



Qualys Cloud Platform v2.x

API Release Notes

Version 2.35.1.0

January 2, 2019

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

[Authentication API: Password visibility in Authentication Details](#)

[Option Profile API: Custom Scan Intensity](#)

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	<a href="https://qualysapi.<customer_base_url>">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.

Authentication API: Password visibility in Authentication Details

API affected	/qps/rest/3.0/get/was/webappauthrecord/<id>
New or Updated APIs	Updated
DTD or XSD changes	No

We have now added support to enable or disable visibility of password when you fetch authentication record details. You need to disable the "View Password in Authentication Record" and "View/download Selenium Script sensitive contents" permissions to mask the password in the response. If the permissions are enabled, the password will be visible in the response.

Sample 1 - Password is masked

Password fields are masked when sub user has disabled "View Password in Authentication Record" and "View/download Selenium Script sensitive contents" permissions.

API request:

```
curl -n -u "USERNAME:PASSWORD"  
"https://qualysapi.qualys.com/qps/rest/3.0/get/was/webappauthrecord/761533"
```

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/was/webappauthrecord.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <data>  
    <WebAppAuthRecord>  
      <id>761533</id>  
      <name><![CDATA[Selenium record]]></name>  
      <owner>  
        <id>75670165</id>  
        <username>quays_js</username>  
        <firstName>  
          <![CDATA[John]]>  
        </firstName>  
        <lastName>  
          <![CDATA[Smith]]>  
        </lastName>  
      </owner>  
      <formRecord>  
        <type>SELENIUM</type>
```

```
<seleniumScript>
  <name>
    <![CDATA[seleniumScript]]>
  </name>
  <data>
    <![CDATA[
      <?xml version="1.0" encoding="UTF-8"?>
      <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
      <html xmlns="http://www.w3.org/1999/xhtml"
xml:lang="en" lang="en">
        <head>
          <meta http-equiv="Content-Type"
content="text/html; charset=UTF-8" />
          <link rel="selenium.base"
href="https://10.113.195.231/" />
          <title>AuthScript</title>
        </head>
        <body>
          <table cellpadding="1" cellspacing="1"
border="1">
            <thead>
              <tr>
                <td rowspan="1"
colspan="3">AuthScript</td>
            </thead>
            <tbody>
              <tr>
                <td>open</td>
                <td>@@webappURL@@</td>
                <td></td>
              </tr>
              <tr>
                <td>click</td>
                <td>name=username</td>
                <td></td>
              </tr>
              <tr>
                <td>type</td>
                <td>name=username</td>
                <td>*****</td>
              </tr>
              <tr>
                <td>type</td>
                <td>name=password</td>
                <td>*****</td>
              </tr>
            </tbody>
          </table>
        </body>
      </html>
    </![CDATA[
  </data>
</seleniumScript>
```

```
 click | name=Login |
```

```

</data>
<regex>
  <![CDATA[selenium]]>
</regex>
</seleniumScript>
</formRecord>
<serverRecord>
  <fields>
    <count>3</count>
    <list>
      <WebAppAuthServerRecordField>
        <id>730020</id>
        <type>BASIC</type>
        <domain>
          <![CDATA[comp]]>
        </domain>
        <username>
          <![CDATA[abc]]>
        </username>
        <password>
          <![CDATA[*****]]>
        </password>
      </WebAppAuthServerRecordField>
      <WebAppAuthServerRecordField>
        <id>730021</id>
        <type>NTLM</type>
        <username>
          <![CDATA[abc3]]>
        </username>
        <password>
          <![CDATA[*****]]>
        </password>
      </WebAppAuthServerRecordField>
      <WebAppAuthServerRecordField>
        <id>730022</id>
        <type>DIGEST</type>
        <domain>
          <![CDATA[comp2]]>
        </domain>
        <username>
          <![CDATA[abc2]]>
        </username>

```

```
        <password>
          <![CDATA[*****]]>
        </password>
      </WebAppAuthServerRecordField>
    </list>
  </fields>
</serverRecord>
...
</updatedBy>
</WebAppAuthRecord>
</data>
</ServiceResponse>
```

Sample 2 - Password is visible

Password fields are visible when sub user has enabled "View Password in Authentication Record" and "View/download Selenium Script sensitive contents" permissions.

API request:

```
curl -n -u "USERNAME:PASSWORD"
"https://qualysapi.qualys.com/qps/rest/3.0/get/was/webappauthrecord/76153
4"
```

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/webappauthrecord.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <WebAppAuthRecord>
      <id>762380</id>
      <name>
        <![CDATA[Selenium with server authentication]]>
      </name>
      <owner>
        <id>75913465</id>
        <username>quays_js2</username>
        <firstName>
          <![CDATA[John]]>
        </firstName>
        <lastName>
          <![CDATA[Smith]]>
        </lastName>
      </owner>
      <formRecord>
        <type>SELENIUM</type>
```

```
<seleniumScript>
  <name>
    <![CDATA[seleniumScript]]>
  </name>
  <data>
    <![CDATA[
      <?xml version="1.0" encoding="UTF-8"?>
      <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
      <html xmlns="http://www.w3.org/1999/xhtml"
xml:lang="en" lang="en">
        <head>
          <meta http-equiv="Content-Type"
content="text/html; charset=UTF-8" />
          <link rel="selenium.base"
href="https://10.113.195.231/" />
          <title>AuthScript</title>
        </head>
        <body>
          <table cellpadding="1" cellspacing="1"
border="1">
            <thead>
              <tr>
                <td rowspan="1"
colspan="3">AuthScript</td>
            </thead>
            <tbody>
              <tr>
                <td>open</td>
                <td>@@webappURL@@</td>
                <td></td>
              </tr>
              <tr>
                <td>click</td>
                <td>name=username</td>
                <td></td>
              </tr>
              <tr>
                <td>type</td>
                <td>name=username</td>
                <td>theuser</td>
              </tr>
              <tr>
                <td>type</td>
                <td>name=password</td>
                <td>thepass</td>
              </tr>
            </tbody>
          </table>
        </body>
      </html>
    </![CDATA[
  </data>
</seleniumScript>
```

```
<td>click</td>
<td>name=Login</td>
<td></td>
```

```
        <password>
          <![CDATA[1234]]>
        </password>
      </WebAppAuthServerRecordField>
    </list>
  </fields>
</serverRecord>
<tags>
  <count>0</count>
</tags>
<comments>
  <count>0</count>
</comments>
<createdDate>2018-12-31T06:30:24Z</createdDate>
<createdBy>
  <id>75913465</id>
  <username>quays_js2</username>
  <firstName>
    <![CDATA[John]]>
  </firstName>
  <lastName>
    <![CDATA[Smith]]>
  </lastName>
</createdBy>
<updatedAt>2018-12-31T06:30:24Z</updatedAt>
<updatedBy>
  <id>75913465</id>
  <username>quays_js2</username>
  <firstName>
    <![CDATA[John]]>
  </firstName>
  <lastName>
    <![CDATA[Smith]]>
  </lastName>
</updatedBy>
</WebAppAuthRecord>
</data>
</ServiceResponse>
```

Option Profile API: Custom Scan Intensity

API affected	/qps/rest/3.0/get/was/optionprofile/<id> /qps/rest/3.0/create/was/optionprofile /qps/rest/3.0/update/was/optionprofile/<id> /qps/rest/3.0/get/was/wasscan/<id>
New or Updated APIs	Updated
DTD or XSD changes	Yes

We have now introduced new parameters so that you can define your custom scan intensity in the option profile and thus control the scan performance accordingly to your configured settings. Using our new parameter <customperformance> you can further configure the number of threads to be used to scan each host and the delay between requests.

Input Parameters

New elements for input parameter is described below.

Parameter	Description
customPerformance*	You can now set the overall intensity level for web application scans. Example: <customPerformance> <numOfHttpThreads>10</numOfHttpThreads> <delayBetweenRequests>5</delayBetweenRequests> </customPerformance> Note: performance and customPerformance are mutually exclusive parameters and cannot be used together. You can use only either of them for an option profile.
numOfHttpThreads	(integer) Number of threads to be used to scan each host. The valid range is from 1 to 10.
delayBetweenRequests	(integer) The duration of delay introduced by WAS in between the scanning engine requests sent to the applications server. The valid range is from 0 to 2000 milliseconds.

Sample 1 - Custom Scan Intensity

Let us create an Option Profile with customized scan intensity.

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/optionprofile" <
```

file.xml

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

Request POST Data:

```
<ServiceRequest>
<data>
  <OptionProfile>
    <name><![CDATA[Option Profile with Custom Scan Intensity]]></name>
    <customPerformance>
      <numOfHttpThreads>5</numOfHttpThreads>
      <delayBetweenRequests>100</delayBetweenRequests>
    </customPerformance>
  </OptionProfile>
</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/optionprofile.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <OptionProfile>
      <id>1608560</id>
      <name>
        <![CDATA[Option Profile with Custom Scan Intensity]]>
      </name>
      ...
      <smartScanSupport>false</smartScanSupport>
      <customPerformance>
        <numOfHttpThreads>5</numOfHttpThreads>
        <delayBetweenRequests>100</delayBetweenRequests>
      </customPerformance>
      <bruteforceOption>MINIMAL</bruteforceOption>
      ...
    </OptionProfile>
  </data>
</ServiceResponse>
```

Sample 2 - Update Option Profile for Custom Scan Intensity

Let us update an Option Profile with customized scan intensity.

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/optionprofile/16085  
60" < file.xml
```

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

Request POST Data:

```
<ServiceRequest>  
<data>  
  <OptionProfile>  
    <name><![CDATA[Update Option Profile with Custom Scan  
Intensity]]></name>  
    <customPerformance>  
      <numOfHttpThreads>10</numOfHttpThreads>  
      <delayBetweenRequests>20</delayBetweenRequests>  
    </customPerformance>  
  </OptionProfile>  
</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/optionprofile.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <data>  
    <OptionProfile>  
      <id>1608560</id>  
    </OptionProfile>  
  </data>  
</ServiceResponse>
```

Sample 3 - Fetch details Option Profile with Custom Scan Intensity

Let us get details of an Option Profile with customized scan intensity.

API request:

```
curl -u "USERNAME:PASSWORD"  
"https://qualysapi.qualys.com/qps/rest/3.0/get/was/optionprofile/1608560"
```

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/was/optionprofile.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <OptionProfile>
      <id>1608560</id>
      <name>
        <![CDATA[Update Option Profile with Custom Scan
Intensity]]>
      </name>
      ...
      <smartScanSupport>false</smartScanSupport>
      <customPerformance>
        <numOfHttpThreads>10</numOfHttpThreads>
        <delayBetweenRequests>20</delayBetweenRequests>
      </customPerformance>
      <bruteforceOption>MINIMAL</bruteforceOption>
      ...
    </OptionProfile>
  </data>
</ServiceResponse>
```

Sample 4 - Get details of a scan that uses Custom scan Intensity

API request:

```
curl -u "USERNAME:PASSWORD"
"https://https://qualysapi.qualys.com/qps/rest/3.0/get/was/wasscan/3214788"
```

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/was/wasscan.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <WasScan>
      <id>3214788</id>
      <name>
        <![CDATA[New Scan]]>
      </name>
      <reference>was/1546407992024.4625855</reference>
      <type>DISCOVERY</type>
      <mode>API</mode>
      ...
    </WasScan>
  </data>
</ServiceResponse>
```

```

    <profile>
      <id>1608560</id>
      <name>
        <![CDATA[Updated Option Profile]]>
      </name>
    </profile>
    <options>
      <count>17</count>
      <list>
        ...
        <WasScanOption>
          <name>Delay Between Requests (ms)</name>
          <value>
            <![CDATA[20]]>
          </value>
        </WasScanOption>
        <WasScanOption>
          <name># of HTTP Threads</name>
          <value>
            <![CDATA[10]]>
          </value>
        </WasScanOption>
        ...
        <WasScanOption>
          <name>Performance Settings</name>
          <value>
            <![CDATA[Custom]]>
          </value>
        </WasScanOption>
        ...
      </WasScan>
    </data>
  </ServiceResponse>

```

XSD update

Changes in optionprofile.xsd (/qps/xsd/3.0/was/optionprofile.xsd)

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">
  ...
  </xs:simpleType>
  </xs:element>
  <xs:element name="customPerformance" type="CustomPerformance"
    minOccurs="0"/>
  <xs:element name="bruteforceOption" default="MINIMAL"
    minOccurs="0">
    <xs:simpleType>

```

```
...
<xs:complexType name="User">
  <xs:all>
    <xs:element name="id" type="xs:long"/>
    <xs:element name="username" type="xs:string" minOccurs="0"/>
    <xs:element name="firstName" type="Cdata" minOccurs="0"/>
    <xs:element name="lastName" type="Cdata" minOccurs="0"/>
  </xs:all>
</xs:complexType>
<xs:complexType name="CustomPerformance">
  <xs:all>
    <xs:element name="numOfHttpThreads" type="xs:long" minOccurs="1"/>
    <xs:element name="delayBetweenRequests" type="xs:long"
minOccurs="1"/>
  </xs:all>
</xs:complexType>

<xs:complexType name="TagList">
  <xs:all>
  ...
```