



Qualys Cloud Platform (VM, PC) v8.x

API Release Notes

Version 8.19

April 15, 2019

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

[Sybase Authentication - Password Encryption and Auto Discover Databases](#)

[PC - New MS Exchange Server Authentication API](#)

[New Support for Microsoft Azure Key Vault](#)

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.

Sybase Authentication - Password Encryption and Auto Discover Databases

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

This release introduces 2 new options for Sybase authentication - Password Encryption and Auto Discover Databases.

Password Encryption - Enable this option when your Sybase database instance requires an encrypted password for successful login. If password encryption is required and you do not enable this option then authentication will fail.

Auto Discover Databases - Enable this option and we'll find all Sybase database names on each host for you. This means you no longer have to create a separate Sybase record for each database name. Create one record with Auto Discover Databases enabled to authenticate to multiple databases on the same host.

Input Parameter

Parameter	Description
password_encryption={0 1}	(Optional to create or update record) When set to 1, password encryption is enabled in the Sybase record. When set to 0 (the default), password encryption is not enabled.
auto_discover_databases={0 1}	(Optional to create or update record) Specify auto_discover_databases=1 and we will find all Sybase database instances on the target host. Note you must either enter a database name (with existing database parameter) OR use the Auto Discover option. When unspecified (auto_discover_databases=0), we will not auto discover database instances and look for the database name that you have entered in the database parameter.

List Sybase records

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>  
<!DOCTYPE AUTH_SYBASE_LIST_OUTPUT SYSTEM  
"https://qualysapi.qualys.com/api/2.0/fo/auth/sybase/auth_sybase_list_out  
put.dtd">  
<AUTH_SYBASE_LIST_OUTPUT>  
  <RESPONSE>  
    <DATETIME>2017-04-10T21:32:21Z</DATETIME>  
    <AUTH_SYBASE_LIST>  
      <AUTH_SYBASE>  
        <ID>78177</ID>  
        <TITLE><![CDATA[api_syb_basic_2IPs_NW2]]></TITLE>  
        <USERNAME><![CDATA[api_user1]]></USERNAME>  
        <AUTO_DISCOVER_DATABASES>1</AUTO_DISCOVER_DATABASES>  
        <PORT>444</PORT>  
        <INSTALLATION_DIR><![CDATA[dir123]]></INSTALLATION_DIR>  
        <PASSWORD_ENCRYPTION><![CDATA[1]]></PASSWORD_ENCRYPTION>  
      ...
```

Create Sybase Record to enable password encryption and auto discovery

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl Sample" -d  
"action=create&title=sybase_record&network_id=19015&username=acme_  
ac12&password=password&password_encryption=1&ips=10.10.24.12&auto_discover_  
r_databases=1&port=444&installation_dir=/dir123&comments=This%20  
Sybase%20comments"  
"https://qualysapi.qualys.com/api/2.0/fo/auth/sybase/" > file.xml
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>  
<!DOCTYPE BATCH_RETURN SYSTEM  
"http://qualysapi.qualys.com/api/2.0/batch_return.dtd">  
<BATCH_RETURN>  
  <RESPONSE>  
    <DATETIME>2019-03-28T15:45:05Z</DATETIME>  
    <BATCH_LIST>  
      <BATCH>
```

```
<TEXT>Successfully Created</TEXT>
<ID_SET>
  <ID>43025</ID>
</ID_SET>
</BATCH>
</BATCH_LIST>
</RESPONSE>
</BATCH_RETURN>
```

Update Sybase Record to disable auto discovery

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl Sample" -d
"action=update&ids=43025&auto_discover_databases=0&database=sybaseDB1&add
_ips=10.10.26.238,10.10.24.13,10.10.24.15&installation_dir=/dir123&commen
ts=This%20Sybase%20comments"
"https://qualysapi.qualys.com/api/2.0/fo/auth/sybase/" > file.xml
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE BATCH_RETURN SYSTEM
"http://qualysapi.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
  <RESPONSE>
    <DATETIME>2019-03-28T15:45:05Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>43025</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Updated DTD

<base_url>/api/2.0/fo/auth/sybase/auth_sybase_list_output.dtd

The elements AUTO_DISCOVER_DATABASES and PASSWORD_ENCRYPTION were added to the DTD. Database is now optional parameter (it was required before).

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

<!ELEMENT REQUEST (DATETIME, USER_LOGIN, RESOURCE, PARAM_LIST?,
```

```

POST_DATA?)>
...
<!ELEMENT PARAM (KEY, VALUE)>
<!ELEMENT KEY (#PCDATA)>
<!ELEMENT VALUE (#PCDATA)>
<!-- if returned, POST_DATA will be urlencoded -->
<!ELEMENT POST_DATA (#PCDATA)>

<!ELEMENT RESPONSE (DATETIME, (AUTH_SYBASE_LIST|ID_SET)?, WARNING_LIST?,
GLOSSARY?)>
<!ELEMENT AUTH_SYBASE_LIST (AUTH_SYBASE+)>

<!ELEMENT AUTH_SYBASE (ID, TITLE, USERNAME, (DATABASE |
AUTO_DISCOVER_DATABASES), PORT, INSTALLATION_DIR?, PASSWORD_ENCRYPTION?,
IP_SET?, LOGIN_TYPE?, DIGITAL_VAULT?, NETWORK_ID?, CREATED,
LAST_MODIFIED, COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT DATABASE (#PCDATA)>
<!ELEMENT PORT (#PCDATA)>
<!ELEMENT INSTALLATION_DIR (#PCDATA)>
<!ELEMENT AUTO_DISCOVER_DATABASES (#PCDATA)>
<!ELEMENT PASSWORD_ENCRYPTION (#PCDATA)>
<!ELEMENT IP_SET (IP|IP_RANGE)+>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT IP_RANGE (#PCDATA)>
...

<!ELEMENT WARNING_LIST (WARNING+)>
<!ELEMENT WARNING (CODE?, TEXT, URL?, ID_SET?)>
<!ELEMENT CODE (#PCDATA)>
<!ELEMENT TEXT (#PCDATA)>
<!ELEMENT URL (#PCDATA)>
<!ELEMENT ID_SET (ID|ID_RANGE)+>
<!ELEMENT ID_RANGE (#PCDATA)>

<!ELEMENT GLOSSARY (USER_LIST?)>
<!ELEMENT USER_LIST (USER+)>
<!ELEMENT USER (USER_LOGIN, FIRST_NAME, LAST_NAME)>
<!ELEMENT FIRST_NAME (#PCDATA)>
<!ELEMENT LAST_NAME (#PCDATA)>
<!-- EOF -->

```

PC - New MS Exchange Server Authentication API

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

Microsoft Exchange Server authentication is now supported for compliance scans. The new MS Exchange Server Authentication API (api/2.0/fo/auth/ms_exchange/) lets you list, create, update and delete MS Exchange Server authentication records. User permissions for this API are the same as other authentication record APIs. Note that the API supports authentication record creation only for MS Exchange Server installed on Windows.

List all record types

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

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>  
<!DOCTYPE AUTH_RECORDS_OUTPUT SYSTEM  
<RESPONSE>  
...  
</AUTH_JBOSS_IDS>  
  <AUTH_INFORMIXDB_IDS>  
    <ID_SET>  
      <ID>1051295</ID>  
      <ID_RANGE>1061837-1061838</ID_RANGE>  
    </ID_SET>  
  <AUTH_MS_EXCHANGE_IDS>  
    <ID_SET>  
      <ID>1089443</ID>  
    </ID_SET>  
  </AUTH_MS_EXCHANGE_IDS>  
</AUTH_RECORDS>  
</RESPONSE>  
...
```

Updated DTD

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

The element AUTH_MS_EXCHANGE_IDS was added to identify MS Exchange Server record IDs.

```
<!-- QUALYS AUTH_RECORDS_OUTPUT DTD -->

<!ELEMENT AUTH_RECORDS_OUTPUT (REQUEST?, RESPONSE)>
...
<!ELEMENT AUTH_RECORDS (AUTH_UNIX_IDS?, AUTH_WINDOWS_IDS?,
AUTH_ORACLE_IDS?, AUTH_ORACLE_LISTENER_IDS?, AUTH_SNMP_IDS?,
AUTH_MS_SQL_IDS?, AUTH_IBM_DB2_IDS?, AUTH_VMWARE_IDS?, AUTH_MS_IIS_IDS?,
AUTH_APACHE_IDS?, AUTH_IBM_WEBSPPHERE_IDS?, AUTH_HTTP_IDS?,
AUTH_SYBASE_IDS?, AUTH_MYSQL_IDS?, AUTH_TOMCAT_IDS?,
AUTH_ORACLE_WEBLOGIC_IDS?, AUTH_DOCKER_IDS?, AUTH_POSTGRESQL_IDS?,
AUTH_MONGODB_IDS?, AUTH_PALO_ALTO_FIREWALL_IDS?, AUTH_VCENTER_IDS?,
AUTH_JBOSS_IDS?, AUTH_MARIADB_IDS?, AUTH_MS_EXCHANGE_IDS?)>

...
<!ELEMENT AUTH_JBOSS_IDS (ID_SET)>
<!ELEMENT AUTH_MARIADB_IDS (ID_SET)>
<!ELEMENT AUTH_MS_EXCHANGE_IDS (ID_SET)>
...
```


List MS Exchange Server records

Use the new MS Exchange Server Authentication Record List API (/api/2.0/fo/auth/ms_exchange/?action=list) to list MS Exchange Server records.

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>  
<!DOCTYPE AUTH_MS_EXCHANGE_LIST_OUTPUT SYSTEM  
"https://qualysapi.qualys.com/api/2.0/fo/auth/ms_exchange/auth_ms_exchang  
e_list_output.dtd">  
<AUTH_MS_EXCHANGE_LIST_OUTPUT>  
  <RESPONSE>  
    <DATETIME>2019-03-20T07:26:38Z</DATETIME>  
    <AUTH_MS_EXCHANGE_LIST>  
      <AUTH_MS_EXCHANGE>  
        <ID>48050</ID>  
        <TITLE>  
          <![CDATA[msexchange01]]>  
        </TITLE>  
        <IP_SET>  
          <IP>10.10.10.10</IP>  
        </IP_SET>  
        <NETWORK_ID>0</NETWORK_ID>  
        <CREATED>  
          <DATETIME>2019-03-14T07:05:05Z</DATETIME>  
          <BY>quays_sp1</BY>  
        </CREATED>  
        <LAST_MODIFIED>  
          <DATETIME>2019-03-14T07:05:05Z</DATETIME>  
        </LAST_MODIFIED>  
        <COMMENTS>  
          <![CDATA[msexchange]]>  
        </COMMENTS>  
      </AUTH_MS_EXCHANGE>  
      <AUTH_MS_EXCHANGE>  
        <ID>49025</ID>  
        <TITLE>  
          <![CDATA[ms exchange]]>  
        </TITLE>  
        <IP_SET>  
          <IP>10.10.10.11</IP>  
        </IP_SET>  
        <NETWORK_ID>0</NETWORK_ID>
```

```
<CREATED>
  <DATETIME>2019-03-19T08:20:31Z</DATETIME>
  <BY>quays_sp1</BY>
</CREATED>
<LAST_MODIFIED>
  <DATETIME>2019-03-19T08:20:31Z</DATETIME>
</LAST_MODIFIED>
<COMMENTS>
  <![CDATA[new]]>
</COMMENTS>
</AUTH_MS_EXCHANGE>
<AUTH_MS_EXCHANGE>
  <ID>49026</ID>
  <TITLE>
    <![CDATA[apicreate]]>
  </TITLE>
  <IP_SET>
    <IP>10.10.10.13</IP>
  </IP_SET>
  <NETWORK_ID>0</NETWORK_ID>
  <CREATED>
    <DATETIME>2019-03-19T11:46:23Z</DATETIME>
    <BY>quays_sp1</BY>
  </CREATED>
  <LAST_MODIFIED>
    <DATETIME>2019-03-19T11:56:57Z</DATETIME>
  </LAST_MODIFIED>
  <COMMENTS>
    <![CDATA[editapicomment]]>
  </COMMENTS>
</AUTH_MS_EXCHANGE>
</AUTH_MS_EXCHANGE_LIST>
</RESPONSE>
</AUTH_MS_EXCHANGE_LIST_OUTPUT>
<!-- CONFIDENTIAL AND PROPRIETARY INFORMATION. Qualys provides the
QualysGuard Service "As Is," without any warranty of any kind. Qualys
makes no warranty that the information contained in this report is
complete or error-free. Copyright 2019, Qualys, Inc. //-->
```

New DTD:

```
<baseurl>/api/2.0/fo/auth/ms_exchange/auth_ms_exchange_list_output.dtd

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

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

<!ELEMENT RESPONSE (DATETIME, (AUTH_MS_EXCHANGE_LIST|ID_SET)?,
WARNING_LIST?, GLOSSARY?)>
<!ELEMENT AUTH_MS_EXCHANGE_LIST (AUTH_MS_EXCHANGE+)>

<!ELEMENT AUTH_MS_EXCHANGE (ID, TITLE, IP_SET, NETWORK_ID?, CREATED,
LAST_MODIFIED, COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>

<!ELEMENT IP_SET (IP|IP_RANGE)+>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT IP_RANGE (#PCDATA)>
<!ELEMENT NETWORK_ID (#PCDATA)>

<!ELEMENT CREATED (DATETIME, BY)>
<!ELEMENT BY (#PCDATA)>
<!ELEMENT LAST_MODIFIED (DATETIME)>
<!ELEMENT COMMENTS (#PCDATA)>

<!ELEMENT WARNING_LIST (WARNING+)>
<!ELEMENT WARNING (CODE?, TEXT, URL?, ID_SET?)>
<!ELEMENT CODE (#PCDATA)>
<!ELEMENT TEXT (#PCDATA)>
<!ELEMENT URL (#PCDATA)>
<!ELEMENT ID_SET (ID|ID_RANGE)+>
<!ELEMENT ID_RANGE (#PCDATA)>

<!ELEMENT GLOSSARY (USER_LIST?)>
<!ELEMENT USER_LIST (USER+)>
<!ELEMENT USER (USER_LOGIN, FIRST_NAME, LAST_NAME)>
```

```
<!ELEMENT FIRST_NAME (#PCDATA) >  
<!ELEMENT LAST_NAME (#PCDATA) >  
  
<!-- EOF -->
```

Create/Update MS Exchange Server record

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

Parameter	Description
action={action}	(Required) Specify create, update, delete (using POST) or list (using GET or POST).
echo_request={0 1}	(Optional) Specify 1 to view (echo) input parameters in the XML output. By default these are not included.
ids={value}	(Required to update or delete record) Record MS Exchange type auth record IDs to update/delete. Specify record IDs and/or ID ranges (for example, 1359-1407). Multiple entries are comma separated.
title={value}	(Required to create record) A title for the record. The title must be unique. Maximum 255 characters (ascii).
comments={value}	(Optional to create or update record) User defined comments. Maximum of 1999 characters.
Target Hosts	
ips={value}	(Required to create record) The IP address(es) the server will log into using the record's credentials. Multiple entries are comma separated. (Optional to update record) IPs specified will overwrite existing IPs in the record, and existing IPs will be removed.
add_ips={value}	(Optional to update record) Add IPs to the IPs list for this record. Multiple IPs/ranges are comma separated.
remove_ips={value}	(Optional to update record) IPs to be removed from your record. You may enter a combination of IPs and ranges. Multiple entries are comma separated. This parameter and the ips parameter cannot be specified in the same request.
network_id={value}	(Optional and valid when the networks feature is enabled) The network ID for the record.

Example: Create MS Exchange Server record

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl sample" -d  
"action=create&network_id=0&title=fordeltes&comments=editapicomment&ips=1  
0.10.10.31"  
"https://qualysapi.qualys.com/api/2.0/fo/auth/ms_exchange/"
```

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>2019-03-20T08:26:54Z</DATETIME>  
    <BATCH_LIST>  
      <BATCH>  
        <TEXT>Successfully Created</TEXT>  
        <ID_SET>  
          <ID>49029</ID>  
        </ID_SET>  
      </BATCH>  
    </BATCH_LIST>  
  </RESPONSE>  
</BATCH_RETURN>
```

Example: Update MS Exchange Server record

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl sample" -d  
"action=update&ids=49029&title=forupdate&comments=editwapicomment&ips=10.  
10.10.11"  
"https://qualysapi.qualys.com/api/2.0/fo/auth/ms_exchange/"
```

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>2019-03-20T08:29:48Z</DATETIME>  
    <BATCH_LIST>  
      <BATCH>  
        <TEXT>Successfully Updated</TEXT>  
        <ID_SET>  
          <ID>49029</ID>  
        </ID_SET>  
      </BATCH>  
    </BATCH_LIST>  
  </RESPONSE>  
</BATCH_RETURN>
```

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

Delete MS Exchange Server records

Use these parameters to delete records.

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

Example: Delete MS Exchange Server record (single)

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl sample" -d  
"action=delete&ids=49026"  
"https://qualysapi.qualys.com/api/2.0/fo/auth/ms_exchange/"
```

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>2019-03-20T07:56:00Z</DATETIME>  
    <BATCH_LIST>  
      <BATCH>  
        <TEXT>Successfully Deleted</TEXT>  
        <ID_SET>  
          <ID>49026</ID>  
        </ID_SET>  
      </BATCH>  
    </BATCH_LIST>  
  </RESPONSE>  
</BATCH_RETURN>
```

Example: Delete MS Exchange Server records (bulk)

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl sample" -d  
"action=delete&ids=49028,49029"
```

"https://qualysapi.qualys.com/api/2.0/fo/auth/ms_exchange/"

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>2019-03-20T08:31:35Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Deleted</TEXT>
        <ID_SET>
          <ID_RANGE>49028-49029</ID_RANGE>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

New Support for Microsoft Azure Key Vault

APIs affected	/api/2.0/fo/vault
New or Updated API	Updated
DTD or XSD changes	Yes
APIs affected	/api/2.0/fo/auth/windows/ /api/2.0/fo/auth/unix/ /api/2.0/fo/auth/ms_sql/ /api/2.0/fo/auth/mysql/ /api/2.0/fo/auth/mariadb/ /api/2.0/fo/auth/mongodb/ /api/2.0/fo/auth/oracle/ /api/2.0/fo/auth/postgresql/
New or Updated API	Updated
DTD or XSD changes	No

This new vault type can be used to retrieve authentication credentials from an Azure key 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.

Authentication Vault API

You can now create, update, list and view authentication records for Azure key vaults.

Create/Update Azure Key Authentication Vault

Use the parameter “action=create” or “action=update” to create/update a new Azure key vault in your account.

Parameter	Description
action=create	(Required)
title={value}	(Required to create and optional to update vault) The vault title.
type={value}	(Required to create and optional to update vault) Specify type=Azure Key
comments={value}	(Optional) User defined comments.
url={value}	(Required to create and optional to update vault) The HTTP or HTTPS URL to access the Azure key Vault HTTP API. The HTTPS URL is required if the ssl_verify parameter is set 1.
app_id={value}	(Required to create and optional to update vault) The application ID associated with the application created in the Azure Key Vault.

Parameter	Description
ssl_verify={0 1}	(Required to create and optional to update vault) 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.
cert={value}	(Required to create and optional to update vault) The client certificate for authentication. Enter the certificate block after the key block and be sure to include the first and last line (-----BEGIN CERTIFICATE----- and -----END CERTIFICATE-----). For a create/update request, if the cert parameter is specified, then the private_key parameter must also be specified.
private_key={value}	(Required to create and optional to update vault) The private key for authentication. Copy the contents of private key file (id_rsa) and be sure to include the first and last line (-----BEGIN PRIVATE KEY----- and -----END PRIVATE KEY-----).
passphrase={value}	(Optional) The private key passphrase is required if the private key is encrypted.

Sample - Create Azure Key Vault

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d  
"action=create&type=Azure Key&title=My Azure key  
Vault&url=https://azkey.com&app_id=app123&ssl_verify=1&cert=-----  
BEGIN CERTIFICATE-----  
MIIEYjCCArKgAwIBAgIBAjANBgkqhkiG9w0BAQUFADCBljEbMBkGA1UE  
AwwSU2Nh-----END CERTIFICATE-----&private_key=key&comments=creating  
Azure key 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>2019-04-01T11:49:48Z</DATETIME>  
    <TEXT>Success</TEXT>  
    <ITEM_LIST>  
      <ITEM>  
        <KEY>ID</KEY>
```

```
<VALUE>910196</VALUE>
</ITEM>
</ITEM_LIST>
</RESPONSE>
</SIMPLE_RETURN>
```

Sample - Update Azure Key Vault

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=update&id=910196&type=Azure Key&title=My Azure key
Vault&url=https://azkey.com&app_id=&app123&ssl_verify=0&cert=----
BEGIN CERTIFICATE----
MIIEEjCCArKgAwIBAgIBAjANBgkqhkiG9w0BAQUFADCBljEbMBkGA1UE
AwwSU2Nh-----END CERTIFICATE-----&private_key=key&comments=updating
Azure key 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>2019-03-06T00:36:12Z</DATETIME>
    <TEXT>Success</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>910196</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Sample - View Azure Key Vault

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d
"action=view&id=27016"
"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>
  <RESPONSE>
    <DATETIME>2019-03-15T18:54:21Z</DATETIME>
    <VAULT_QUEST>
      <TITLE>
        <![CDATA[My Azure key Vault]]>
      </TITLE>
      <COMMENTS>
        <![CDATA[creating Azure key vault from api]]>
      </COMMENTS>
      <VAULT_TYPE>
        <![CDATA[Azure Key]]>
      </VAULT_TYPE>
      <CREATED_ON>2019-03-12T21:42:30Z</CREATED_ON>
      <OWNER>quays_rs12</OWNER>
      <LAST_MODIFIED>
        <DATETIME>2019-03-12T21:46:38Z</DATETIME>
        <BY>quays_rs12</BY>
      </LAST_MODIFIED>
      <APP_ID>
        <![CDATA[app12345]]>
      </APP_ID>
      <URL>
        <![CDATA[https://azkey1.com]]>
      </URL>
      <SSL_VERIFY>
        <![CDATA[1]]>
      </SSL_VERIFY>
      <ID>28008</ID>
    </VAULT_QUEST>
  </RESPONSE>
</VAULT_OUTPUT>
```

Updated DTD

<base_url>/api/2.0/fo/vault/vault_view.dtd

The element APP_ID has been added to the DTD.

```
<!-- QUALYS VAULT_VIEW DTD -->
<!ELEMENT VAULT_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)>
...
<!ELEMENT VAULT_QUEST (TITLE, COMMENTS, VAULT_TYPE, CREATED_ON?,
OWNER?, LAST_MODIFIED?, APPID?, APPKEY?, USERNAME?, URL?,
SSL_VERIFY?, DOMAIN?, API_USERNAME?, WEB_USERNAME?,
SERVER_ADDRESS?, PORT?, SAFE?, API_VERSION?, AUTH_TYPE?, PATH?,
ROLE_NAME?, ROLE_ID?, SECRET_ID?, APP_ID?, (UUID|ID))>
<!ELEMENT UUID (#PCDATA)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT COMMENTS (#PCDATA)>
<!ELEMENT VAULT_TYPE (#PCDATA)>
<!ELEMENT CREATED_ON (#PCDATA)>
<!ELEMENT OWNER (#PCDATA)>
<!ELEMENT APPID (#PCDATA)>
<!ELEMENT APPKEY (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
...
<!ELEMENT APP_ID (#PCDATA)>
<!ELEMENT LAST_MODIFIED (DATETIME, BY?)>
<!ELEMENT BY (#PCDATA)>
<!-- EOF -->
```

Authentication Record API

Create, update, list authentication records with Azure key vaults. You can use Azure key vaults with Windows, Unix, Cisco, MS SQL, MySQL, MariaDB, Oracle, MongoDB. For PostgreSQL, private key and passphrase retrieval are supported.

Create/Update Authentication Record

Choose the Azure key vault in your authentication record and provide secret name where your login credentials (secrets) are stored.

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=Azure Key
ips={value}	(Required to create record) The IP address(es) the server will log into using the record's credentials. Multiple entries are comma separated. (Optional to update record) IPs specified will overwrite existing IPs in the record, and existing IPs will be removed.
ak_secret_name={value}	(Optional) The secret name assigned to the secret stored in the vault.

Sample - Create Unix record with Azure key Vault

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d  
"action=create&title=unix-api-pk-  
azure1&username=root&login_type=vault&vault_id=31003&vault_type=Az  
ure Key&ips=20.0.0.2&ak_secret_name=ak-ux-secret-api1"  
"https://qualysapi.qualys.com/api/2.0/fo/auth/unix/"--data-binary  
@create-azure-key-pk.xml
```

create-azure-key-pk.xml

```
<?xml version="1.0" encoding="UTF-8" ?>  
<UNIX_AUTH_PARAMS>  
  <PRIVATE_KEY_CERTIFICATES>  
    <PRIVATE_KEY_CERTIFICATE>
```

```
<PRIVATE_KEY_INFO type="vault">
<DIGITAL_VAULT>
<VAULT_TYPE>Azure Key</VAULT_TYPE>
<VAULT_ID>31005</VAULT_ID>
<AK_SECRET_NAME><![CDATA[pk-secret_name]]></AK_SECRET_NAME>
</DIGITAL_VAULT>
</PRIVATE_KEY_INFO>
<PASSPHRASE_INFO type="vault">
<DIGITAL_VAULT>
<VAULT_USERNAME>root</VAULT_USERNAME>
<VAULT_TYPE>Azure Key</VAULT_TYPE>
<VAULT_ID>31003</VAULT_ID>
<AK_SECRET_NAME><![CDATA[pp-secret_name]]></AK_SECRET_NAME>
</DIGITAL_VAULT>
</PASSPHRASE_INFO>
</PRIVATE_KEY_CERTIFICATE>
</PRIVATE_KEY_CERTIFICATES>
</UNIX_AUTH_PARAMS>
```

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>2019-03-05T00:09:43Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>79027</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Sample - Create MongoDB record with Azure key Vault

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d  
"action=create&title=mongo-api-pk-azure-  
1&username=root&login_type=vault&  
vault_id=31003&vault_type=Azure Key&ips=20.0.0.2&unix_conf_path=  
/etc/mongo.conf&database_name=mongo_adm&ak_secret_name=ak-secret-  
api&port=3434"  
"https://qualysapi.qualys.com/api/2.0/fo/auth/mongodb/"
```

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>2019-03-26T21:51:31Z</DATETIME>  
    <BATCH_LIST>  
      <BATCH>  
        <TEXT>Successfully Created</TEXT>  
        <ID_SET>  
          <ID>79025</ID>  
        </ID_SET>  
      </BATCH>  
    </BATCH_LIST>  
  </RESPONSE>  
</BATCH_RETURN>
```

Sample - List authentication records

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -d  
"action=list&ids=79027"  
"https://qualysapi.qualys.com/api/2.0/fo/auth/unix/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>  
<!DOCTYPE AUTH_UNIX_LIST_OUTPUT SYSTEM  
"https://qualysapi.qualys.com/api/2.0/fo/auth/unix/auth_unix_list_  
output.dtd">
```

```
<AUTH_UNIX_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2019-03-26T22:07:20Z</DATETIME>
    <AUTH_UNIX_LIST>
      <AUTH_UNIX>
        <ID>79027</ID>
        <TITLE>
          <![CDATA[unix-api-pk-azure-1]]>
        </TITLE>
        <USERNAME>
          <![CDATA[root]]>
        </USERNAME>
        <SKIP_PASSWORD>0</SKIP_PASSWORD>
        <CLEARTEXT_PASSWORD>0</CLEARTEXT_PASSWORD>
        <PRIVATE_KEY_CERTIFICATE_LIST>
          <PRIVATE_KEY_CERTIFICATE>
            <ID>15004</ID>
            <PRIVATE_KEY_INFO type="vault">
              <DIGITAL_VAULT>
                <DIGITAL_VAULT_ID>
                  <![CDATA[31005]]>
                </DIGITAL_VAULT_ID>
                <DIGITAL_VAULT_TYPE>
                  <![CDATA[Azure Key]]>
                </DIGITAL_VAULT_TYPE>
                <DIGITAL_VAULT_TITLE>
                  <![CDATA[azkey-api-test1]]>
                </DIGITAL_VAULT_TITLE>
                <VAULT_SECRET_NAME>
                  <![CDATA[pk-secret_name]]>
                </VAULT_SECRET_NAME>
              </DIGITAL_VAULT>
            </PRIVATE_KEY_INFO>
          </PRIVATE_KEY_CERTIFICATE>
          <PASSPHRASE_INFO type="vault">
            <DIGITAL_VAULT>
              <DIGITAL_VAULT_ID>
                <![CDATA[31003]]>
              </DIGITAL_VAULT_ID>
              <DIGITAL_VAULT_TYPE>
                <![CDATA[Azure Key]]>
              </DIGITAL_VAULT_TYPE>
            </DIGITAL_VAULT>
          </PASSPHRASE_INFO>
        </PRIVATE_KEY_CERTIFICATE_LIST>
      </AUTH_UNIX>
    </AUTH_UNIX_LIST>
  </RESPONSE>
</AUTH_UNIX_LIST_OUTPUT>
```



```
<DIGITAL_VAULT_TITLE>
  <![CDATA[az-ui-1]]>
</DIGITAL_VAULT_TITLE>
<VAULT_USERNAME>
  <![CDATA[root]]>
</VAULT_USERNAME>
<VAULT_SECRET_NAME>
  <![CDATA[pp-secret_name]]>
</VAULT_SECRET_NAME>
</DIGITAL_VAULT>
</PASSPHRASE_INFO>
</PRIVATE_KEY_CERTIFICATE>
</PRIVATE_KEY_CERTIFICATE_LIST>
<IP_SET>
  <IP>20.0.0.2</IP>
</IP_SET>
<LOGIN_TYPE>
  <![CDATA[vault]]>
</LOGIN_TYPE>
<DIGITAL_VAULT>
  <DIGITAL_VAULT_ID>
    <![CDATA[31003]]>
  </DIGITAL_VAULT_ID>
  <DIGITAL_VAULT_TYPE>
    <![CDATA[Azure Key]]>
  </DIGITAL_VAULT_TYPE>
  <DIGITAL_VAULT_TITLE>
    <![CDATA[az-ui-1]]>
  </DIGITAL_VAULT_TITLE>
  <VAULT_SECRET_NAME>
    <![CDATA[ak-ux-secret-api1]]>
  </VAULT_SECRET_NAME>
</DIGITAL_VAULT>
<NETWORK_ID>0</NETWORK_ID>
<CREATED>
  <DATETIME>2019-03-26T22:02:11Z</DATETIME>
  <BY>quays_rs12</BY>
</CREATED>
<LAST_MODIFIED>
  <DATETIME>2019-03-26T22:02:11Z</DATETIME>
</LAST_MODIFIED>
```

```
        </AUTH_UNIX>  
    </AUTH_UNIX_LIST>  
</RESPONSE>  
</AUTH_UNIX_LIST_OUTPUT>
```