Qualys Cloud Platform (VM, PC) v8.x

API Release Notes

Version 8.22.2

March 2, 2020 (updated March 11, 2020)

This new version of the Qualys Cloud Platform (VM, PC) includes improvements to the Qualys API. You'll find all the details in our user guides, available at the time of release. Just log in to your Qualys account and go to Help > Resources.

What's New

Qualys API Server URL

The Qualys API URL you should use for API requests depends on the Qualys platform where your account is located.

Click here to identify your Qualys platform and get the API URL

This documentation uses the API server URL for Qualys US Platform 1 (https://qualysapi.qualys.com) in sample API requests. If you're on another platform, please replace this URL with the appropriate server URL for your account.

Support for PostgreSQL Authentication on Windows Hosts

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

The PostgreSQL Authentication API (api/2.0/fo/auth/postgresql/) lets you list, create, update and delete PostgreSQL authentication records. User permissions for this API are the same as other authentication record APIs.

Which technologies are supported?

We've added support for PostgreSQL 9.x, PostgreSQL 10.x, PostgreSQL 11.x and PostgreSQL 12.x authentication for compliance scans on Windows hosts.

List PostgreSQL Records

Supported parameters for PostgreSQL Authentication Record List API call (/api/2.0/fo/auth/postgresql/?action=list) are described in the Qualys API User Guide under List Authentication Records. New element WIN_CONF_FILE is added to authentication record List API.

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_RECORDS_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/postgresql/auth_postgresql_
list_output.dtd">
<AUTH_POSTGRESQL_LIST_OUTPUT>
    <RESPONSE>
        <DATETIME>2020-01-28T11:16:48Z</DATETIME>
        <AUTH_POSTGRESQL>
                <ID>72172</ID>
                <TITLE>
                    <![CDATA[Postgres Sql 12.x]]>
                </TITLE>
                <USERNAME>
                    <![CDATA[qualys_scan]]>
                </USERNAME>
                <DATABASE>
                    <![CDATA[postgres]]>
                </DATABASE>
```

```
                    <PORT>5432</PORT>
                    <SSL_VERIFY>
                        <![CDATA[0]]>
                    </SSL_VERIFY>
                    <IP_SET>
                        <IP>10.115.105.243</IP>
                    </IP_SET>
                    <WIN_CONF_FILE>
                      <![CDATA[C:\Program Files\pgsql\data\postgresql.conf]]>
                    </WIN_CONF_FILE>
                    <UNIX_CONF_FILE>
                        <![CDATA[/var/lib/pgsql/12/data/postgresql.conf]]>
                    </UNIX_CONF_FILE>
                    <NETWORK_ID>0</NETWORK_ID>
                    <CREATED>
                        <DATETIME>2020-01-24T06:09:27Z</DATETIME>
                        <BY>quays_sp1</BY>
                    </CREATED>
                    <LAST_MODIFIED>
                        <DATETIME>2020-01-24T07:36:40Z</DATETIME>
                    </LAST_MODIFIED>
                </AUTH_POSTGRESQL>
            </AUTH_POSTGRESQL_LIST>
        </RESPONSE>
    </AUTH_POSTGRESQL_LIST_OUTPUT>
```

Updated DTD

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

The element WIN_CONF_FILE has been added to identify PostgreSQL records for Windows.

```
<!-- QUALYS AUTH_POSTGRESQL_LIST_OUTPUT DTD -->
<!ELEMENT AUTH_POSTGRESQL_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)>
<!ELEMENT POST_DATA (#PCDATA)>
<!ELEMENT RESPONSE (DATETIME, (AUTH_POSTGRESQL_LIST|ID_SET)?,
WARNING_LIST?, GLOSSARY?)>
<!ELEMENT AUTH_POSTGRESQL_LIST (AUTH_POSTGRESQL+)>
<!ELEMENT AUTH_POSTGRESQL (ID, TITLE, USERNAME, DATABASE, PORT,
SSL_VERIFY, HOSTS?, IP_SET?, LOGIN_TYPE?, DIGITAL_VAULT?, WIN_CONF_FILE?,
UNIX_CONF_FILE?, PRIVATE_KEY_CERTIFICATE_LIST?, NETWORK_ID?, CREATED,
```

```
LAST_MODIFIED, COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT PRIVATE_KEY_CERTIFICATE_LIST (PRIVATE_KEY_CERTIFICATE)*>

<!ELEMENT PRIVATE_KEY_CERTIFICATE (ID, PRIVATE_KEY_INFO, PASSPHRASE_INFO,
CERTIFICATE?)+>
<!ELEMENT PRIVATE_KEY_INFO (PRIVATE_KEY|DIGITAL_VAULT)>
<!ATTLIST PRIVATE_KEY_INFO>
<!-- Private key contents will never be rendered -->
<!ELEMENT PRIVATE_KEY EMPTY>
<!ELEMENT PASSPHRASE_INFO (DIGITAL_VAULT?)>
<!ATTLIST PASSPHRASE_INFO type (basic|vault) "basic">
<!-- Certificate contents will never be rendered -->
<!ELEMENT CERTIFICATE EMPTY>

<!ELEMENT PORT (#PCDATA)>
<!ELEMENT DATABASE (#PCDATA)>
<!ELEMENT SSL_VERIFY (#PCDATA)>
<!ELEMENT HOSTS (HOST+)>
<!ELEMENT HOST (#PCDATA)>
<!ELEMENT IP_SET (IP|IP_RANGE)+>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT IP_RANGE (#PCDATA)>
<!ELEMENT LOGIN_TYPE (#PCDATA)>
<!ELEMENT WIN_CONF_FILE (#PCDATA)>
<!ELEMENT UNIX_CONF_FILE (#PCDATA)>
<!ELEMENT CLIENT_CERT (#PCDATA)>
...
<!ELEMENT VAULT_SECRET_KV_KEY (#PCDATA)>
<!-- EOF -->
```

Create/Update PostgreSQL Authentication Records for Windows

A new optional parameter is added to create or update PostgreSQL authentication records for Windows. You'll also notice that the pgsql_unix_conf_file input parameter is now optional.

Parameter	Description
pgsql_win_conf_path= {value}	The full path to the configuration file (postgresql.conf) on your Windows assets (IP addresses). The file must be in the same location on all assets for this record.

Sample - Create PostgreSQL Record on Windows

API request:

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: curl' -d
"action=create&title=api-windows-postgres&pgsql_win_conf_path=C:\Program
Files\PostgreSQL\11\data\postgresql.conf&pgsql_db_name=postgres&username=
qualys_scan&password=password&ips=10.10.10.35"
"https://qualysapi.qualys.com/api/2.0/fo/auth/postgresql/"
```

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>2020-01-28T10:55:39Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Created</TEXT>
                <ID_SET>
                    <ID>72178</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

Sample - Update PostgreSQL Record on Windows

API request:

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: curl' -d
"action=update&ids=72178&pgsql_win_conf_path=C:\Program
Files\PostgreSQL\11\data\postgresql11.conf"
"https://qualysapi.qualys.com/api/2.0/fo/auth/postgresql/"
```

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>2020-01-28T11:06:36Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Updated</TEXT>
                <ID_SET>
                    <ID>72178</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

New Microsoft SharePoint Authentication API

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

Compliance scans now support Microsoft SharePoint authentication on Windows and Database hosts. The new Microsoft SharePoint Authentication API (api/2.0/fo/auth/microsoft_sharepoint/) lets you list, create, update and delete Microsoft SharePoint records. User permissions for this API are the same as other authentication record APIs. Microsoft SharePoint authentication is supported for Microsoft SharePoint versions 2010, 2013, 2016 and 2019.

List all records

Use the Authentication Record List API (api/2.0/fo/auth with action=list) to list authentication records for all types. You'll see <AUTH_MICROSOFT_SHAREPOINT_IDS> in the output when you have Microsoft SharePoint records in your account.

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_RECORDS_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/auth_records.dtd">
<AUTH_RECORDS_OUTPUT>
    <RESPONSE>
        <DATETIME>2020-02-14T06:40:29Z</DATETIME>
        <AUTH_RECORDS>
            <AUTH_UNIX_IDS>
                <ID_SET>
                    <ID>63215</ID>
                    <ID>63239</ID>
                    <ID>65170</ID>
                    <ID>65172</ID>
                    <ID>66185</ID>
                </ID_SET>
            </AUTH_UNIX_IDS>
```

```
                <AUTH_VMWARE_IDS>
                    <ID_SET>
                        <ID>63213</ID>
                        <ID>63235</ID>
                        <ID>63237</ID>
                        <ID>63241</ID>
                    </ID_SET>
                </AUTH_VMWARE_IDS>
                <AUTH_POSTGRESQL_IDS>
                    <ID_SET>
                        <ID>66387</ID>
                        <ID>66389</ID>
                        <ID>69602</ID>
                        <ID>72224</ID>
                    </ID_SET>
                </AUTH_POSTGRESQL_IDS>
                <AUTH_ORACLE_HTTP_SERVER_IDS>
                    <ID_SET>
                        <ID>66388</ID>
                    </ID_SET>
                </AUTH_ORACLE_HTTP_SERVER_IDS>
                <AUTH_MICROSOFT_SHAREPOINT_IDS>
                    <ID_SET>
                        <ID>72222</ID>
                    </ID_SET>
                </AUTH_MICROSOFT_SHAREPOINT_IDS>
                <AUTH_GREENPLUM_IDS>
                    <ID_SET>
                        <ID_RANGE>66183-66184</ID_RANGE>
                        <ID>66186</ID>
                        <ID>69598</ID>
                        <ID>69601</ID>
                        <ID>72225</ID>
                    </ID_SET>
                </AUTH_GREENPLUM_IDS>
            </AUTH_RECORDS>
        </RESPONSE>
    </AUTH_RECORDS_OUTPUT>
```

Updated DTD

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

The element AUTH_MICROSOFT_SHAREPOINT_IDS has been added to identify Microsoft SharePoint record IDs.

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

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

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

<!ELEMENT AUTH_RECORDS (AUTH_UNIX_IDS?, AUTH_WINDOWS_IDS?,
AUTH_ORACLE_IDS?, AUTH_ORACLE_LISTENER_IDS?, AUTH_SNMP_IDS?,
AUTH_MS_SQL_IDS?, AUTH_IBM_DB2_IDS?, AUTH_VMWARE_IDS?, AUTH_MS_IIS_IDS?,
AUTH_APACHE_IDS?, AUTH_IBM_WEBSPHERE_IDS?, AUTH_HTTP_IDS?,
AUTH_SYBASE_IDS?, AUTH_MYSQL_IDS?, AUTH_TOMCAT_IDS?,
AUTH_ORACLE_WEBLOGIC_IDS?, AUTH_DOCKER_IDS?, AUTH_POSTGRESQL_IDS?,
AUTH_MONGODB_IDS?, AUTH_PALO_ALTO_FIREWALL_IDS?, AUTH_VCENTER_IDS?,
AUTH_JBOSS_IDS?, AUTH_MARIADB_IDS?, AUTH_INFORMIXDB_IDS?,
AUTH_MS_EXCHANGE_IDS?, AUTH_ORACLE_HTTP_SERVER_IDS?, AUTH_GREENPLUM_IDS?,
AUTH_MICROSOFT_SHAREPOINT_IDS? )>
...
<!ELEMENT AUTH_MARIADB_IDS (ID_SET)>
<!ELEMENT AUTH_INFORMIXDB_IDS (ID_SET)>
<!ELEMENT AUTH_MS_EXCHANGE_IDS (ID_SET)>
<!ELEMENT AUTH_ORACLE_HTTP_SERVER_IDS (ID_SET)>
<!ELEMENT AUTH_GREENPLUM_IDS (ID_SET)>
<!ELEMENT AUTH_MICROSOFT_SHAREPOINT_IDS (ID_SET)>
...
```

List Microsoft SharePoint records

Use these parameters to list Microsoft SharePoint authentication records.

Parameter	Description
action={action}	(Required) Specify list (using GET or POST) to list records.
details={value}	(Optional) Default value is Basic. You can choose from None, Basic and All.
ids={value}	(Optional) Microsoft SharePoint auth record IDs to list. Specify record IDs and/or ID ranges (for example, 1359-1407). Multiple entries are comma separated.

Sample - List Microsoft SharePoint Records with Basic Details

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/microsoft_sharepoint/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_MICROSOFT_SHAREPOINT_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/microsoft_sharepoint/auth_m
icrosoft_sharepoint_list_output.dtd">
    <AUTH_MICROSOFT_SHAREPOINT_LIST>
      <AUTH_MICROSOFT_SHAREPOINT>
        <ID>2372474</ID>
        <TITLE><![CDATA[SharePoint_WindowsAuth]]></TITLE>
        <USERNAME><![CDATA[username]]></USERNAME>
        <IP_SET>
          <IP>10.10.10.13</IP>
        </IP_SET>
        <MSSQL>
          <DB_LOCAL><![CDATA[0]]></DB_LOCAL>
          <WINDOWS_DOMAIN><![CDATA[sample.qualys.com]]></WINDOWS_DOMAIN>
          <KERBEROS><![CDATA[1]]></KERBEROS>
          <NTLMV2><![CDATA[1]]></NTLMV2>
        </MSSQL>
        <LOGIN_TYPE><![CDATA[basic]]></LOGIN_TYPE>
        <CREATED>
          <DATETIME>2020-03-10T18:47:26Z</DATETIME>
          <BY>joe_user</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2020-03-10T18:47:26Z</DATETIME>
        </LAST_MODIFIED>
```

```
    </AUTH_MICROSOFT_SHAREPOINT>
    <AUTH_MICROSOFT_SHAREPOINT>
      <ID>2372483</ID>
      <TITLE><![CDATA[SharePoint_DatabaseAuth]]></TITLE>
      <USERNAME><![CDATA[username]]></USERNAME>
      <IP_SET>
        <IP_RANGE>10.10.10.19-10.10.10.20</IP_RANGE>
      </IP_SET>
      <MSSQL>
        <DB_LOCAL><![CDATA[1]]></DB_LOCAL>
        <KERBEROS><![CDATA[1]]></KERBEROS>
        <NTLMV2><![CDATA[1]]></NTLMV2>
        <NTLMV1><![CDATA[1]]></NTLMV1>
      </MSSQL>
      <LOGIN_TYPE><![CDATA[basic]]></LOGIN_TYPE>
      <CREATED>
        <DATETIME>2020-03-10T20:53:37Z</DATETIME>
        <BY>joe_user</BY>
      </CREATED>
      <LAST_MODIFIED>
        <DATETIME>2020-03-10T20:53:37Z</DATETIME>
      </LAST_MODIFIED>
    </AUTH_MICROSOFT_SHAREPOINT>
```

Sample - List Microsoft SharePoint Records with All Details

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_MICROSOFT_SHAREPOINT_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/microsoft_sharepoint/auth_m
icrosoft_sharepoint_list_output.dtd">
<AUTH_MICROSOFT_SHAREPOINT_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2020-03-11T22:56:20Z</DATETIME>
    <AUTH_MICROSOFT_SHAREPOINT_LIST>
      <AUTH_MICROSOFT_SHAREPOINT>
        <ID>2372474</ID>
        <TITLE><![CDATA[SharePoint_WindowsAuth]]></TITLE>
        <USERNAME><![CDATA[username]]></USERNAME>
        <IP_SET>
          <IP>10.10.10.13</IP>
        </IP_SET>
```

```
    <MSSQL>
      <DB_LOCAL><![CDATA[0]]></DB_LOCAL>
      <WINDOWS_DOMAIN><![CDATA[sample.qualys.com]]></WINDOWS_DOMAIN>
      <KERBEROS><![CDATA[1]]></KERBEROS>
      <NTLMV2><![CDATA[1]]></NTLMV2>
    </MSSQL>
    <LOGIN_TYPE><![CDATA[basic]]></LOGIN_TYPE>
    <CREATED>
      <DATETIME>2020-03-10T18:47:26Z</DATETIME>
      <BY>joe_user</BY>
    </CREATED>
    <LAST_MODIFIED>
      <DATETIME>2020-03-10T18:47:26Z</DATETIME>
    </LAST_MODIFIED>
  </AUTH_MICROSOFT_SHAREPOINT>
  <AUTH_MICROSOFT_SHAREPOINT>
    <ID>2372483</ID>
    <TITLE><![CDATA[SharePoint_DatabaseAuth]]></TITLE>
    <USERNAME><![CDATA[username]]></USERNAME>
    <IP_SET>
      <IP_RANGE>10.10.10.19-10.10.10.20</IP_RANGE>
    </IP_SET>
    <MSSQL>
      <DB_LOCAL><![CDATA[1]]></DB_LOCAL>
      <KERBEROS><![CDATA[1]]></KERBEROS>
      <NTLMV2><![CDATA[1]]></NTLMV2>
      <NTLMV1><![CDATA[1]]></NTLMV1>
    </MSSQL>
    <LOGIN_TYPE><![CDATA[basic]]></LOGIN_TYPE>
    <CREATED>
      <DATETIME>2020-03-10T20:53:37Z</DATETIME>
      <BY>joe_user</BY>
    </CREATED>
    <LAST_MODIFIED>
      <DATETIME>2020-03-10T20:53:37Z</DATETIME>
    </LAST_MODIFIED>
  </AUTH_MICROSOFT_SHAREPOINT>
  <AUTH_MICROSOFT_SHAREPOINT>
    <ID>2372484</ID>
    <TITLE><![CDATA[SharePoint123]]></TITLE>
    <USERNAME><![CDATA[userupdate]]></USERNAME>
    <IP_SET>
      <IP_RANGE>10.10.10.25-10.10.10.26</IP_RANGE>
    </IP_SET>
    <MSSQL>
      <DB_LOCAL><![CDATA[0]]></DB_LOCAL>
      <WINDOWS_DOMAIN><![CDATA[sample2.qualys.com]]></WINDOWS_DOMAIN>
      <KERBEROS><![CDATA[1]]></KERBEROS>
      <NTLMV1><![CDATA[1]]></NTLMV1>
```

```
          </MSSQL>
          <LOGIN_TYPE><![CDATA[basic]]></LOGIN_TYPE>
          <CREATED>
            <DATETIME>2020-03-10T20:55:50Z</DATETIME>
            <BY>joe_user</BY>
          </CREATED>
          <LAST_MODIFIED>
            <DATETIME>2020-03-11T16:19:19Z</DATETIME>
          </LAST_MODIFIED>
        </AUTH_MICROSOFT_SHAREPOINT>
      </AUTH_MICROSOFT_SHAREPOINT_LIST>
      <GLOSSARY>
        <USER_LIST>
          <USER>
            <USER_LOGIN>joe_user</USER_LOGIN>
            <FIRST_NAME>Joe</FIRST_NAME>
            <LAST_NAME>User</LAST_NAME>
          </USER>
        </USER_LIST>
      </GLOSSARY>
    </RESPONSE>
</AUTH_MICROSOFT_SHAREPOINT_LIST_OUTPUT>
```

New DTD

<base_url>/api/2.0/fo/auth/microsoft_sharepoint/auth_microsoft_sharepoint_list_output.dtd

```
<!-- QUALYS AUTH_MICROSOFT_SHAREPOINT_LIST_OUTPUT DTD -->
<!ELEMENT AUTH_MICROSOFT_SHAREPOINT_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)>
<!ELEMENT POST_DATA (#PCDATA)>
<!ELEMENT RESPONSE (DATETIME, (AUTH_MICROSOFT_SHAREPOINT_LIST|ID_SET)?,
WARNING_LIST?, GLOSSARY?)>
<!ELEMENT AUTH_MICROSOFT_SHAREPOINT_LIST (AUTH_MICROSOFT_SHAREPOINT+)>
<!ELEMENT AUTH_MICROSOFT_SHAREPOINT (ID, TITLE, USERNAME?, IP_SET?,
MSSQL?, LOGIN_TYPE?, DIGITAL_VAULT?, NETWORK_ID?, CREATED, LAST_MODIFIED,
COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
```

```
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT MSSQL (DB_LOCAL?, WINDOWS_DOMAIN?, KERBEROS?, NTLMV2?, NTLMV1?)>
<!ELEMENT DB_LOCAL (#PCDATA)>
<!ELEMENT WINDOWS_DOMAIN (#PCDATA)>
<!ELEMENT KERBEROS (#PCDATA)>
<!ELEMENT NTLMV2 (#PCDATA)>
<!ELEMENT NTLMV1 (#PCDATA)>
<!ELEMENT HOSTS (HOST+)>
<!ELEMENT HOST (#PCDATA)>
<!ELEMENT IP_SET (IP|IP_RANGE)+>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT IP_RANGE (#PCDATA)>

<!ELEMENT LOGIN_TYPE (#PCDATA)>

<!ELEMENT NETWORK_ID (#PCDATA)>

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

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

<!ELEMENT GLOSSARY (USER_LIST?)>
<!ELEMENT USER_LIST (USER+)>
<!ELEMENT USER (USER_LOGIN, FIRST_NAME, LAST_NAME)>
<!ELEMENT FIRST_NAME (#PCDATA)>
<!ELEMENT LAST_NAME (#PCDATA)>
<!ELEMENT DIGITAL_VAULT (DIGITAL_VAULT_ID, DIGITAL_VAULT_TYPE,
DIGITAL_VAULT_TITLE, VAULT_FOLDER?, VAULT_FILE?, VAULT_SECRET_NAME?,
VAULT_SYSTEM_NAME?, VAULT_ACCOUNT_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_ACCOUNT_NAME (#PCDATA)>
<!-- EOF -->
```

Create/Update Microsoft SharePoint Authentication Records

Use these parameters to create or update Microsoft SharePoint authentication records.

Parameter	Description
action={action}	(Required) Specify create, update, delete (using POST) or list (using GET or POST).
echo_request={**0**|1}	(Optional) Specify 1 to view (echo) input parameters in the XML output. By default these are not included.
ids={value}	(Required to update or delete record) Record IDs to update/delete. Specify record IDs and/or ID ranges (for example, 1359-1407). Multiple entries are comma separated.
title={value}	(Required to create record) A title for the record. The title must be unique. Maximum 255 characters (ascii).
comments={value}	(Optional to create or update record) User defined comments. Maximum of 1999 characters.
Microsoft SharePoint	
db_local={0|1}	(Optional to create or update record) Set to 1 when login credentials are for a MS SQL Server database account. Set to 0 when login credentials are for a Microsoft Windows operating system account that is associated with a MS SQL Server database account. When db_local is not specified during a create request, the flag is set to 1.
windows_domain={value}	(Required when db_local=0, otherwise invalid) The domain name where the login credentials are stored when the login credentials are for a Microsoft Windows operating system account that is associated with a MS SQL Server database account. The domain name may include 1-256 characters (ascii). For an update request when the credentials for the record are for a Microsoft Windows account (db_local=0) and you want to change the record to use credentials for a MS SQL Server account (db_local=1), then you must set windows_domain='' (the empty string) to clear the current parameter setting.
kerberos={0|1}	(Optional to create or update record) When not specified, Kerberos is enabled allowing the scanning engine to try Kerberos when negotiating authentication to target hosts. Specify kerberos=0 if you do not want Kerberos attempted.
ntlmv2={0|1}	(Optional to create or update record) When not specified, NTLMv2 is enabled allowing the scanning engine to try NTLMv2 when negotiating authentication to target hosts. Specify ntlmv2=0 if you do not want NTLMv2 attempted.

Parameter	Description
ntlmv1={0|1}	(Optional to create or update record) When not specified, NTLMv1 will not be attempted. Specify ntlmv1=1 to try NTMLv1 when negotiating authentication to target hosts.
Login credentials	
username={value}	(Required for create request) The username of the account to be used for authentication. If password is specified this is the username of a MS SQL Server database user account used for SharePoint. If login_type=vault is specified, this is the username of a vault account. Maximum 255 characters (ascii).
password={value}	(For create request, password or login_type=vault is required) The password of the MS SQL Server database user account to be used for authentication. Maximum 100 characters (ascii).
login_type={value}	(For create request, password or login_type=vault is required) Login type can be basic (default) or vault. Set to vault if a third party vault will be used to retrieve the password. Vault parameters need to be provided in the record. See "Vault Definition" in the API user guide.
vault_id={value}	(Required if login_type=vault) The ID of the vault to be used to retrieve the password for login.
vault_type={value}	(Required if login_type=vault) The third party vault to be used to retrieve the password for login. Certain vaults support this capability. See "Vault Support Matrix" in the API user guide.
secret_name={value}	(Required if vault type is Thycotic Secret Server) Specify the secret name that contains the password to be used for authentication. The scanning engine will perform a search for the secret name and then get the password from the secret returned by the search. A single exact match of the secret name must be found in order for authentication to be successful. The secret name may contain a maximum of 256 characters, and must not contain multibyte characters.
system_name={value}	(Optional if vault type is BeyondTrust PBPS or Quest Vault) The managed system name (also known as asset name). When not specified, we'll attempt to auto-discover the system name at scan time.
account_name={value}	(Optional if vault type is BeyondTrust PBPS) The account name. When not specified, we'll try the username specified in the authentication record.

Parameter	Description
folder={value}	(Required if vault type is CyberArk AIM and Cyber-ARK PIM Suite) Specify the name of the folder in the secure digital safe where the password to be used for authentication should be stored. The folder name can contain a maximum of 169 characters. Entering a trailing /, as in folder/, is optional (when specified, the service removes the trailing / and does not save it in the folder name). The maximum length of a folder name with a file name is 170 characters (the leading and/or trailing space in the input value will be removed). These special characters cannot be included in a folder name: / : * ? " < > | <tab>
file={value}	(Required if vault type is CyberArk AIM and Cyber-ARK PIM Suite) Specify the name of the file in the secure digital safe where the password to be used for authentication should be stored. The file name can contain a maximum of 165 characters. The maximum length of a folder name plus a file name is 170 characters (the leading and/or trailing space in the input value will be removed). These special characters cannot be included in a file name: \ / : * ? " < > | <tab>
Target Hosts	
ips={value}	(Required to create record) The IP address(es) for the Microsoft SharePoint targets you want to authenticate to. Multiple entries are comma separated. (Optional to update record) IPs specified will overwrite existing IPs in the record, and existing IPs will be removed. This parameter and the add_ips parameter or the remove_ips parameter cannot be specified in the same request.
add_ips={value}	(Optional to update record) Add IPs and/or ranges to the IPs list for this record. Multiple IPs/ranges are comma separated. This parameter and the ips parameter cannot be specified in the same request.
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.

Parameter	Description
network_id={value}	(Optional to create or update record, and valid only when the networks feature is enabled) The network ID for the record.

Sample - Create Microsoft SharePoint Record

API request with Microsoft Windows login (db_local=0):

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: Curl' -d
"action=create&title=SharePoint&ips=10.10.10.13&username=username&passwor
d=password&db_local=0&windows_domain=sample.qualys.com"
"https://qualysapi.qualys.com/api/2.0/fo/auth/microsoft_sharepoint/"
```

API request with MS SQL Server database login (db_local=1):

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: Curl' -d
"action=create&title=SharePoint_withDatabase&ips=10.10.10.14&username=use
rname&password=password&db_local=1"
"https://qualysapi.qualys.com/api/2.0/fo/auth/microsoft_sharepoint/"
```

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>2020-02-13T07:31:33Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Created</TEXT>
                <ID_SET>
                    <ID>72223</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

Sample - Update Microsoft SharePoint Record

API request to update basic information:

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: Curl' -d
"action=update&ids=10002&title=SharePoint2&username=newuser&password=newp
assword&comments=auth-updated"
"https://qualysapi.qualys.com/api/2.0/fo/auth/microsoft_sharepoint/"
```

API request to update vault login and change to different vault:

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: Curl' -d
"action=update&ids=10003&login_type=vault&vault_type=Thycotic+Secret+Serv
er&vault_id=123&secret_name=secret-name"
"https://qualysapi.qualys.com/api/2.0/fo/auth/microsoft_sharepoint/"
```

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>2020-02-13T07:39:09Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Updated</TEXT>
                <ID_SET>
                    <ID>72223</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

Delete Microsoft SharePoint Records

Use the following input parameter to delete one or more Microsoft SharePoint authentication records.

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

Sample - Delete Microsoft SharePoint Records

API request for deleting single record:

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: Curl' -d
"action=delete&ids=10000"
"https://qualysapi.qualys.com/api/2.0/fo/auth/microsoft_sharepoint/"
```

API request for deleting multiple records:

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: Curl' -d
"action=list&ids=10000,10001"
"https://qualysapi.qualys.com/api/2.0/fo/auth/microsoft_sharepoint/"
```

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>2020-02-13T07:40:06Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Deleted</TEXT>
                <ID_SET>
                    <ID>72223</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

New Pivotal Greenplum Authentication API

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

Pivotal Greenplum authentication is now supported for compliance scans on Unix hosts. The new Greenplum Authentication API (api/2.0/fo/auth/greenplum/) lets you list, create, update and delete Greenplum authentication records. User permissions for this API are the same as other authentication record APIs. Authentication is supported for Greenplum versions 5.x and 6.x.

List all record types

Use the Authentication Record List API (/api/2.0/fo/auth/ with action=list) to list authentication records for all types. You'll see <AUTH_GREENPLUM_IDS> in the output when you have Pivotal Greenplum records in your account.

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_RECORDS_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/auth_records.dtd">
<AUTH_RECORDS_OUTPUT>
  <RESPONSE>
    <DATETIME>2019-10-04T09:24:19Z</DATETIME>
    <AUTH_RECORDS>
      <AUTH_UNIX_IDS>
        <ID_SET>
          <ID>1029116</ID>
          <ID>1296290</ID>
          <ID_RANGE>1375563-1375564</ID_RANGE>
          <ID>1505926</ID>
        </ID_SET>
      </AUTH_UNIX_IDS>
      <AUTH_GREENPLUM_IDS>
        <ID_SET>
          <ID>1505929</ID>
```

```
        </ID_SET>
      </AUTH_GREENPLUM_IDS>
    </AUTH_RECORDS>
  </RESPONSE>
</AUTH_RECORDS_OUTPUT>
```

Updated DTD

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

The element AUTH_GREENPLUM_IDS has been added to identify Greenplum record IDs.

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

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

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

<!ELEMENT AUTH_RECORDS (AUTH_UNIX_IDS?, AUTH_WINDOWS_IDS?,
AUTH_ORACLE_IDS?, AUTH_ORACLE_LISTENER_IDS?, AUTH_SNMP_IDS?,
AUTH_MS_SQL_IDS?, AUTH_IBM_DB2_IDS?, AUTH_VMWARE_IDS?, AUTH_MS_IIS_IDS?,
AUTH_APACHE_IDS?, AUTH_IBM_WEBSPHERE_IDS?, AUTH_HTTP_IDS?,
AUTH_SYBASE_IDS?, AUTH_MYSQL_IDS?, AUTH_TOMCAT_IDS?,
AUTH_ORACLE_WEBLOGIC_IDS?, AUTH_DOCKER_IDS?, AUTH_POSTGRESQL_IDS?,
AUTH_MONGODB_IDS?, AUTH_PALO_ALTO_FIREWALL_IDS?, AUTH_VCENTER_IDS?,
AUTH_JBOSS_IDS?, AUTH_MARIADB_IDS?, AUTH_INFORMIXDB_IDS?,
AUTH_MS_EXCHANGE_IDS?, AUTH_ORACLE_HTTP_SERVER_IDS?, AUTH_GREENPLUM_IDS?,
AUTH_MICROSOFT_SHAREPOINT_IDS? )>
...
<!ELEMENT AUTH_INFORMIXDB_IDS (ID_SET)>
<!ELEMENT AUTH_MS_EXCHANGE_IDS (ID_SET)>
<!ELEMENT AUTH_ORACLE_HTTP_SERVER_IDS (ID_SET)>
<!ELEMENT AUTH_GREENPLUM_IDS (ID_SET)>
<!ELEMENT AUTH_MICROSOFT_SHAREPOINT_IDS (ID_SET)>
...
```

List Greenplum records

Use these parameters to list Greenplum authentication records.

Parameter	Description
action={action}	(Required) Specify list (using GET or POST) to list records.
details={value}	(Optional) Default value is Basic. You can choose from None, Basic, and All.
ids={value}	(Optional) Greenplum auth record IDs to list. Specify record IDs and/or ID ranges (for example, 1359-1407). Multiple entries are comma separated.

Sample - List Greenplum Records with All Details

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_GREENPLUM_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/greenplum/auth_greenplum_li
st_output.dtd">
<AUTH_GREENPLUM_LIST_OUTPUT>
    <RESPONSE>
        <DATETIME>2020-01-05T11:41:28Z</DATETIME>
        <AUTH_GREENPLUM_LIST>
            <AUTH_GREENPLUM>
                <ID>66186</ID>
                <TITLE>
                    <![CDATA[greenplum auth]]>
                </TITLE>
                <USERNAME>
                    <![CDATA[root]]>
                </USERNAME>
                <DATABASE>
                    <![CDATA[postgres]]>
                </DATABASE>
                <PORT>5432</PORT>
                <SSL_VERIFY>
                    <![CDATA[0]]>
                </SSL_VERIFY>
                <IP_SET>
                    <IP>10.20.32.111</IP>
                </IP_SET>
```

```
                <UNIX_CONF_FILE>
                    <![CDATA[/usr/local/greenplum-db/master/gpseg-
1/postgresql.conf]]>
                </UNIX_CONF_FILE>
                <NETWORK_ID>0</NETWORK_ID>
                <CREATED>
                    <DATETIME>2019-12-31T10:51:10Z</DATETIME>
                    <BY>qualys_jd</BY>
                </CREATED>
                <LAST_MODIFIED>
                    <DATETIME>2019-12-31T10:51:10Z</DATETIME>
                </LAST_MODIFIED>
            </AUTH_GREENPLUM>
            <AUTH_GREENPLUM>
                <ID>66390</ID>
                <TITLE>
                    <![CDATA[my greenplum record]]>
                </TITLE>
                <USERNAME>
                    <![CDATA[root]]>
                </USERNAME>
                <DATABASE>
                    <![CDATA[postgres]]>
                </DATABASE>
                <PORT>5432</PORT>
                <SSL_VERIFY>
                    <![CDATA[0]]>
                </SSL_VERIFY>
                <IP_SET>
                    <IP>10.10.10.1</IP>
                </IP_SET>
                <UNIX_CONF_FILE>
                    <![CDATA[ /var/lib/pgsql/data/postgresql.conf]]>
                </UNIX_CONF_FILE>
                <NETWORK_ID>0</NETWORK_ID>
                <CREATED>
                    <DATETIME>2020-01-05T09:14:54Z</DATETIME>
                    <BY>qualys_jd</BY>
                </CREATED>
                <LAST_MODIFIED>
                    <DATETIME>2020-01-05T09:14:54Z</DATETIME>
                </LAST_MODIFIED>
            </AUTH_GREENPLUM>
        </AUTH_GREENPLUM_LIST>
        <GLOSSARY>
            <USER_LIST>
                <USER>
                    <USER_LOGIN>qualys_jd</USER_LOGIN>
                    <FIRST_NAME>John</FIRST_NAME>
```

```
                        <LAST_NAME>Doe</LAST_NAME>
                    </USER>
                </USER_LIST>
            </GLOSSARY>
        </RESPONSE>
</AUTH_GREENPLUM_LIST_OUTPUT>
```

New DTD

<base_url>/api/2.0/fo/auth/greenplum/auth_greenplum_list_output.dtd

```
<!-- QUALYS AUTH_GREENPLUM_LIST_OUTPUT DTD -->
<!ELEMENT AUTH_GREENPLUM_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)>
<!ELEMENT POST_DATA (#PCDATA)>
<!ELEMENT RESPONSE (DATETIME, (AUTH_GREENPLUM_LIST|ID_SET)?,
WARNING_LIST?, GLOSSARY?)>
<!ELEMENT AUTH_GREENPLUM_LIST (AUTH_GREENPLUM+)>
<!ELEMENT AUTH_GREENPLUM (ID, TITLE, USERNAME, DATABASE, PORT, SSL_VERIFY,
HOSTS?, IP_SET?, LOGIN_TYPE?, DIGITAL_VAULT?, UNIX_CONF_FILE,
PRIVATE_KEY_CERTIFICATE_LIST?, NETWORK_ID?, CREATED, LAST_MODIFIED,
COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT PRIVATE_KEY_CERTIFICATE_LIST (PRIVATE_KEY_CERTIFICATE)*>

<!ELEMENT PRIVATE_KEY_CERTIFICATE (ID, PRIVATE_KEY_INFO, PASSPHRASE_INFO,
CERTIFICATE?)+>
<!ELEMENT PRIVATE_KEY_INFO (PRIVATE_KEY|DIGITAL_VAULT)>
<!ATTLIST PRIVATE_KEY_INFO type (basic|vault) "basic">
<!-- Private key contents will never be rendered -->
<!ELEMENT PRIVATE_KEY EMPTY>
<!ELEMENT PASSPHRASE_INFO (DIGITAL_VAULT?)>
<!ATTLIST PASSPHRASE_INFO type (basic|vault) "basic">
<!-- Certificate contents will never be rendered -->
<!ELEMENT CERTIFICATE EMPTY>

<!ELEMENT PORT (#PCDATA)>
<!ELEMENT DATABASE (#PCDATA)>
```

```
<!ELEMENT SSL_VERIFY (#PCDATA)>
<!ELEMENT HOSTS (HOST+)>
<!ELEMENT HOST (#PCDATA)>
<!ELEMENT IP_SET (IP|IP_RANGE)+>
<!ELEMENT IP (#PCDATA)>
<!ELEMENT IP_RANGE (#PCDATA)>
<!ELEMENT LOGIN_TYPE (#PCDATA)>
<!ELEMENT UNIX_CONF_FILE (#PCDATA)>
<!ELEMENT CLIENT_CERT (#PCDATA)>
<!ELEMENT CLIENT_KEY (#PCDATA)>
<!ELEMENT CERT_PASSPHASE (#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)>
<!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_SECRET_KV_PATH?, VAULT_SECRET_KV_NAME?,
VAULT_SECRET_KV_KEY?)>
<!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_SECRET_KV_PATH (#PCDATA)>
<!ELEMENT VAULT_SECRET_KV_NAME (#PCDATA)>
<!ELEMENT VAULT_SECRET_KV_KEY (#PCDATA)>
<!-- EOF -->
```

Create/Update Greenplum Authentication Records

Use these parameters to create or update Greenplum authentication records.

Parameter	Description
action={action}	(Required) Specify create, update, delete (using POST) or list (using GET or POST).
echo_request={**0**|1}	(Optional) Specify 1 to view (echo) input parameters in the XML output. By default these are not included.
ids={value}	(Required to update or delete record) Record IDs to update/delete. Specify record IDs and/or ID ranges (for example, 1359-1407). Multiple entries are comma separated.
title={value}	(Required to create record) A title for the record. The title must be unique. Maximum 255 characters (ascii).
comments={value}	(Optional to create or update record) User defined comments. Maximum of 1999 characters.
Greenplum	
greenplum_unix_conf_file={value}	(Required for create request) The full path to the configuration file (postgresql.conf) on your Unix assets (IP addresses). The file must be in the same location on all assets for this record.
greenplum_db_name={value}	(Required for create request) The database instance you want to authenticate to.
port={value}	(Optional) The port where the database instance is running. Default is 5432.
ssl_verify={**0**|1}	(Optional) SSL verification is skipped by default. Set to 1 if you want to verify the server's certificate is valid and trusted.
hosts={value}	(Required if ssl_verify=1) A list of FQDNs for all host IP addresses on which a custom SSL certificate signed by a trusted root CA is installed.
Login credentials	
username={value}	(Required for create request) The username of the account to be used for authentication. If password is specified this is the username of a Greenplum account. If login_type=vault is specified, this is the username of a vault account. Maximum 255 characters (ascii).
password={value}	(For create request, password or login_type=vault is required) The password of the Greenplum account to be used for authentication. Maximum 100 characters (ascii).

Parameter	Description
login_type={value}	(For create request, password or login_type=vault is required) Login type can be basic (default) or vault. Set to vault if a third party vault will be used to retrieve the password. Vault parameters need to be provided in the record. See "Vault Definition" in the API user guide.
vault_id={value}	(Required if login_type=vault) The ID of the vault to be used to retrieve the password for login.
vault_type={value}	(Required if login_type=vault) The third party vault to be used to retrieve the password for login. Certain vaults support this capability. See "Vault Support Matrix" in the API user guide.
Keys, Passphrase	
client_key_type={value}	(Optional) Client key type basic (default) or vault.
client_key={value}	(Optional if client_key_type=basic) Client key content, if private key not in vault.
client_key_vault_type={value}	(Required if client_key_type=vault) The third party vault to be used to retrieve the private key. Certain vaults support this capability. See "Vault Support Matrix" in the API user guide.
client_key_vault_id={value}	(Required if client_key_type=vault) The ID of the vault to get the private key from.
passphrase_type={value}	(Optional) Passphrase type can be basic (default) or vault.
passphrase={value}	(Optional if passphrase_type=basic) The passphrase value.
client_cert={value}	(Optional if passphrase_type=basic) The passphrase certificate content.
passphrase_vault_type={value}	(Required if passphrase_type=vault) The vault where the private key passphrase is stored. For example CA Access Control, CyberArk AIM, Thycotic Secret Server.
passphrase_vault_id={value}	(Required if passphrase_type=vault) The ID of the vault to get the passphrase from.
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. This parameter and the add_ips parameter or the remove_ips parameter cannot be specified in the same request.

Parameter	Description
add_ips={value}	(Optional to update record) Add IPs and/or ranges to the IPs list for this record. Multiple IPs/ranges are comma separated. This parameter and the ips parameter cannot be specified in the same request.
remove_ips={value}	(Optional to update record) IPs to be removed from your record. You may enter a combination of IPs and ranges. Multiple entries are comma separated. This parameter and the ips parameter cannot be specified in the same request.
network_id={value}	(Optional to create or update record, and valid only when the networks feature is enabled) The network ID for the record.

Sample - Create Greenplum Record

API request:

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: curl' -d
"action=create&title=my greenplum
record&ips=10.10.10.1&username=root&password=root&greenplum_db_name=postg
res&port=5421&greenplum_unix_conf_path=/tmp/postgresql.conf"
"https://qualysapi.qualys.com/api/2.0/fo/auth/greenplum/"
```

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>2020-01-05T12:04:32Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Created</TEXT>
                <ID_SET>
                    <ID>66391</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

Sample - Update Greenplum Record

API request:

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: curl' -d
"action=update&ids=66391&title=my greenplum record&comments=new comment"
"https://qualysapi.qualys.com/api/2.0/fo/auth/greenplum/"
```

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>2020-01-05T12:09:25Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Updated</TEXT>
                <ID_SET>
                    <ID>66391</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

Delete Greenplum Records

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

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

Sample - Delete Greenplum Records

API request:

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: curl' -d
"action=delete&ids=66391"
"https://qualysapi.qualys.com/api/2.0/fo/auth/greenplum/"
```

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>2020-01-05T12:10:16Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Deleted</TEXT>
                <ID_SET>
                    <ID>66391</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```