <set>
        <WebAppFindingHistory>
          <scanData>
            <id>3683803</id>
            <reference>was/1604327406233.961814.4</reference>
            <launchedDate>2020-11-02T15:20:21Z</launchedDate>
          </scanData>
        </WebAppFindingHistory>
        <WebAppFindingHistory>
          <scanData>
            <id>3682382</id>
            <reference>was/1604154619159.959569.4</reference>
            <launchedDate>2020-10-31T15:21:50Z</launchedDate>
          </scanData>
        </WebAppFindingHistory>
        <WebAppFindingHistory>
          <scanData>
            <id>3679966</id>
            <reference>was/1603981827556.956713.4</reference>
            <launchedDate>2020-10-29T15:15:09Z</launchedDate>
          </scanData>
        </WebAppFindingHistory>
      </set>
    </history>
  </Finding>
  ...
</data>
</ServiceResponse>
```

## Updated XSD

<platform API server>/qps/xsd/3.0/was/finding.xsd

We added two new elements: "history" and "WebAppFindingHistory."

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">

  <!-- REQUEST -->
  <xs:element name="ServiceRequest">
    ...
    <xs:element name="history" type="HistoryList"/>
  </xs:all>
</xs:complexType>

...

<xs:complexType name="WebAppFindingHistory">
  <xs:sequence>
    <xs:element name="scan" type="WasScan"/>
  </xs:sequence>
</xs:complexType>

...

</xs:schema>
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">

  <!-- REQUEST -->
  <xs:element name="ServiceRequest">
    ...
    <xs:element name="cvssV3" type="CvssV3" minOccurs="0"/>
  </xs:all>
</xs:complexType>

<xs:complexType name="CvssV3">
  <xs:all>
    <xs:element name="base" type="xs:float"/>
    <xs:element name="temporal" type="xs:float"/>
    <xs:element name="attackVector" type="xs:string"/>
  </xs:all>
</xs:complexType>

...
```

## WAS API: Set "From Address" for Emails Sent on Scan Completion

|                     |                                                                                                                                         |
|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| API affected        | /qps/rest/3.0/get/was/wasscanschedule/<id><br>/qps/rest/3.0/create/was/wasscanschedule<br>/qps/rest/3.0/update/was/wasscanschedule/<id> |
| New or Updated APIs | Updated                                                                                                                                 |
| DTD or XSD changes  | Yes                                                                                                                                     |

We support "from address" as support@qualys.com or the owner's name like user@qualys.com from the Create/Update Schedule Scan > Settings tab for sending emails from UI. This same functionality is now supported from API.

We have added a new input parameter "sendMailFromAddressOption" to the Create and Update Schedule API to let you specify the sender of the email that is sent on the completion of the scheduled scan when creating or updating a scheduled scan on a web application.

To set this parameter, the sendMail parameter must be set to true. The SendMail parameter is set to true by default. If the sendMail parameter is true, then sendMailFromAddressOption is by default set to QUALYS\_SUPPORT. You can change the value of the sendMailFromAddressOption parameter to OWNER.

If you specify OWNER, then in the email the from address will show the email of the user whose account is used to create the schedule. If the sendMail parameter is false, then specifying sendMailFromAddressOption in the request will return an error in the response.

### Permissions

- You must have the WAS module enabled.
- You must have the "API access" and "Access WAS module" permissions.

### Sample - Create a Schedule

Let us create a schedule and set "sendMailFromAddressOption" as Owner.

#### API Request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/3.0/create/was/wasscanschedule"  
< file.xml
```

Note: "file.xml" contains the request POST data.

#### Request POST data

```
<ServiceRequest>
```

```
<data>
  <WasScanSchedule>
    <name>
      <![CDATA[Create Schedule]]>
    </name>
    <type>VULNERABILITY</type>
    <active>true</active>
    <scheduling>
      <cancelAfterNHours>8</cancelAfterNHours>
      <startDate>2021-03-25T09:50:11Z</startDate>
      <timeZone>
        <code>America/Vancouver</code>
        <offset>-07:00</offset>
      </timeZone>
      <occurrenceType>WEEKLY</occurrenceType>
      <occurrence>
        <weeklyOccurrence>
          <everyNWeeks>2</everyNWeeks>
          <occurrenceCount>20</occurrenceCount>
          <onDays>
            <WeekDay>SATURDAY</WeekDay>
            <WeekDay>SUNDAY</WeekDay>
          </onDays>
        </weeklyOccurrence>
      </occurrence>
    </scheduling>
    <notification>
      <active>true</active>
      <reschedule>true</reschedule>
      <delay>
        <nb>1</nb>
        <scale>DAY</scale>
      </delay>
      <message>
        <![CDATA[A Qualys scan is scheduled to start soon.]]>
      </message>
    </notification>
    <target>
      <webApp>
        <id>4830580</id>
      </webApp>
    </target>
    <profile>
      <id>1032294</id>
    </profile>
  </WasScanSchedule>
</sendMail>true</sendMail>
<sendMailFromAddressOption>OWNER</sendMailFromAddressOption>
</data>
```

```
</ServiceRequest>
```

### XML Output

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="https://qualysapi.qualys.com/qps/xsd/3.0/w
as/wasscanschedule.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  ...
    <sendMail>true</sendMail>
    <sendMailFromAddressOption>OWNER</sendMailFromAddressOption>
    <sendOneMail>false</sendOneMail>
    <enableWAFAuth>false</enableWAFAuth>
  </WasScanSchedule>
</data>
</ServiceResponse>
```

### **Sample - Update a Schedule**

Let us update a schedule and set the "sendMailFromAddressOption" parameter to OWNER.

### API Request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --
data-binary @-
"https://qualysapi.qualys.com/qps/rest/3.0/update/was/wasscanschedule/
807146" < file.xml
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<ServiceRequest>
  <data>
    <WasScanSchedule>
      <sendMailFromAddressOption>OWNER</sendMailFromAddressOption>
    </WasScanSchedule>
  </data>
</ServiceRequest>
```

### XML Output

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="https://qualysapi.qualys.com/qps/xsd/3.0/w
as/wasscanschedule.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <WasScanSchedule>
```

```
        <id>807146</id>
      </WasScanSchedule>
    </data>
  </ServiceResponse>
```

## Sample - Get Schedule Details

Let us view the from address set for the was scan schedule with ID 714393.

### API Request

```
curl -n -u "USERNAME:PASSWORD"
"https://qualysapi.qualys.com/qps/rest/3.0/get/was/wasscanschedule/714393"
```

### XML Output

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="https://qualysapi.qualys.com/qps/xsd/3.0/w
as/wasscanschedule.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  ...
    <sendMail>true</sendMail>
    <sendMailFromAddressOption>QUALYS_SUPPORT
  </sendMailFromAddressOption>
    <sendOneMail>false</sendOneMail>
    <enableWAFAuth>false</enableWAFAuth>
  </WasScanSchedule>
</data>
</ServiceResponse>
```

## Updated XSD

<platform API server>/qps/xsd/3.0/was/wasscanschedule.xsd

We added a new element "sendMailFromAddressOption." in the XSD.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  ...
    <xs:element name="sendOneMail" type="xs:boolean"
default="false" minOccurs="0"/>
    <xs:element name="sendMailFromAddressOption"
type="FromAddressOption" default="QUALYS_SUPPORT" minOccurs="0"/>
    <xs:element name="enableWAFAuth" type="xs:boolean"
default="false" minOccurs="0"/>
  </xs:all>
</xs:complexType>
```

## **WAS API: Assign System and Dynamic tags from the UI and API for the WAS module**

In the WAS module, you can now assign system and dynamic tags from the UI and API again. However, this is in violation of the defined Qualys Platform Tag behavior. Consider this as a formal deprecation notification. 60 days on, you will no longer be able to assign system and dynamic tags in the WAS module either from the UI or using the API.

## Issues Addressed

- We fixed an issue in the WAS API Guide where in the “Get Web Application Details” and in the sample response for the “View details for the web application” sections, we incorrectly mentioned that the web application screenshot is converted into the (.png) format instead of the “.jpg” format on encoding. We have fixed this issue in the API Guide.

In the same sections, we fixed an issue where we incorrectly mentioned that “urlSafe” was used to encode the web application screenshots. We removed this information from the respective sections.

- When the performance attribute was updated using WAS API, despite successful response, the update was not reflected. We have now fixed the issue so that update of the performance attribute using Option profile API is correctly reflected.
- A web application report successfully generated using API, had report ID missing on the WAS UI and was not downloadable either. We have now fixed this issue so that the web application report generated from API reflects correct ID on the UI and is downloadable as well.
- We fixed an issue where the “WAS Finding Count API” returned an incorrect finding count when the user after tagging several applications used the Finding Count API to find the total number of findings for the tagged web applications. After the fix, the Finding Count API is returning a correct number of findings for the tagged web applications.
- We fixed an issue where the user was getting an error when trying to download a particular Scan Report using the API: “qps/rest/3.0/download/was/wasscan/<id>”. After the fix, the WAS Scan report is downloadable from API.