Qualys API Release Notes

Version 8.8

Qualys 8.8 includes improvements to the Qualys API, giving you more ways to integrate your programs and API calls with Qualys Vulnerability Management (VM) and Qualys Policy Compliance (PC). Looking for our API user guides? Just log in to your Qualys account and go to Help > Resources.

What's New

Tell me about the base URL Our documentation and sample code use the API server URL for Qualys US Platform 1. Do you have another base URL? If yes please use it instead.

Account Login	Base URL
Qualys US Platform 1	https://qualysapi.qualys.com
Qualys US Platform 2	https://qualysapi.qg2.apps.qualys.com
Qualys EU Platform	https://qualysapi.qualys.eu
Qualys Private Cloud Platform	https://qualysapi.<customer_base_url>

Authentication API - Assign Vault Info to Records

You can now add or update vault settings like username, vault type, vault title, folder, file for Unix and Windows authentication records.

List Authentication Record

By default, all the authentication records with vault information (if any) are listed.

API request (Windows):

```
curl -X -u "USERNAME:PASSWORD" GET -H "Authorization: Basic
cXVhc3lfdGI6cWF0ZW1w" -H "X-Requested-With: test"
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/?action=list&detail
s=All&ids=30623203"
```

XML output (Windows):

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_WINDOWS_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/auth_windows_list_o
utput.dtd">
<AUTH_WINDOWS_LIST_OUTPUT>
    <RESPONSE>
        <DATETIME>2016-04-21T13:23:48Z</DATETIME>
        <AUTH_WINDOWS_LIST>
            <AUTH_WINDOWS>
                <ID>30623203</ID>
                <TITLE>
                    <![CDATA[Windows Quest1]]>
                </TITLE>
                <USERNAME>
                    <![CDATA[Qualys]]>
                </USERNAME>
                <NTLM>1</NTLM>
                <IP_SET>
                    <IP>10.113.195.151</IP>
                </IP_SET>
                <LOGIN_TYPE>
                    <![CDATA[vault]]>
                </LOGIN_TYPE>
                <DIGITAL_VAULT>
                    <DIGITAL_VAULT_ID>
                        <![CDATA[10873203]]>
                    </DIGITAL_VAULT_ID>
                    <DIGITAL_VAULT_TYPE>
                        <![CDATA[Lieberman ERPM]]>
                    </DIGITAL_VAULT_TYPE>
```

```
                    <DIGITAL_VAULT_TITLE>
                        <![CDATA[Lieberman]]>
                    </DIGITAL_VAULT_TITLE>
                    <VAULT_SYSTEM_NAME>
                        <![CDATA[Auto Discovery]]>
                    </VAULT_SYSTEM_NAME>
                    <VAULT_NS_TYPE>
                        <![CDATA[custom]]>
                    </VAULT_NS_TYPE>
                    <VAULT_NS_NAME>
                        <![CDATA[custom name]]>
                    </VAULT_NS_NAME>
                </DIGITAL_VAULT>
                 <CREATED>
                     <DATETIME>2016-04-21T13:21:42Z</DATETIME>
                     <BY>mark_t</BY>
                 </CREATED>
                 <LAST_MODIFIED>
                     <DATETIME>2016-04-21T13:21:42Z</DATETIME>
                 </LAST_MODIFIED>
             </AUTH_WINDOWS>
             </AUTH_WINDOWS_LIST>
         <GLOSSARY>
             <USER_LIST>
                 <USER>
                     <USER_LOGIN>mark_t</USER_LOGIN>
                     <FIRST_NAME>Mark</FIRST_NAME>
                     <LAST_NAME>Twain</LAST_NAME>
                 </USER>
             </USER_LIST>
         </GLOSSARY>
       </RESPONSE>
</AUTH_WINDOWS_LIST_OUTPUT>
```

DTD update (Windows):

```
<!ELEMENT AUTH_WINDOWS_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_WINDOWS_LIST|ID_SET)?, WARNING_LIST?,
                    GLOSSARY?)>
<!ELEMENT AUTH_WINDOWS_LIST (AUTH_WINDOWS+)>

<!-- If WINDOWS_DOMAIN is set, then IP_SET is optional (not specified
means service selects IPs) -->
<!ELEMENT AUTH_WINDOWS (ID, TITLE, USERNAME, NTLM?, WINDOWS_DOMAIN?,
                        WINDOWS_AD_DOMAIN?, WINDOWS_AD_TRUST?, IP_SET?,
                        LOGIN_TYPE, DIGITAL_VAULT?, NETWORK_ID?,
                        CREATED, LAST_MODIFIED, COMMENTS?,
                        USE_AGENTLESS_TRACKING?, MINIMUM_SMB_VERSION?,
                        REQUIRE_SMB_SIGNING?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT NTLM (#PCDATA)>
<!ELEMENT WINDOWS_DOMAIN (#PCDATA)>
<!ELEMENT WINDOWS_AD_DOMAIN (#PCDATA)>
<!ELEMENT WINDOWS_AD_TRUST (#PCDATA)>

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

<!ELEMENT LOGIN_TYPE (#PCDATA)>
<!ELEMENT DIGITAL_VAULT (DIGITAL_VAULT_ID, DIGITAL_VAULT_TYPE,
                         DIGITAL_VAULT_TITLE, VAULT_FOLDER?,
                         VAULT_FILE?, VAULT_SECRET_NAME?,
                         VAULT_SYSTEM_NAME?, VAULT_EP_NAME?,
                         VAULT_EP_TYPE?, VAULT_EP_CONT?, VAULT_NS_TYPE?,
                         VAULT_NS_NAME?)>
<!ELEMENT DIGITAL_VAULT_ID (#PCDATA)>
<!ELEMENT DIGITAL_VAULT_TYPE (#PCDATA)>
<!ELEMENT DIGITAL_VAULT_TITLE (#PCDATA)>
<!ELEMENT VAULT_FOLDER (#PCDATA)>
<!ELEMENT VAULT_FILE (#PCDATA)>
<!ELEMENT VAULT_SECRET_NAME (#PCDATA)>
<!ELEMENT VAULT_SYSTEM_NAME (#PCDATA)>
<!ELEMENT VAULT_EP_NAME (#PCDATA)>
<!ELEMENT VAULT_EP_TYPE (#PCDATA)>
<!ELEMENT VAULT_EP_CONT (#PCDATA)>
<!ELEMENT VAULT_NS_TYPE (#PCDATA)>
<!ELEMENT VAULT_NS_NAME (#PCDATA)>

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

```
<!ELEMENT USE_AGENTLESS_TRACKING (#PCDATA)>
<!ELEMENT MINIMUM_SMB_VERSION (#PCDATA)>
<!ELEMENT REQUIRE_SMB_SIGNING (#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)>
```

API request (Unix):

```
curl -X -u "USERNAME:PASSWORD" GET -H "Authorization: Basic
cXVhc3lfdGI6cWF0ZW1w" -H "X-Requested-With: test"
"https://qualysapi.qualys.com/api/2.0/fo/auth/unix/?action=list&ids=30633
203"
```

XML output (Unix):

```
<?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>2016-04-14T09:51:42Z</DATETIME>
        <AUTH_UNIX_LIST>
            <AUTH_UNIX>
                <ID>30573203</ID>
                <TITLE>
                    <![CDATA[AA2014_API_v2_UNIX1]]>
                </TITLE>
                <USERNAME>
                    <![CDATA[Qualys]]>
                </USERNAME>
                <CLEARTEXT_PASSWORD>0</CLEARTEXT_PASSWORD>
                <ROOT_TOOL>None</ROOT_TOOL>
                <PORT>5857</PORT>
                <IP_SET>
                    <IP>10.113.195.151</IP>
                </IP_SET>
```

```
                <LOGIN_TYPE>
                    <![CDATA[vault]]>
                </LOGIN_TYPE>
                <CREATED>
                    <DATETIME>2016-04-14T09:49:15Z</DATETIME>
                    <BY>user_name</BY>
                </CREATED>
                <LAST_MODIFIED>
                    <DATETIME>2016-04-14T09:49:15Z</DATETIME>
                </LAST_MODIFIED>
            </AUTH_UNIX>
          </AUTH_UNIX_LIST>
       </RESPONSE>
</AUTH_UNIX_LIST_OUTPUT>
```

DTD update (Unix):

```
<!-- QUALYS AUTH_UNIX_LIST_OUTPUT DTD -->
<!ELEMENT AUTH_UNIX_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_UNIX_LIST|ID_SET)?, WARNING_LIST?,
                    GLOSSARY?)>
<!ELEMENT AUTH_UNIX_LIST (AUTH_UNIX+)>

<!ELEMENT AUTH_UNIX (ID, TITLE, USERNAME, CLEARTEXT_PASSWORD, ROOT_TOOL,
                     RSA_PRIVATE_KEY?, DSA_PRIVATE_KEY?, PORT?, IP_SET,
                     LOGIN_TYPE, DIGITAL_VAULT?, NETWORK_ID?, CREATED,
                     LAST_MODIFIED, COMMENTS?, USE_AGENTLESS_TRACKING?,
                     AGENTLESS_TRACKING_PATH?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT CLEARTEXT_PASSWORD (#PCDATA)>
<!ELEMENT ROOT_TOOL (#PCDATA)>
<!-- Private key contents will never be rendered -->
<!ELEMENT RSA_PRIVATE_KEY EMPTY>
<!ELEMENT DSA_PRIVATE_KEY EMPTY>
```

```
<!ELEMENT PORT (#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 USE_AGENTLESS_TRACKING (#PCDATA)>
<!ELEMENT AGENTLESS_TRACKING_PATH (#PCDATA)

<!ELEMENT DIGITAL_VAULT (DIGITAL_VAULT_ID, DIGITAL_VAULT_TYPE,
                         DIGITAL_VAULT_TITLE, VAULT_FOLDER?,
                         VAULT_FILE?, VAULT_SECRET_NAME?,
                         VAULT_SYSTEM_NAME?, VAULT_EP_NAME?,
                         VAULT_EP_TYPE?, VAULT_EP_CONT?,
                         VAULT_NS_TYPE?, VAULT_NS_NAME?)>
<!ELEMENT DIGITAL_VAULT_ID (#PCDATA)>
<!ELEMENT DIGITAL_VAULT_TYPE (#PCDATA)>
<!ELEMENT DIGITAL_VAULT_TITLE (#PCDATA)>
<!ELEMENT VAULT_FOLDER (#PCDATA)>
<!ELEMENT VAULT_FILE (#PCDATA)>
<!ELEMENT VAULT_SECRET_NAME (#PCDATA)>
<!ELEMENT VAULT_SYSTEM_NAME (#PCDATA)>
<!ELEMENT VAULT_EP_NAME (#PCDATA)>
<!ELEMENT VAULT_EP_TYPE (#PCDATA)>
<!ELEMENT VAULT_EP_CONT (#PCDATA)>
<!ELEMENT VAULT_NS_TYPE (#PCDATA)>
<!ELEMENT VAULT_NS_NAME (#PCDATA)>

<!ELEMENT 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 Authentication Record

You can now create an authentication record along with the vault information. You can update the authentication record to change it.

Good to Know: To add vault information to an authentication record, configure the new parameter login_type = vault. This is a mandatory parameter for creating or updating vault information. By default, this parameter is set to basic.

Input parameters

Parameter	Description
action=create/update	(Required)
login_type={value}	(Required only when you want to create or update vault information) The valid value is: basic vault Set login_type=vault, to add vault information. By default, the parameter is set to basic.
vault_id={value}	(Required only when action=create and login_type=vault) A vault ID.
	For Windows, vault_id and password parameters are mutually exclusive and cannot be specified in the same request.
	For Unix, vault_id and password, cleartext_password, rsa_private_key, dsa_private_key parameters are mutually exclusive and cannot be specified in the same request.
vault_type={value}	(Required only when action=create and login_type=vault) Include a certain vault type only. A valid value is: Cyber-Ark PIM Suite Thycotic Secret Server Quest Vault CA Access Control Hitachi ID PAM (There are no parameters specific to Hitachi IDPAM vault type.) Lieberman ERPM
Cyber-Ark PIM Suite	

Parameter	Description
folder={value}	(Required if vault type is 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 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>
Thycotic Secret Server:	
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.
Quest Vault	
system_name={value}	(Required if vault type is Quest Vault) Specify the system name. During a scan we'll perform a search for the system name and then retrieve the password. A single exact match of the system name must be found in order for authentication to be successful.
CA Access Control	
end_point_name={value}	(Required if vault type is CA Access Control) The End-Point name identifies a managed system, either a target for local accounts or a domain controller for domain accounts. An End-Point name is a user-defined value within your installation of CA Access Control Enterprise Management. The End-Point name entered in this record must match a pre-defined name exactly.

Parameter	Description
end_point_type={value}	(Required if vault type is CA Access Control) The End-Point type represents the method of access to the End-Point system. CA Access Control Enterprise Management uses pre-defined values for various methods and the End-Point type value must match a pre-defined value exactly. Examples: "Windows Agentless" (for Windows accounts) and "SSH Device" (for Unix via SSH).
end_point_container={value}	(Required if vault type is CA Access Control) The End-Point container stores configuration values. CA Access Control Enterprise Management uses pre-defined values for various methods and the End-Point container value must match a pre-defined value exactly. Examples: "Accounts" (for Windows accounts) and "SSH Accounts" (for Unix via SSH).
Lieberman ERPM	
auto_discover_system_name ={0|1}	(Required if vault type is Lieberman ERPM) Specify 1 to enable auto discovery of the system name and 0 to disable auto discovery. Each system in your ERPM environment has a system name and this is needed in order to retrieve the password for authentication. Use auto discovery to allow the service to find the system name for you at scan time. The service uses information known about each host (like the IP address and FQDN) to query ERPM for the system name. Auto discovery is the only option available when your record includes multiple IPs.
system_name_single_host= {value}	(Required if vault type is Lieberman ERPM) Specify the system name that is needed to retrieve password for authentication. To specify system_name_single_host, ensure that auto discovery of system name is disabled (auto_discover_system_name=0). If auto discovery of system name is enabled (auto_discover_system_name=1), specifying system_name_single_host is invalid.
system_type={value}	(Required if vault type is Lieberman ERPM) A valid value is one of the following system type: auto windows unix oracle mssq ldap cisco custom

Parameter	Description
custom_system_type={value}	(Required if vault type is Lieberman ERPM) Specify the custom system type name.
	custom_system_type is valid only when system_type=custom.

API request (Windows):

```
curl -X -u "USERNAME:PASSWORD" POST -H "Authorization: Basic
cXVhc3lfdGI6cWF0ZW1w" -H "X-Requested-With: test" -H "Content-Type:
application/x-www-form-urlencoded" -d
'action=create&title=Windows&username=Qualys&ips=10.113.195.151&login_typ
e=vault&vault_id=10873203&vault_type=Lieberman
ERPM&auto_discover_system_name=1&system_type=custom&custom_system_type=cu
stom name' "https://qualysapi.qualys.com/api/2.0/fo/auth/windows/"
```

XML output (Windows):

```
<?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>2016-04-21T13:21:42Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Created</TEXT>
                <ID_SET>
                    <ID>30623203</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

API request (Unix):

```
curl -X -u "USERNAME:PASSWORD" POST -H "Authorization: Basic
cXVhc3lfdGI6cWF0ZW1w" -H "X-Requested-With: test" -H "Content-Type:
application/x-www-form-urlencoded" -d 'action=create&title=Unix
vault&username=Qualys&ips=10.113.195.152&port=5857&login_type=vault&vault
_type=Lieberman
ERPM&vault_id=10873203&auto_discover_system_name=0&system_name_single_hos
t=a&custom_system_type=custom&system_type=custom'
"https://qualysapi.qualys.com/api/2.0/fo/auth/unix/"
```

XML output (Unix):

```
<?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>2016-04-21T13:26:03Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Created</TEXT>
                <ID_SET>
                    <ID>30633203</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Update Authentication Record Samples

API request (Windows):

```
curl -X -u "USERNAME:PASSWORD" POST -H "Authorization: Basic
cXVhc3lfdGI6cWF0ZW1w" -H "X-Requested-With: test" -H "Content-Type:
application/x-www-form-urlencoded" -d
'action=update&ids=30623203&login_type=vault&vault_id=10873203&vault_type
=Lieberman ERPM&auto_discover_system_name=0&system_name_single_host=as1'
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/"
```

XML output (Windows):

```
<?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>2016-04-21T13:29:08Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Updated</TEXT>
                <ID_SET>
                    <ID>30623203</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

API request (Unix):

```
curl -X POST -H "Authorization: Basic cXVhc3lfdGI6cWF0ZW1w" -H "X-
Requested-With: test" -H "Content-Type: application/x-www-form-
urlencoded" -d
'action=update&ids=30633203&root_tool=Sudo&login_type=vault&vault_type=Li
eberman ERPM&vault_id=10873203&auto_discover_system_name=1'
"https://qualysapi.qualys.com/api/2.0/fo/auth/unix/"
```

XML output (Unix):

```
<?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>2016-04-21T13:30:34Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Updated</TEXT>
                <ID_SET>
                    <ID>30633203</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

Choose Kerberos, NTLM protocols for Windows and MS SQL Authentication

You can now choose the authentication protocols you want to use for authentication to Windows and MS SQL Server target hosts. Your options are Kerberos, NTLMv2 and NTLMv1. You'll choose authentication protocols when defining login credentials for your authentication records.

Good to Know

- For Windows domain level authentication, all three authentication protocols are supported. Kerberos and NTLMv2 are enabled by default in new records. If NTLM was enabled in a record prior to this release, then NTLMv1 is enabled.
- For Windows local host level authentication, NTLMv2 and NTLMv1 protocols are supported. NTLMv2 is enabled by default in new records. If NTLM was enabled in a record prior to this release, then NTLMv1 is enabled.
- For MS SQL Server records (PC only), all three authentication protocols are supported. Kerberos and NTLMv2 are enabled by default in new records. MS SQL records created prior to this release will have all three protocols enabled.

Create/Update Windows Records

Use these input parameters to enable authentication protocols in Windows records.

Parameter	Description
kerberos={**0**|**1**}	(Optional) 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.
	Kerberos is supported for domain authentication only. When **kerberos=1** you must include **windows_ad_domain** or **windows_domain** in the same request.
ntlmv2={**0**|**1**}	(Optional) 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.
ntlm={**0**|1}	(Optional) When not specified, NTLMv1 will not be attempted. Specify **ntlm=1** to allow the scanning engine to try NTMLv1 when negotiating authentication to target hosts.

Sample

API request:

```
curl -H "X-Requested-With: curl demo" -u USERNAME:PASSWORD -d
"action=create&title=Win_Domain_Auth&username=Qualys&password=Password&ip
s=10.10.10.28&echo_request=1&windows_domain=esxi-50-33-
7.qualys.com&ntlm=1&ntlmv2=1&kerberos=1"
https://qualysapi.qualys.com/api/2.0/fo/auth/windows/
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE BATCH_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
  <REQUEST>
    <DATETIME>2016-05-06T09:18:41Z</DATETIME>
    <USER_LOGIN>qualys_test</USER_LOGIN>

<RESOURCE>https://qualysapi.qualys.com/api/2.0/fo/auth/windows/</RESOURCE
>
    <PARAM_LIST>
      <PARAM>
        <KEY>action</KEY>
        <VALUE>create</VALUE>
      </PARAM>
      <PARAM>
        <KEY>title</KEY>
        <VALUE>Win_Domain_Auth</VALUE>
      </PARAM>
      <PARAM>
        <KEY>username</KEY>
        <VALUE>Qualys</VALUE>
      </PARAM>
      <PARAM>
        <KEY>password</KEY>
        <VALUE>Password</VALUE>
      </PARAM>
      <PARAM>
        <KEY>ips</KEY>
        <VALUE>10.10.10.28</VALUE>
      </PARAM>
      <PARAM>
        <KEY>echo_request</KEY>
        <VALUE>1</VALUE>
      </PARAM>
      <PARAM>
        <KEY>windows_domain</KEY>
        <VALUE>esxi-50-33-7.qualys.com</VALUE>
```

```
      </PARAM>
      <PARAM>
        <KEY>ntlm</KEY>
        <VALUE>1</VALUE>
      </PARAM>
      <PARAM>
        <KEY>ntlmv2</KEY>
        <VALUE>1</VALUE>
      </PARAM>
      <PARAM>
        <KEY>kerberos</KEY>
        <VALUE>1</VALUE>
      </PARAM>
    </PARAM_LIST>
  </REQUEST>
  <RESPONSE>
    <DATETIME>2016-05-06T09:18:42Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>2704422279</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Create/Update MS SQL Records

Use these input parameters to enable authentication protocols in MS SQL records.

Parameter	Description
kerberos={0|**1**}	(Optional) 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) 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.
ntlmv1={**0**|1}	(Optional) When not specified, NTLMv1 will not be attempted. Specify **ntlmv1=1** to try NTMLv1 when negotiating authentication to target hosts.

Sample

API request:

```
curl -H "X-Requested-With: curl demo" -u USERNAME:PASSWORD -d
"action=create&title=MS_SQL_Auth&username=Qualys&password=Password&db_loc
al=1&ips=10.10.10.205&port=80&echo_request=0&comments=ACCOUNT_CREATED&ins
tance=MSSQLSERVER_1&database=master_1&ntlmv1=1&ntlmv2=1&kerberos=1"
https://qualysapi.qualys.com/api/2.0/fo/auth/ms_sql/
```

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>2016-05-06T11:57:48Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>2704452279</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

List Windows Records

You'll see the authentication protocols enabled for each record in your records list.

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_WINDOWS_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/auth_windows_list_o
utput.dtd">
<AUTH_WINDOWS_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2016-05-06T09:51:41Z</DATETIME>
      <AUTH_WINDOWS>
        <ID>2704422279</ID>
        <TITLE><![CDATA[Win_Domain_Auth]]></TITLE>
```

```
        <USERNAME><![CDATA[Qualys]]></USERNAME>
        <NTLM>1</NTLM>
        <NTLM_V2>1</NTLM_V2>
        <KERBEROS>1</KERBEROS>
        <WINDOWS_DOMAIN><![CDATA[esxi-50-33-
7.qualys.com]]></WINDOWS_DOMAIN>
        <IP_SET>
          <IP>10.10.10.28</IP>
        </IP_SET>
        <CREATED>
          <DATETIME>2016-05-06T09:18:41Z</DATETIME>
          <BY>qualys_test</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2016-05-06T09:51:31Z</DATETIME>
        </LAST_MODIFIED>
      </AUTH_WINDOWS>
    </AUTH_WINDOWS_LIST>
  </RESPONSE>
</AUTH_WINDOWS_LIST_OUTPUT>
```

DTD update:

We added the NTLM_V2 and KERBEROS elements to the Windows record list output DTD (/api/2.0/fo/auth/windows/auth_windows_list_output.dtd). The NTLM element existed prior to this release.

```
<!-- QUALYS AUTH_WINDOWS_LIST_OUTPUT DTD -->
...

<!ELEMENT AUTH_WINDOWS (ID, TITLE, USERNAME, NTLM?, NTLM_V2?, KERBEROS?,
                        WINDOWS_DOMAIN?, WINDOWS_AD_DOMAIN?,
                        WINDOWS_AD_TRUST?, IP_SET?, LOGIN_TYPE,
                        DIGITAL_VAULT?, NETWORK_ID?, CREATED,
                        LAST_MODIFIED, COMMENTS?,
                        USE_AGENTLESS_TRACKING?, MINIMUM_SMB_VERSION?,
                        REQUIRE_SMB_SIGNING?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT NTLM (#PCDATA)>
<!ELEMENT NTLM_V2 (#PCDATA)>
<!ELEMENT KERBEROS (#PCDATA)>
<!ELEMENT WINDOWS_DOMAIN (#PCDATA)>
<!ELEMENT WINDOWS_AD_DOMAIN (#PCDATA)>
<!ELEMENT WINDOWS_AD_TRUST (#PCDATA)>

...
```

List MS SQL Records

You'll see the authentication protocols enabled for each record in your records list.

API request:

```
curl -H "X-Requested-With: curl" -u "USERNAME:PASSWORD" -d
"action=list&echo_request=1"
"https://qualysapi.qualys.com/api/2.0/fo/auth/ms_sql/"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_MS_SQL_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/ms_sql/auth_ms_sql_list_out
put.dtd">
<AUTH_MS_SQL_LIST_OUTPUT>
  <REQUEST>
    <DATETIME>2016-05-06T11:58:53Z</DATETIME>
    <USER_LOGIN>qualys_test</USER_LOGIN>

<RESOURCE>https://qualysapi.qualys.com/api/2.0/fo/auth/ms_sql/</RESOURCE>
    <PARAM_LIST>
      <PARAM>
        <KEY>action</KEY>
        <VALUE>list</VALUE>
      </PARAM>
      <PARAM>
        <KEY>echo_request</KEY>
        <VALUE>1</VALUE>
      </PARAM>
    </PARAM_LIST>
  </REQUEST>
  <RESPONSE>
    <DATETIME>2016-05-06T11:58:53Z</DATETIME>
    <AUTH_MS_SQL_LIST>
      <AUTH_MS_SQL>
        <ID>2704452279</ID>
        <TITLE><![CDATA[MS_SQL_Auth]]></TITLE>
        <USERNAME><![CDATA[Qualys]]></USERNAME>
        <NTLM_V1>1</NTLM_V1>
        <NTLM_V2>1</NTLM_V2>
        <KERBEROS>1</KERBEROS>
        <INSTANCE><![CDATA[MSSQLSERVER_1]]></INSTANCE>
        <DATABASE><![CDATA[master_1]]></DATABASE>
        <PORT>80</PORT>
        <DB_LOCAL>1</DB_LOCAL>
        <IP_SET>
          <IP>10.10.10.205</IP>
        </IP_SET>
```

```
        <CREATED>
          <DATETIME>2016-05-06T11:57:48Z</DATETIME>
          <BY>qualys_test</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2016-05-06T11:57:48Z</DATETIME>
        </LAST_MODIFIED>
        <COMMENTS><![CDATA[ACCOUNT_CREATED]]></COMMENTS>
      </AUTH_MS_SQL>
    </AUTH_MS_SQL_LIST>
  </RESPONSE>
</AUTH_MS_SQL_LIST_OUTPUT>
```

DTD update:

We added the NTLM_V1, NTLM_V2 and KERBEROS elements to the MS SQL record list output DTD (/api/2.0/fo/auth/ms_sql/auth_ms_sql_list_output.dtd).

```
<!-- QUALYS AUTH_MS_SQL_LIST_OUTPUT DTD -->
...
<!ELEMENT AUTH_MS_SQL (ID, TITLE, USERNAME, NTLM_V1?, NTLM_V2?,
                       KERBEROS?, (INSTANCE | AUTO_DISCOVER_INSTANCES),
                      (DATABASE | AUTO_DISCOVER_DATABASES),
                      (PORT|AUTO_DISCOVER_PORTS), DB_LOCAL,
                       WINDOWS_DOMAIN?, IP_SET, NETWORK_ID?,
                       CREATED, LAST_MODIFIED, COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT NTLM_V1 (#PCDATA)>
<!ELEMENT NTLM_V2 (#PCDATA)>
<!ELEMENT KERBEROS (#PCDATA)>
...
```

Require SMB Signing for Windows Authentication

Use the Windows Record API(/api/2.0/fo/auth/windows) to define if the SMB protocol is required for Windows authentication or what should the minimum version of SMB be while authenticating Windows records. You can set the minimum required SMB version for authentication without enabling SMB signing required.

Authentication will fail for target hosts that have an SMB version that is older than the minimum version selected. For example, if you set a minimum of 2.0.2 and you scan a Windows host with version 1.0 then authentication will fail and the host will not be scanned.

Should I require SMB Signing?

The answer is No for most cases. This option is disabled by default, meaning SMB signing is not required. This is the recommended setting. When disabled, we can authenticate to any Windows version regardless of how SMB signing is configured on the target. You are not protected, however, against man-in-the-middle (MITM) attacks.

If you enable this option in your record, we will require each Windows target to support SMB signing. If SMB signing is disabled on a target host, authentication will fail and the host will not be scanned. This option protects against MITM attacks but we won't be able to authenticate to some hosts.

Parameter	Description
require_smb_signing	Set value to 0 (default) when SMB signing is not required. Set value to 1 to require SMB signing.
minimum_smb_version	Specify the minimum SMB protocol version. Valid values are: 1, 2.0.2, 2.1, 3.0, 3.0.2, 3.1.1

API request for SMB signing (list):

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: curl -k demo 2' -d
"action=list&ids=30642093"
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/"
```

XML output:

```
...
<NETWORK_ID>0</NETWORK_ID>
        <CREATED>
          <DATETIME>2016-05-04T09:52:35Z</DATETIME>
          <BY>USERNAME</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2016-05-04T09:52:35Z</DATETIME>
        </LAST_MODIFIED>
```

```
        <REQUIRE_SMB_SIGNING>1</REQUIRE_SMB_SIGNING>
      </AUTH_WINDOWS>
    </AUTH_WINDOWS_LIST>
  </RESPONSE>
...
```

API request for SMB signing and SMB version (create):

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: curl -k demo 2' -d
"action=create&title=smb required
2&username=jp&password=123456&ips=10.10.10.129&require_smb_signing=1&mini
mum_smb_version=2.1"
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/"
```

XML output:

```
...
<RESPONSE>
    <DATETIME>2016-05-04T09:53:57Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>30652093</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
...
```

API request for SMB signing and SMB version (update):

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: curl -k demo 2' -d
"action=update&ids=30652093&ips=10.10.10.130&require_smb_signing=0&minimu
m_smb_version=2.1"
"https://qualysapi.qualys.com/api/2.0/fo/auth/windows/"
```

XML output:

```
...
<RESPONSE>
    <DATETIME>2016-05-04T09:55:04Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Updated</TEXT>
        <ID_SET>
          <ID>30652093</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
```

```
  </RESPONSE>
...
```

API request for SMB version (list):

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: curl demo 2' -d
"action=list&ids=2687"
https://qualysapi.qualys.com/api/2.0/fo/auth/windows/index.php
```

XML output:

```
...
        <NETWORK_ID>0</NETWORK_ID>
                <CREATED>
                    <DATETIME>2014-12-10T12:09:41Z</DATETIME>
                    <BY>USERNAME</BY>
                </CREATED>
                <LAST_MODIFIED>
                    <DATETIME>2016-02-29T12:33:14Z</DATETIME>
                </LAST_MODIFIED>
                <MINIMUM_SMB_VERSION>3.0</MINIMUM_SMB_VERSION>
            </AUTH_WINDOWS>
        </AUTH_WINDOWS_LIST>
    </RESPONSE>
</AUTH_WINDOWS_LIST_OUTPUT>
...
```

DTD update:

We have added the elements REQUIRE_SMB_SIGNING and MINIMUM_SMB_VERSION to the Auth Windows List Output DTD (auth_windows_list_output.dtd)

```
<!-- QUALYS AUTH_WINDOWS_LIST_OUTPUT DTD -->
...
<!ELEMENT AUTH_WINDOWS (ID, TITLE, USERNAME, NTLM?, NTLM_V2?, KERBEROS?,
WINDOWS_DOMAIN?, WINDOWS_AD_DOMAIN?, WINDOWS_AD_TRUST?, IP_SET?,
LOGIN_TYPE?, DIGITAL_VAULT?, NETWORK_ID?, CREATED, LAST_MODIFIED,
COMMENTS?, USE_AGENTLESS_TRACKING?, MINIMUM_SMB_VERSION?,
REQUIRE_SMB_SIGNING?)>
...
<!ELEMENT MINIMUM_SMB_VERSION (#PCDATA)>
<!ELEMENT REQUIRE_SMB_SIGNING (#PCDATA)>
...
```

VM - Display CVSS v3 scores in reports

The new KnowledgeBase API (v2) (/api/2.0/fo/knowledge_base/vuln/?action=list) and KnowledgeBase Download (v1) (msp/knowledgebase_download.php) now lets you display the CVSS v3.0 scores of vulnerabilities in patch reports, scan reports, and KnowledgeBase data list. The CVSS v3 scores assigned to CVEs by NIST are displayed.

Qualys 8.8 shows CVSS v3 scores assigned to CVEs by NIST in vulnerability reports and KnowledgeBase QIDs. For reports we now show the CVSS v3 scores with CVSS v2 scores, and no changes were made to DTDs for scan reports, patch reports, scan results, KnowledgeBase datalist download.

These APIs were updated to add CVSS v3 scores to the XML output:

API	API Request
KnowledgeBase API (v2)	api/2.0/fo/knowledge_base/vuln/?action=list
KnowledgeBase Download (v1)	msp/knowledgebase_download.php
Scan Results	/api/2.0/fo/scan/?action=fetch
Scan Report	/api/2.0/fo/report/?action=fetch

KnowledgeBase API (v2)

This API returns the CVSS v3 Base scores and CVSS v3 Temporal scores of the QIDs in the KnowledgeBase.

API request:

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: curl demo 2' -d
"action=list& ids=105095, 87008,38477"
'https://qualysapi.qualys.com/api/2.0/fo/knowledge_base/vuln/'
```

XML output:

```
...
  <VULN_LIST>
            <VULN>
                <QID>105095</QID>
                <VULN_TYPE>Vulnerability</VULN_TYPE>
                <SEVERITY_LEVEL>4</SEVERITY_LEVEL>
                <TITLE>
                    <![CDATA[User(s) With Blank Password]]>
                </TITLE>
                <CATEGORY>Security Policy</CATEGORY>
                <LAST_SERVICE_MODIFICATION_DATETIME>2014-09-
22T17:45:19Z</LAST_SERVICE_MODIFICATION_DATETIME>
                <PUBLISHED_DATETIME>2005-02-
```

```
23T08:00:00Z</PUBLISHED_DATETIME>
                <PATCHABLE>0</PATCHABLE>
                <DIAGNOSIS>
                 <![CDATA[The users have the blank password in the shadow
file. These users connect to the system without entering a password.]]>
                </DIAGNOSIS>
                <CONSEQUENCE>
                    <![CDATA[An attacker may connect to the system by
knowing just the username.]]>
                </CONSEQUENCE>
                <SOLUTION>
                    <![CDATA[Set the password for all the users.]]>
                </SOLUTION>
                <CVSS>
                    <BASE source="service">7.5</BASE>
                    <TEMPORAL>7.1</TEMPORAL>
                    <VERSION>2</VERSION>
                </CVSS>
                <CVSS_V3>
                    <BASE>2.1</BASE>
                    <TEMPORAL>6.5</TEMPORAL>
                </CVSS_V3>
                <PCI_FLAG>1</PCI_FLAG>
                <DISCOVERY>
                    <REMOTE>0</REMOTE>
                    <AUTH_TYPE_LIST>
                        <AUTH_TYPE>Unix</AUTH_TYPE>
                    </AUTH_TYPE_LIST>
                </DISCOVERY>
            </VULN>
        </VULN_LIST>
        ...
```

DTD update:

We added the CVSS_V3 element to the KnowledgeBase Output DTD (knowledge_base_vuln_list_output.dtd).

```
<!-- QUALYS KNOWLEDGE_BASE_VULN_LIST_OUTPUT DTD -->
...
    <!ELEMENT VULN_LIST (VULN*)>
      <!ELEMENT VULN (QID, VULN_TYPE, SEVERITY_LEVEL, TITLE, CATEGORY?,
                      DETECTION_INFO?, LAST_CUSTOMIZATION?,
                      LAST_SERVICE_MODIFICATION_DATETIME?,
                      PUBLISHED_DATETIME, BUGTRAQ_LIST?, PATCHABLE,
                      SOFTWARE_LIST?, VENDOR_REFERENCE_LIST?, CVE_LIST?,
                      DIAGNOSIS?, DIAGNOSIS_COMMENT?, CONSEQUENCE?,
                      CONSEQUENCE_COMMENT?, SOLUTION?, SOLUTION_COMMENT?,
                      COMPLIANCE_LIST?, CORRELATION?, CVSS?, CVSS_V3?,
```

```
                    PCI_FLAG, PCI_REASONS?, SUPPORTED_MODULES?,
                    DISCOVERY )>
...
<!ELEMENT CVSS_V3 (BASE, TEMPORAL?, VERSION?, ACCESS?, IMPACT?,
                   AUTHENTICATION?, EXPLOITABILITY?, REMEDIATION_LEVEL?,
                   REPORT_CONFIDENCE?)>
...
```

KnowledgeBase Download (v1)

Use the parameters CVSS3_BASE and CVSS3_TEMPORAL to view the NIST CVSS v3 scores for each vulnerability.

API request:

```
curl -u "USERNAME:PASSWORD" -H 'X-Requested-With: curl demo 2' -d
"vuln_id=105095"
'https://qualysapi.qualys.com/msp/knowledgebase_download.php'
```

XML output:

```
...
<VULNS>
    <VULN>
        <QID>105095</QID>
        <VULN_TYPE>Vulnerability</VULN_TYPE>
        <SEVERITY_LEVEL>4</SEVERITY_LEVEL>
        <TITLE>
            <![CDATA[User(s) With Blank Password]]>
        </TITLE>
        <CATEGORY>Security Policy</CATEGORY>
        <LAST_UPDATE>
            <![CDATA[2014-09-22T17:45:19Z]]>
        </LAST_UPDATE>
        <PATCHABLE>0</PATCHABLE>
        <DIAGNOSIS>
           <![CDATA[The users have the blank password in the shadow file.
These users connect to the system without entering a password.]]>
        </DIAGNOSIS>
        <CONSEQUENCE>
           <![CDATA[An attacker may connect to the system by knowing just
the username.]]>
        </CONSEQUENCE>
        <SOLUTION>
            <![CDATA[Set the password for all the users.]]>
        </SOLUTION>
        <CVSS_BASE source="service">7.5</CVSS_BASE>
        <CVSS_TEMPORAL>7.1</CVSS_TEMPORAL>
        <CVSS3_BASE>2.1</CVSS3_BASE>
```

```
        <CVSS3_TEMPORAL>6.5</CVSS3_TEMPORAL>
    </VULN>
</VULNS>
...
```

DTD update:

We added the CVSS3_BASE and CVSS3_TEMPORAL elements to the Knowledge base download DTD (knowledgebase_download.dtd).

```
<!-- QUALYS KNOWLEDGEBASE DOWNLOAD DTD -->
...
<!ELEMENT VULN (QID, VULN_TYPE, SEVERITY_LEVEL, TITLE, CATEGORY?,
                DETECTION_INFO?, LAST_UPDATE?, BUGTRAQ_ID_LIST?,
                PATCHABLE, VENDOR_REFERENCE_LIST?, CVE_ID_LIST?,
                DIAGNOSIS?, CONSEQUENCE?, SOLUTION?, COMPLIANCE?,
                CORRELATION?, CVSS_BASE?, CVSS_TEMPORAL?, CVSS3_BASE?,
                CVSS3_TEMPORAL?, CVSS_ACCESS_VECTOR?,
                CVSS_ACCESS_COMPLEXITY?, CVSS_AUTHENTICATION?,
                CVSS_CONFIDENTIALITY_IMPACT?, CVSS_INTEGRITY_IMPACT?,
                CVSS_AVAILABILITY_IMPACT?, CVSS_EXPLOITABILITY?,
                CVSS_REMEDIATION_LEVEL?, CVSS_REPORT_CONFIDENCE?,
                PCI_FLAG?, PCI_REASONS?, SUPPORTED_MODULES?,DISCOVERY?)>
...
<!ELEMENT CVSS3_BASE (#PCDATA)>
<!ELEMENT CVSS3_TEMPORAL (#PCDATA)>
...
```

Scan Results XML

When you download scan results from your account you'll see CVSS v3 scores in the output. We updated the DTD (scan-1.dtd).

XML output:

```
...
<CVSS_BASE source="service">5.1</CVSS_BASE>
<CVSS_TEMPORAL>4.1</CVSS_TEMPORAL>
<CVSS3_BASE>6</CVSS3_BASE>
<CVSS3_TEMPORAL>2.2</CVSS3_TEMPORAL>
...
```

DTD update:

We added the CVSS3_BASE and CVSS3_TEMPORAL elements to the Scan Results DTD (scan-1.dtd).

```
...
<!ELEMENT VULN (TITLE, LAST_UPDATE?, CVSS_BASE?, CVSS_TEMPORAL?,
                CVSS3_BASE?, CVSS3_TEMPORAL?, PCI_FLAG, INSTANCE?,
```

```
                VENDOR_REFERENCE_LIST?, CVE_ID_LIST?, BUGTRAQ_ID_LIST?,
                DIAGNOSIS?, DIAGNOSIS_COMMENT?, CONSEQUENCE?,
                CONSEQUENCE_COMMENT?, SOLUTION?, SOLUTION_COMMENT?,
                COMPLIANCE?, CORRELATION?, RESULT?)>
...
<!ELEMENT CVSS3_BASE (#PCDATA)>
<!ELEMENT CVSS3_TEMPORAL (#PCDATA)>
...
<!ELEMENT PRACTICE (TITLE, LAST_UPDATE?, CVSS_BASE?, CVSS_TEMPORAL?,
                    CVSS3_BASE?, CVSS3_TEMPORAL?, PCI_FLAG, INSTANCE?,
                    VENDOR_REFERENCE_LIST?, CVE_ID_LIST?,
                    BUGTRAQ_ID_LIST?, DIAGNOSIS?, DIAGNOSIS_COMMENT?,
                    CONSEQUENCE?, CONSEQUENCE_COMMENT?,
                    SOLUTION?, SOLUTION_COMMENT?, COMPLIANCE?,
                    CORRELATION?, RESULT?)>
...
```

Scan Report XML

When you download a scan report (with host based findings) from your account you'll see CVSS v3 scores in the XML output. We updated the DTD (asset_data_report.dtd).

XML output:

```
...
<CVSS3_SCORE>
<CVSS3_BASE>2</CVSS3_BASE>
<CVSS3_TEMPORAL>4</CVSS3_TEMPORAL>
</CVSS3_SCORE>
...
```

DTD update:

We added the CVSS3_SCORE, CVSS3_BASE and CVSS3_TEMPORAL elements to the Asset Data Report DTD (asset_data_report.dtd).

```
...
<!ELEMENT VULN_DETAILS (QID, TITLE, SEVERITY, CATEGORY,
                        CUSTOMIZED?, THREAT, THREAT_COMMENT?, IMPACT,
                        IMPACT_COMMENT?, SOLUTION, SOLUTION_COMMENT?,
                        COMPLIANCE?, CORRELATION?, PCI_FLAG,
                        LAST_UPDATE?, CVSS_SCORE?, CVSS3_SCORE?,
                        VENDOR_REFERENCE_LIST?,
                        CVE_ID_LIST?, BUGTRAQ_ID_LIST?)>
...
<!ELEMENT CVSS3_SCORE (CVSS3_BASE?, CVSS3_TEMPORAL?)>
<!ELEMENT CVSS3_BASE (#PCDATA)>
<!ELEMENT CVSS3_TEMPORAL (#PCDATA)>
...
```

VM - New Asset Search Report

The new Asset Search API v2 (/api/2.0/fo/report/asset) helps you easily create reports on assets you're interested in. The new DTD for the asset search report is available here: https://<baseurl>/api/2.0/fo/report/asset/asset_search_report_v2.dtd.

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl"
"https://qualysapi.qualys.com/api/2.0/fo/report/asset/?action=search&outp
ut_format=xml&echo_request=1&ips=10.10.10.10-10.10.10.20"
```

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE ASSET_SEARCH_REPORT SYSTEM
"https://qualysapi.qualys.com/asset_search_report_v2.dtd">

<ASSET_SEARCH_REPORT>
<HEADER>
  <REQUEST>
    <DATETIME>2016-06-03T20:21:13Z</DATETIME>
    <USER_LOGIN>john_sm</USER_LOGIN>
     <RESOURCE>https://qualysapi.qualys.com/api/2.0/fo/report/asset/
     </RESOURCE>
    <PARAM_LIST>
      <PARAM>
        <KEY>action</KEY>
        <VALUE>search</VALUE>
      </PARAM>
      <PARAM>
        <KEY>output_format</KEY>
        <VALUE>xml</VALUE>
      </PARAM>
      <PARAM>
        <KEY>echo_request</KEY>
        <VALUE>1</VALUE>
      </PARAM>
      <PARAM>
        <KEY>ips</KEY>
        <VALUE>10.10.10.10-10.10.10.15</VALUE>
      </PARAM>
    </PARAM_LIST>
  </REQUEST>
  <COMPANY>Corsa</COMPANY>
  <USERNAME>John Smith</USERNAME>
  <GENERATION_DATETIME>2016-06-03T20:21:13Z</GENERATION_DATETIME>
  <TOTAL>2</TOTAL>
  <FILTERS>
```

```
    <IP_LIST>
      <RANGE>
        <START>10.10.10.10</START>
        <END>10.10.10.15</END>
      </RANGE>
    </IP_LIST>
  </FILTERS>
</HEADER>

<HOST_LIST>
  <HOST>
    <IP><![CDATA[10.10.10.10]]></IP>
    <TRACKING_METHOD>IP address</TRACKING_METHOD>
    <OPERATING_SYSTEM><![CDATA[Linux 2.4-2.6 / Embedded Device / F5
Networks Big-IP]]></OPERATING_SYSTEM>
    <LAST_SCAN_DATE>2016-06-03T09:11:21Z</LAST_SCAN_DATE>
    <FIRST_FOUND_DATE>2016-06-03T07:11:46Z</FIRST_FOUND_DATE>
  </HOST>

  <HOST>
    <IP><![CDATA[10.10.10.11]]></IP>
    <TRACKING_METHOD>IP address</TRACKING_METHOD>
    <DNS><![CDATA[10-10-10-11.bogus.tld]]></DNS>
    <NETBIOS><![CDATA[SYS_10_10_10_11]]></NETBIOS>
    <OPERATING_SYSTEM><![CDATA[Windows 2000 Server Service Pack
4]]></OPERATING_SYSTEM>
    <LAST_SCAN_DATE>2016-06-03T07:12:47Z</LAST_SCAN_DATE>
    <LAST_COMPLIANCE_SCAN_DATE>2016-05-
13T21:15:01Z</LAST_COMPLIANCE_SCAN_DATE>
    <FIRST_FOUND_DATE>2016-05-12T15:16:54Z</FIRST_FOUND_DATE>
  </HOST>

</HOST_LIST>
</ASSET_SEARCH_REPORT>
```

CSV Output

```
----BEGIN_RESPONSE_HEADER_CSV
"Launch Datetime","User Login","Resource","Parameter Name","Parameter
Value"
"2016-06-
07T22:51:23Z","john_sm","https://qualysapi.qualys.com/api/2.0/fo/report/a
sset/",,
,,,"action","search"
,,,"output_format","csv"
,,,"echo_request","1"
,,,"ips","10.10.10.10-10.10.10.20"
----END_RESPONSE_HEADER_CSV
"Company","UserName","ReportDate","AssetGroups","IPAddresses","DNSHostnam
```

```
e","NetBIOSHostname","TargetTrackingMethod","TargetOperatingSystem","Targ
etService","TargetPort","TargetQID","QIDTitle","TargetLastScanDate","Targ
etFirstFoundDate","OSCPE","Tags","TargetComplianceLastScanDate","Total"
"Corsa","John Smith","2016-06-07T22:51:23Z",,"10.10.10.10-
10.10.10.20",,,,,,,,,,,,,"2"
"IP","DNSHostname","NetBIOSHostname","OperatingSystem","OSCPE","Port/Serv
ice/Default
Service","TrackingMethod","LastScanDate","LastComplianceScanDate","First
Found","Tags"
"10.10.10.10",,,"Linux 2.4-2.6 / Embedded Device / F5 Networks Big-
IP",,,"IP address","2016-06-03T09:11:21Z",,"2016-06-03T07:11:46Z",
"10.10.10.11",,"SYS_10_10_10_11",,,,"IP address","2016-06-
03T07:12:47Z","2016-05-13T21:15:01Z","2016-05-12T15:16:54Z",
```

DTD (new):

```
<!-- QUALYS ASSET SEARCH REPORT DTD -->

<!ELEMENT ASSET_SEARCH_REPORT (ERROR | (HEADER, HOST_LIST?))>

<!ELEMENT ERROR (#PCDATA)*>
<!ATTLIST ERROR number CDATA #IMPLIED>


<!-- HEADER -->

<!ELEMENT HEADER (REQUEST?, COMPANY, USERNAME, GENERATION_DATETIME,
TOTAL?, FILTERS)>

<!-- REQUEST Header -->
<!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 COMPANY (#PCDATA)>
<!ELEMENT USERNAME (#PCDATA)>
<!ELEMENT GENERATION_DATETIME (#PCDATA)>
<!ELEMENT FILTERS
((IP_LIST|ASSET_GROUPS|ASSET_TAGS|FILTER_DNS|FILTER_NETBIOS|TRACKING_METH
OD|
```

```
FILTER_OPERATING_SYSTEM|FILTER_OS_CPE|FILTER_PORT|FILTER_SERVICE|

FILTER_QID|FILTER_RESULT|FILTER_LAST_SCAN_DATE|FILTER_FIRST_FOUND_DATE|NE
TWORK|FILTER_DISPLAY_AG_TITLES|FILTER_QID_WITH_TEXT|FILTER_LAST_COMPLIANC
E_SCAN_DATE)+)>

<!ELEMENT IP_LIST (RANGE*)>
<!ELEMENT RANGE (START, END)>
<!ELEMENT START (#PCDATA)>
<!ELEMENT END (#PCDATA)>

<!ELEMENT ASSET_GROUPS (ASSET_GROUP_TITLE+)>
<!ELEMENT ASSET_GROUP_TITLE (#PCDATA)>
<!ELEMENT NETWORK (#PCDATA)>
<!ELEMENT ASSET_TAGS (INCLUDED_TAGS, EXCLUDED_TAGS?)>

<!ELEMENT INCLUDED_TAGS (ASSET_TAG*)>
<!ATTLIST INCLUDED_TAGS scope CDATA #IMPLIED>

<!ELEMENT EXCLUDED_TAGS (ASSET_TAG*)>
<!ATTLIST EXCLUDED_TAGS scope CDATA #IMPLIED>

<!ELEMENT ASSET_TAG (#PCDATA)>

<!ELEMENT FILTER_DNS (#PCDATA)>
<!ATTLIST FILTER_DNS criterion CDATA #IMPLIED>

<!ELEMENT FILTER_NETBIOS (#PCDATA)>
<!ATTLIST FILTER_NETBIOS criterion CDATA #IMPLIED>

<!ELEMENT TRACKING_METHOD (#PCDATA)>

<!ELEMENT FILTER_OPERATING_SYSTEM (#PCDATA)>
<!ATTLIST FILTER_OPERATING_SYSTEM criterion CDATA #IMPLIED>
<!ELEMENT FILTER_OS_CPE (#PCDATA)>
<!ELEMENT FILTER_PORT (#PCDATA)>
<!ELEMENT FILTER_SERVICE (#PCDATA)>
<!ELEMENT FILTER_QID (#PCDATA)>
<!ELEMENT FILTER_RESULT (#PCDATA)>
<!ATTLIST FILTER_RESULT criterion CDATA #IMPLIED>
<!ELEMENT FILTER_LAST_SCAN_DATE (#PCDATA)>
<!ATTLIST FILTER_LAST_SCAN_DATE criterion CDATA #IMPLIED>
<!ELEMENT FILTER_LAST_COMPLIANCE_SCAN_DATE (#PCDATA)>
<!ATTLIST FILTER_LAST_COMPLIANCE_SCAN_DATE criterion CDATA #IMPLIED>
<!ELEMENT FILTER_FIRST_FOUND_DATE (#PCDATA)>
<!ELEMENT FILTER_DISPLAY_AG_TITLES (#PCDATA)>
<!ELEMENT FILTER_QID_WITH_TEXT (#PCDATA)>
<!ELEMENT TOTAL (#PCDATA)>
<!-- HOST_LIST -->
```

```
<!ELEMENT HOST_LIST ((HOST|WARNING)*)>

<!ELEMENT HOST (ERROR | (IP, HOST_TAGS?,TRACKING_METHOD,
                         DNS?, NETBIOS?, OPERATING_SYSTEM?, OS_CPE?,
QID_LIST?, PORT_SERVICE_LIST?,
                         ASSET_GROUPS?, NETWORK?, LAST_SCAN_DATE?,
LAST_COMPLIANCE_SCAN_DATE?, FIRST_FOUND_DATE?))>

<!ELEMENT IP (#PCDATA)>
<!ATTLIST IP network_id CDATA #IMPLIED>
<!ELEMENT HOST_TAGS (#PCDATA)>
<!ELEMENT DNS (#PCDATA)>
<!ELEMENT NETBIOS (#PCDATA)>
<!ELEMENT OPERATING_SYSTEM (#PCDATA)>
<!ELEMENT OS_CPE (#PCDATA)>
<!ELEMENT QID_LIST (QID+)>
<!ELEMENT QID (ID, RESULT?)>
<!ELEMENT ID (#PCDATA)>
<!-- if format is set to "table" -->
<!-- tab '\t' is the col separator -->
<!-- and new line '\n' is the end of row -->
<!ELEMENT RESULT (#PCDATA)>
<!ATTLIST RESULT
    format CDATA #IMPLIED
>
<!ELEMENT PORT_SERVICE_LIST (PORT_SERVICE+)>
<!ELEMENT PORT_SERVICE (PORT, SERVICE, DEFAULT_SERVICE?)>
<!ELEMENT PORT (#PCDATA)>
<!ELEMENT SERVICE (#PCDATA)>
<!ELEMENT DEFAULT_SERVICE (#PCDATA)>
<!ELEMENT LAST_SCAN_DATE (#PCDATA)>
<!ELEMENT LAST_COMPLIANCE_SCAN_DATE (#PCDATA)>
<!ELEMENT FIRST_FOUND_DATE (#PCDATA)>

<!ELEMENT WARNING (#PCDATA)>
<!ATTLIST WARNING number CDATA #IMPLIED>
```

Input parameters:

Parameter	Description
action=search	(Required) GET or POST method may be used.
output_format={csv | xml}	(Required) The output format of the asset search report. One output format may be specified: csv or xml.
tracking_method={value}	(Optional) Show only IP addresses/ranges which have a certain tracking method. A valid value is: IP, DNS, NETBIOS, EC2, or AGENT.

Parameter	Description
ips={value}	(Optional) Show only certain IP addresses/ranges. Use this parameter if you want to include only certain IP addresses in the report. One or more IPs/ranges may be specified. Multiple entries are comma separated. An IP range is specified with a hyphen (for example, 10.10.10.1-10.10.10.100).
	One of these parameters must be specified in a request: ips, asset_groups, asset_group_ids, or use_tags.
ips_network_id={value}	(Optional) The network ID applied on IPs. The default value is ALL.
asset_group_ids={value}	(Optional) The IDs of asset groups containing the hosts to be included in the asset search report. Multiple IDs are comma separated.
	One of these parameters must be specified in a request: ips, asset_groups, asset_group_ids, or use_tags.
asset_groups={value}	(Optional) The titles of asset groups containing the hosts to be included in the asset search report. Multiple titles are comma separated.
	One of these parameters must be specified in a request: ips, asset_groups, asset_group_ids, or use_tags.
assets_in_my_network_only={**0**|1}	(Optional) Specify 1 to include the specified asset groups and/or IP ranges. Valid for 'All' Asset Group and/or specified IP ranges.
display_ag_titles={**0**|1}	(Optional) Specify 1 to display AssetGroup Titles for each Host in the output. Otherwise the AssetGroup Titles are not displayed in the output.
ports={value}	(Optional) Shows the hosts that has the specified open ports. One or more ports may be specified. Multiple ports are comma separated. You can specify upto 10 values.
services={value}	(Optional) Shows the hosts that has the specified services running on it. One or more services may be specified. Multiple services are comma separated.You can specify upto 10 values.
qids={value}	(Optional) Shows vulnerabilities (QIDs) in the KnowledgeBase applicable to the host. Allows up to 20 values
qid_with_text={value}	(Optional) Shows vulnerabilities (QIDs) with the specified text in the KnowledgeBase applicable to the host.
	qid_with_text is valid only when qids parameter is specified.

Parameter	Description
qid_with_modifier={value}	(Optional) Shows vulnerabilities (QIDs) with the specified criteria in the KnowledgeBase applicable to the host.
	qid_with_modifier is valid only when qid_with_text is specified.
use_tags={**0** | 1}}	(Optional) Specify 0 (the default) if you want to select hosts based on IP addresses/ranges and/or asset groups. Specify 1 if you want to select hosts based on asset tags.
	One of these parameters must be specified in a request: ips, asset_groups, asset_group_ids, or use_tags.
tag_set_by={id | name}	(Optional when use_tags=1) Specify "id" (the default) to select a tag set by providing tag IDs. Specify "name" to select a tag set by providing tag names.
tag_include_selector={any | all}	(Optional when use_tags=1) Select "any" (the default) to include hosts that match at least one of the selected tags. Select "all" to include hosts that match all of the selected tags.
tag_exclude_selector={any | all}	(Optional when use_tags=1) Select "any" (the default) to exclude hosts that match at least one of the selected tags. Select "all" to exclude hosts that match all of the selected tags.
tag_set_include={value}	(Required when use_tags=1) Specify a tag set to include. Hosts that match these tags will be included. You identify the tag set by providing tag name or IDs. Multiple entries are comma separated.
tag_set_exclude={value}	(Optional when use_tags=1) Specify a tag set to exclude. Hosts that match these tags will be excluded. You identify the tag set by providing tag name or IDs. Multiple entries are comma separated.
first_found_days={value}	(Optional) Specify a number of days along with the first_found_modifier so that the range includes the first found date to be searched for
	first_found_days is valid only when first_found_modifier is specified.
first_found_modifier={within | not within}	(Optional) Show only hosts whose first found date is within or not within the specified days.
	first_found_modifier is valid only when first_found_days is specified.

Parameter	Description
last_vm_scan_days={value}	(Optional) Specify a number of days so that it includes the last vm scan date to be searched for.
	last_vm_scan_days is valid only when last_vm_scan_modifier is specified.
last_vm_scan_modifier={within|not within}	(Optional) Show only hosts whose last_vm_scan_date is within or not within the specified days.
	last_vm_scan_modifier is valid only when last_vm_scan_days is specified.
last_pc_scan_days={value}	(Optional) Specify a number of days so that the specified value along with the modifier forms the date range that includes the last scan date to be searched for.
	This parameter is valid only when the policy compliance module is enabled for the user account.
last_pc_scan_modifier={within|not within}	(Optional) Show only hosts whose last_pc_scan_date is within or not within the specified days.
	This parameter is valid only when the policy compliance module is enabled for the user account.
dns_name={value}	(Optional) Specify the DNS name of the host that needs to be searched.
	dns_name is valid only when dns_modifier is specified.
dns_modifier={value}	(Optional) Show only hosts with dns_name that is either: beginning with, containing, matching, ending with, not empty.
	dns_modifier is valid only when dns_name is specified.
netbios_name={value}	(Optional) Specify the NETBIOS name of the host to be searched.
	netbios_name is valid only when netbios_modifier is specified.
netbios_modifier={value}	(Optional) Show only hosts with netbios_name that is either: beginning with, containing, matching, ending with, not empty.
	netbios_modifier is valid only when netbios_name is specified.
os_cpe_name={value}	(Optional) Specify the OS CPE name of the host to searched.
	os_cpe_name is valid only when os_cpe_name is specified.

Parameter	Description
os_cpe_modifier={value}	(Optional)) Show only hosts with os cpe_name that is either: beginning with, containing, matching, ending with, not empty.
	os_cpe_modifier is valid only when os_cpe_name is specified.
os_name={value}	(Optional) Specify the operating system name of the host to be searched.
	os_name is valid only when os_modifier is specified.
os_modifier={value}	(Optional) Show only hosts with os_name that is either: beginning with, containing, matching, ending with.
	os_modifier is valid only when os_name is specified.

VM - Dynamic Search List API v2

You can now use Dynamic Search List API (v2) (/api/2.0/fo/qid/search_list/dynamic/) to set the "cvss_base_operand", "cvss_temp_operand", "cvss3_base_operand" and "cvss3_temp_operand" parameters to request for CVSS and CVSS3 scores less than specified CVSS and CVSS3 Base and Temporal values. Earlier you could only select the greater than equal to operand while creating or updating the dynamic search list.

Create / Update Dynamic Search Lists

Use the "cvss_base_operand", "cvss_temp_operand", "cvss3_base_operand" and "cvss3_temp_operand" to filter CVSS and CVSS3 base and temporal score in dynamic search list criteria.

Parameter	Description
cvss_base_operand	Set the value to **1** to use the greater than equal to operand. Set the value to **2** to use the less than operand. You must always specify the "cvss_base" parameter along with the "cvss_base_operand" parameter in the API request.
cvss_temp_operand	Set the value to **1** to use the greater than equal to operand. Set the value to **2** to use the less than operand. You must always specify the "cvss_temp" parameter along with the "cvss_temp_operand" parameter in the API request.
cvss3_base	CVSS3 base score value assigned to the CVEs by NIST (matches greater than, less than, or equal to this value); to unset value use update request and set to empty value.
cvss3_temp	CVSS3 temporal score value assigned to the CVEs by NIST (matches greater than, less than, or equal to this value); to unset value use update request and set to empty value.
cvss3_base_operand	Set the value to **1** to use the greater than equal to operand. Set the value to **2** to use the less than operand. You must always specify the "cvss3_base" parameter along with the "cvss3_base_operand" parameter in the API request.
cvss3_temp_operand	Set the value to **1** to use the greater than equal to operand. Set the value to **2** to use the less than operand. You must always specify the "cvss3_temp" parameter along with the "cvss3_temp_operand" parameter in the API request.

API request (create list):

```
curl -u ’USERNAME:PASSWORD’ -H 'X-Requested-With:curl demo2' -d
"action=create&title=mytest_DL313&cvss_base=3&cvss_base_operand=1&cvss_te
mp=2&cvss_temp_operand=2&cvss3_base=2&cvss3_base_operand=1&cvss3_temp=2&c
vss3_temp_operand=2"
"https://qualysapi.qualys.com/api/2.0/fo/qid/search_list/dynamic/"
```

XML output:

```
...
   <RESPONSE>
        <DATETIME>2016-05-19T10:16:51Z</DATETIME>
        <TEXT>New search list created successfully</TEXT>
        <ITEM_LIST>
            <ITEM>
                <KEY>ID</KEY>
                <VALUE>248892093</VALUE>
            </ITEM>
        </ITEM_LIST>
    </RESPONSE>...
```

API request (update list):

```
curl -k -S -u 'USERNAME:PASSWORD' -H 'X-Requested-With:curl demo2' -d
"action=update&id=248882093&cvss_base=3&cvss_base_operand=1&cvss_temp=2&c
vss_temp_operand=2&cvss3_base=5&cvss3_base_operand=1&cvss3_temp=4&cvss3_t
emp_operand=2"
"https://qualysapi.qualys.com/api/2.0/fo/qid/search_list/dynamic/"
```

XML output:

```
...
 <RESPONSE>
        <DATETIME>2016-05-19T10:23:37Z</DATETIME>
        <TEXT>search list updated successfully</TEXT>
        <ITEM_LIST>
            <ITEM>
                <KEY>ID</KEY>
                <VALUE>248882093</VALUE>
            </ITEM>
        </ITEM_LIST>
    </RESPONSE>
...
```

API request (List Dynamic Search list API)

```
curl -u ’USERNAME:PASSWORD’  -H 'X-Requested-With:curl demo2'
"https://qualysapi.qualys.com/api/2.0/fo/qid/search_list/dynamic/?action=
list&ids=242443203"
```

XML output:

```
...
         <CRITERIA>
          <DISCOVERY_METHOD><![CDATA[All]]></DISCOVERY_METHOD>
          <CVSS_BASE_SCORE><![CDATA[1.2]]></CVSS_BASE_SCORE>
          <CVSS_TEMPORAL_SCORE><![CDATA[2]]></CVSS_TEMPORAL_SCORE>
          <CVSS3_BASE_SCORE><![CDATA[5.2]]></CVSS3_BASE_SCORE>
          <CVSS3_TEMPORAL_SCORE><![CDATA[3]]></CVSS3_TEMPORAL_SCORE>
          <CVSS_BASE_SCORE_OPERAND><![CDATA[&lt;]]>
             </CVSS_BASE_SCORE_OPERAND>
          <CVSS_TEMPORAL_SCORE_OPERAND><![CDATA[&gt;=]]>
             </CVSS_TEMPORAL_SCORE_OPERAND>
          <CVSS3_BASE_SCORE_OPERAND><![CDATA[&gt;=]]>
             </CVSS3_BASE_SCORE_OPERAND>
          <CVSS3_TEMPORAL_SCORE_OPERAND><![CDATA[&lt;]]>
             </CVSS3_TEMPORAL_SCORE_OPERAND>
        </CRITERIA>...
```

DTD output:

We added the CVSS3_BASE_SCORE?, CVSS3_TEMPORAL_SCORE?, CVSS_BASE_SCORE_OPERAND?, CVSS_TEMPORAL_SCORE_OPERAND?, CVSS3_BASE_SCORE_OPERAND?, and CVSS3_TEMPORAL_SCORE_OPERAND? elements to the Dynamic Search List Output DTD (dynamic_list_output.dtd)

```
<!-- QUALYS DYNAMIC_SEARCH_LIST_OUTPUT DTD -->
...
<!ELEMENT CRITERIA (VULNERABILITY_TITLE?, DISCOVERY_METHOD?,
AUTHENTICATION_TYPE?, USER_CONFIGURATION?, CATEGORY?,
CONFIRMED_SEVERITY?, POTENTIAL_SEVERITY?, INFORMATION_SEVERITY?, VENDOR?,
PRODUCT?, CVSS_BASE_SCORE?, CVSS_TEMPORAL_SCORE?, CVSS3_BASE_SCORE?,
CVSS3_TEMPORAL_SCORE?, CVSS_ACCESS_VECTOR?, PATCH_AVAILABLE?,
VIRTUAL_PATCH_AVAILABLE?, CVE_ID?, EXPLOITABILITY?, ASSOCIATED_MALWARE?,
VENDOR_REFERENCE?, BUGTRAQ_ID?, VULNERABILITY_DETAILS?,
SUPPORTED_MODULES?, COMPLIANCE_DETAILS?, COMPLIANCE_TYPE?,
QUALYS_TOP_20?, OTHER?, NETWORK_ACCESS?, PROVIDER?,
CVSS_BASE_SCORE_OPERAND?, CVSS_TEMPORAL_SCORE_OPERAND?,
CVSS3_BASE_SCORE_OPERAND?, CVSS3_TEMPORAL_SCORE_OPERAND?, USER_MODIFIED?,
PUBLISHED?, SERVICE_MODIFIED?)>
...
<!ELEMENT PROVIDER (#PCDATA)>
<!ELEMENT CVSS_BASE_SCORE_OPERAND (#PCDATA)>
<!ELEMENT CVSS_TEMPORAL_SCORE_OPERAND (#PCDATA)>
<!ELEMENT CVSS3_BASE_SCORE (#PCDATA)>
<!ELEMENT CVSS3_TEMPORAL_SCORE (#PCDATA)>
<!ELEMENT CVSS3_BASE_SCORE_OPERAND (#PCDATA)>
<!ELEMENT CVSS3_TEMPORAL_SCORE_OPERAND (#PCDATA)>
<!ELEMENT OPTION_PROFILES (OPTION_PROFILE+)>
...
```

VM - Scan API - Fetch Host Data from Scan Results

More information is provided when you fetch scan results using the API. Use the output formats json_extended and csv_extended. You'll see scan summary details at the top of your report and more host and vulnerability information for each detection.

The scan summary includes: company details (name, address), user details (name, login, role), scan date, number of active hosts, number of total hosts, scan type (On Demand or Scheduled), status, scan reference, scanner appliance, scan duration, scan title, asset groups, IPs, excluded IPs, and the option profile used.

The scan results section was updated to include: operating system, IP status, vulnerability title, type, severity, port, protocol, FQDN, SSL, CVE ID, vendor reference, Bugtraq ID, CVSS scores, threat, impact, solution, exploitability, associated malware, PCI vuln flag, OS CPE and category.

API request (JSON):

```
curl -H "X-Requested-With: curl" -u "USERNAME:PASSWORD" -X "POST" -d
"action=fetch&scan_ref=scan/1458209465.03203&output_format=json_extended&
ips=10.10.10.1-10.10.10.5"
"https://qualysapi.qualys.com/api/2.0/fo/scan/"
```

JSON Output:

```
[{"scan_report_template_title":"Scan Results","result_date":"05\/10\/2016
at 07:45:39AM (GMT+0000)","company":"\"Qualys\"","add1":"1600 Bridge
Pkwy","add2":"2nd Floor.","city":"Redwood
City","state":"California","country":"United States of
America","zip":"94065","name":"Patrick
Slimmer","username":"qualys_ps","role":"Manager"},
{"launch_date":"05\/10\/2016
07:06:12","active_hosts":"35","total_hosts":"63","type":"On
Demand","status":"Finished","reference":"scan\/146283973.3229","scanner_a
ppliance":"10.10.21.160 (Scanner 8.0.15-1, Vulnerability Signatures
2.3.263-
2)","duration":"00:22:38","scan_title":"My_VM_Scan","asset_groups":null,"
ips":"10.10.10.23,10.10.10.28,10.10.10.65,10.10.24.8,10.10.24.10-
10.10.24.12,10.10.24.67-10.10.24.95,10.10.25.27-
10.10.25.44,10.10.25.163,10.10.25.224,10.10.26.229,10.10.30.159,10.10.32.
17,10.10.33.124,10.11.72.21,10.20.30.187,10.40.1.200","excluded_ips":"","
option_profile":"Initial Options"},
{"ip":"10.10.10.28","dns":"xpsp3-10-28test","netbios":"XPSP3-10-
28TEST","os":"Windows XP Service Pack 3","ip_status":"host scanned, found
vuln","qid":70000,"title":"NetBIOS Name
Accessible","type":"Vuln","severity":"2","port":"","protocol":"","fqdn":"
","ssl":"no","cve_id":null,"vendor_reference":null,"bugtraq_id":null,"cvs
s_base":"0 (AV:N\/AC:L\/Au:N\/C:N\/I:N\/A:N)","cvss_temporal":"0
(E:H\/RL:W\/RC:C)","threat":"Unauthorized users can obtain this host's
```

```
NetBIOS server name from a remote system.","impact":"Unauthorized users
can obtain the list of NetBIOS servers on your network.  This list
outlines trust relationships between server and client computers.
Unauthorized users can therefore use a vulnerable host to penetrate secure
servers.","solution":"If the NetBIOS service is not required on this host,
disable it. Otherwise, block any NetBIOS traffic at your network
boundaries.","exploitability":null,"associated_malware":null,"results":"X
PSP3-10-
28TEST","pci_vuln":"no","instance":null,"os_cpe":null,"category":"SMB \/
NETBIOS"},
...
```

API request (CSV):

```
curl -H "X-Requested-With: curl" -u "USERNAME:PASSWORD" -X "POST" -d
"action=fetch&scan_ref=scan/1458209465.03203&output_format=csv_extended&i
ps=10.10.10.1-10.10.10.5" "https://qualysapi.qualys.com/api/2.0/fo/scan/"
```

CSV output:

The new scan summary section is highlighted in the sample below. You'll also see new columns in the scan results section. Not all columns are shown. Fetch your own scan to see a full report.

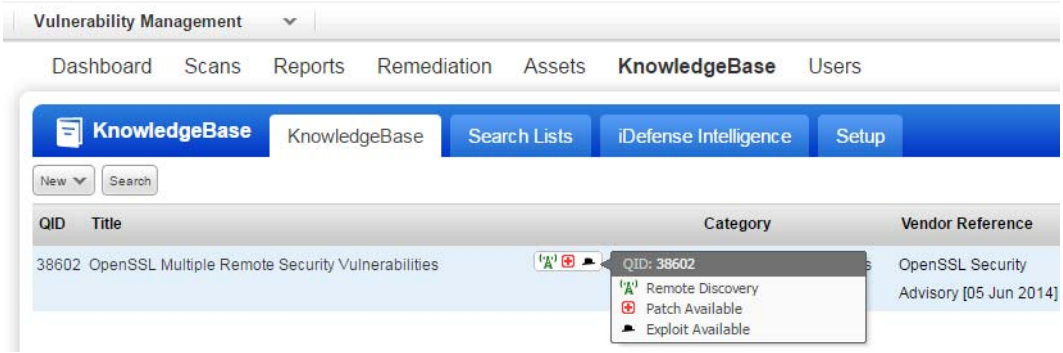
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Scan Results	05/10/2016 at 07:33:56AM (GMT+0000)													
2	Qualys, Inc.	1600 Bridge Pkwy	2nd Floor	Redwood City	California	United States	94065		Scan Summary						
3	Patrick Slimmer	qualys_ps	Manager												
4															
5	Launch Date	Active Hosts	Total Hosts	Type	Status	Reference	Scanner Appliance	Duration	Scan Title	Asset Groups	IPs	Excluded IPs	Option Profile		
6	5/10/2016 7:06	35	63	On Demand	Finished	scan/1462863	10.10.21.160 (Scanr	0:22:38	My Scan		10.10.10.23,10.10.10.28,		Initial Options		
7															
8	IP	DNS	NetBIOS	OS	IP Status	QID	Title	Type	Severity	Port	Protocol	FQDN	SSL	CVE ID	Vendor
9	10.10.10.28	xpsp3-10-28	XPSP3-10-28	Windows XP Serv	host scanned, found vuln	70000	NetBIOS Name Acc	Vuln	2				no		
10	10.10.10.28	xpsp3-10-28	XPSP3-10-28	Windows XP Serv	host scanned, found vuln	45002	Global User List	Vuln	2				no		
11	10.10.24.72	2k3x64sp2-24-72	2K3X64SP2-2	Windows Server 2	host scanned, found vuln	45002	Global User List	Vuln	2				no		
12	10.10.24.73	2k3x64sp2-ie6-p.2	2K3X64SP2-II	Windows 2003 Se	host scanned, found vuln	70000	NetBIOS Name Acc	Vuln	2				no		
13	10.10.24.73	2k3x64sp2-ie6-p.2	2K3X64SP2-II	Windows 2003 Se	host scanned, found vuln	45002	Global User List	Vuln	2				no		

Scan Results

VM - KnowledgeBase Download returns Remote Discovery, Patch and Exploit Available in CSV, XML

The KnowlegeBase Download option (New > Download), now returns this QID information when applicable - Remote Discovery (i.e. QID can be exploited through remote discovery), Patch Available, and Exploit Available in XML and CSV formats. For these formats we've added a new column Sub Category. No changes were made to datalist.dtd.

QID 38602 has the attributes Remote Discovery, Patch Available, Exploit Available. When you download this QID in XML or CSV format you'll see all 3 attributes.



Sample XML:

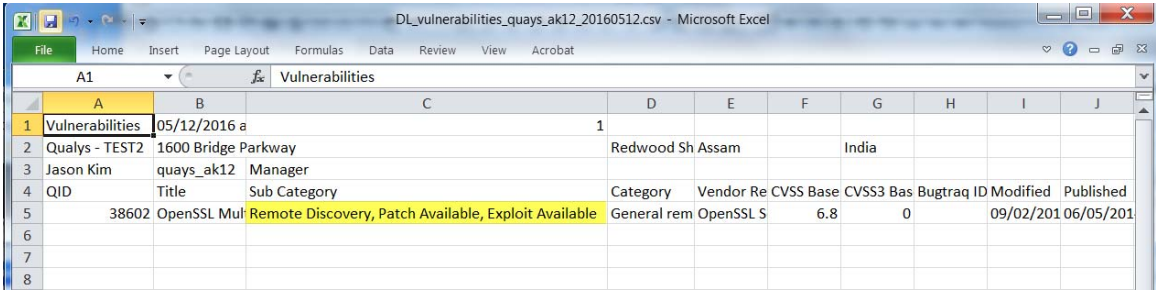
For QID 38602 the SUB_CATEGORY attribute shows Remote Discovery, Patch Available and Exploit Available.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE DATALIST SYSTEM "https://qualysguard.qualys.com/datalist.dtd">
<DATALIST>
  <HEADER>
    <NAME>Vulnerabilities</NAME>
    <GENERATION_DATETIME>04/27/2016 at 08:47:07PM
(GMT+0000)</GENERATION_DATETIME>
    <COMPANY_INFO>
...
  </HEADER>
  <LIST>
    <NB_RECORDS>1</NB_RECORDS>
    <RECORD>
      <KEY name="QID"><![CDATA[38602]]></KEY>
      <KEY name="TITLE"><![CDATA[OpenSSL Multiple Remote Security
Vulnerabilities]]></KEY>
      <KEY name="SUB_CATEGORY"><![CDATA[Remote Discovery, Patch
Available, Exploit Available]]></KEY>
```

```
        <KEY name="CATEGORY"><![CDATA[General remote services]]></KEY>
...
```

Sample CSV:

For QID 38602 the Sub Category column shows Remote Discovery, Patch Available and Exploit Available.

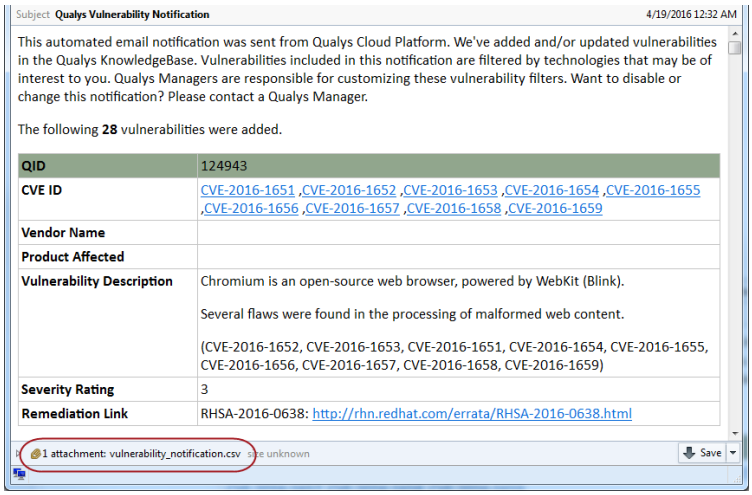


	A	B	C	D	E	F	G	H	I	J
1	Vulnerabilities	05/12/2016 a	1							
2	Qualys - TEST2	1600 Bridge Parkway		Redwood Sh	Assam		India			
3	Jason Kim	quays_ak12	Manager							
4	QID	Title	Sub Category	Category	Vendor Re	CVSS Base	CVSS3 Bas	Bugtraq ID	Modified	Published
5	38602	OpenSSL Mul	Remote Discovery, Patch Available, Exploit Available	General rem	OpenSSL S	6.8	0		09/02/201	06/05/201
6										
7										
8										

VM - Vulnerability Notification shows more QID attributes in CSV

The Vulnerability Notification email is sent from the Qualys Cloud Platform when we've added and/or updated vulnerabilities in the Qualys KnowledgeBase. We've added the attributes Remediation Link and Product Affected to the attached CSV file.

Vulnerability Notification email (no changes):



Subject: **Qualys Vulnerability Notification** 4/19/2016 12:32 AM

This automated email notification was sent from Qualys Cloud Platform. We've added and/or updated vulnerabilities in the Qualys KnowledgeBase. Vulnerabilities included in this notification are filtered by technologies that may be of interest to you. Qualys Managers are responsible for customizing these vulnerability filters. Want to disable or change this notification? Please contact a Qualys Manager.

The following **28** vulnerabilities were added.

QID	124943
CVE ID	CVE-2016-1651 ,CVE-2016-1652 ,CVE-2016-1653 ,CVE-2016-1654 ,CVE-2016-1655 ,CVE-2016-1656 ,CVE-2016-1657 ,CVE-2016-1658 ,CVE-2016-1659
Vendor Name	
Product Affected	
Vulnerability Description	Chromium is an open-source web browser, powered by WebKit (Blink). Several flaws were found in the processing of malformed web content. (CVE-2016-1652, CVE-2016-1653, CVE-2016-1651, CVE-2016-1654, CVE-2016-1655, CVE-2016-1656, CVE-2016-1657, CVE-2016-1658, CVE-2016-1659)
Severity Rating	3
Remediation Link	RHSA-2016-0638: http://rhn.redhat.com/errata/RHSA-2016-0638.html

1 attachment: vulnerability_notification.csv Save

CSV file: This shows the QID Remediation Link and Product Affected when available.

	A	B	C	D	E	F	G	H	I	J
1	STATUS	SEVERITY	QID	CVE	TITLE	IMPACT	THREAT	REMEDIATION LINK	PRODUCT AFFECTED	
2	UPDATED	4	38304	CVE-2001-	SSH Proto	The consequences of vulnerabilities present in SSH Version 1 include: <UL> <LI>SSH protected traffic compromise <LI>root shell access to the system running SSH server </UL>	SSH1 protocol was deprecated due to multiple vulnerabilities and design flaws. Among multiple vulnerabilities that exist in SSH protocol Version 1 are: <UL> <LI>a CRC32 compensation attack detector vulnerability (buffer overflow) <LI>an unauthorized session key recovery problem </UL> Multiple vendors' implementations are vulnerable due to the fact that these are protocol design errors. Version 2 of the SSH protocol fixed these errors. <P> Please refer to the following URL for more information: <A HREF="http://www.kb.cert.org/vuls/id/684820" TARGET="_blank">http://www.kb.cert.org/vuls/id/684820</A>		ssh	
3										
4										
5										

VM - Map Report Output shows network ID for IPs

Have the networks feature enabled? If yes you'll notice we updated the map report XML output when you launch and then fetch a map report using the Report APIv2 (/api/2.0/fo/report/). Now the XML output shows the network ID for each host IP. The map_report.dtd was updated.

Launch Report API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: cmdline curl" -d
"action=launch&template_id=15963512353&output_format=xml&report_refs=map/
1461278788.22279&domain=none"
"https://qualysapi.qualys.com/api/2.0/fo/report/"
```

Sample XML:

```
<?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>2016-05-07T01:32:55Z</DATETIME>
    <TEXT>New report launched</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>ID</KEY>
        <VALUE>24591432279</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

Fetch Report API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: cmdline curl" -d
"action=fetch&echo_request=1&id=24591432279"
"https://qualysapi.qualys.com/api/2.0/fo/report/"
```

Sample XML:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE MAPREPORT SYSTEM "https://qualysapi.qualys.com/map_report.dtd">
<MAPREPORT>
  <HEADER>
...
  <HOST_LIST>
    <HOST>
      <IP network_id="31850018">10.10.0.10</IP>
      <HOSTNAME><![CDATA[test1.test.domain.com]]></HOSTNAME>
```

```
      <NETBIOS><![CDATA[]]></NETBIOS>
      <ROUTER>10.11.51.2</ROUTER>
      <OS></OS>
      <APPROVED>0</APPROVED>
      <SCANNABLE>0</SCANNABLE>
      <IN_NETBLOCK>0</IN_NETBLOCK>
      <LIVE>1</LIVE>
      <ASSET_GROUPS />
      <AUTHENTICATION_RECORDS />
      <LAST_SCAN_DATE>N/A</LAST_SCAN_DATE>
    </HOST>
    <HOST>
      <IP network_id="31850018">10.10.10.3</IP>
      <HOSTNAME><![CDATA[test2.test.domain.com]]></HOSTNAME>
      <NETBIOS><![CDATA[]]></NETBIOS>
      <ROUTER></ROUTER>
      <OS></OS>
      <APPROVED>0</APPROVED>
      <SCANNABLE>1</SCANNABLE>
      <IN_NETBLOCK>1</IN_NETBLOCK>
      <LIVE>0</LIVE>
...
```

DTD update (map_report.dtd):

We added the network_id attribute to the IP element.

```
...
<!ELEMENT HOST (IP, HOSTNAME, NETBIOS, ROUTER, OS, APPROVED?, SCANNABLE?,
IN_NETBLOCK?, LIVE?, DISCOVERY_LIST?, ASSET_GROUPS?,
AUTHENTICATION_RECORDS?, HOST_STATUS?, LAST_SCAN_DATE?)>
<!ELEMENT IP (#PCDATA)>
   <!ATTLIST IP network_id CDATA #IMPLIED>
...
```

PC - New Oracle WebLogic Server Authentication API

The Oracle WebLogic Server Authentication API (/api/2.0/fo/auth/oracle_weblogic) lets you list, create, update and delete Oracle WebLogic Server authentication records.

Good to Know

- The Oracle WebLogic Server record type is only available in accounts with PC (Policy Compliance) and is only supported for compliance scans.
- We support these technologies: Oracle WebLogic Server 11g and Oracle WebLogic Server 12c
- Unix authentication is required so you'll need a Unix record for each host running an Oracle WebLogic Server.
- User permissions for this API are the same as other authentication record APIs. Want to know more? No problem, check out the Qualys API v2 User Guide - just log in to your account and go to Help > Resources.

List Oracle WebLogic Server authentication records

Use these parameters:

Parameter	Description
action=list	(Required) GET or POST method may be used.
ids={value}	(Optional) One or more IDs for Oracle WebLogic Server records. Multiple IDS are comma separated.
id_min={value}	(Optional) Show Oracle WebLogic Server authentication records with IDs greater than or equal to the record ID you specify.
id_max={value}	(Optional) Show Oracle WebLogic Server records with IDs less than or equal to the record ID you specify.
title={value}	(Optional) Show records with a certain string in the record's title.
comments={value}	(Optional) Show records with a certain string in the record's comments.
details=All | **`Basic`** | None	(Optional) Show the requested amount of information about each record. A valid value is: Basic (default) - show the record ID and all record attributes None - show the record ID only All - show the record ID, all record attributes, and glossary with owner's name and login

API request:

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

XML output:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE AUTH_ORACLE_WEBLOGIC_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/auth/oracle_weblogic/auth_oracle
_weblogic_list_output.dtd">
<AUTH_ORACLE_WEBLOGIC_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2016-05-10T13:28:19Z</DATETIME>
    <AUTH_ORACLE_WEBLOGIC_LIST>
      <AUTH_ORACLE_WEBLOGIC>
        <ID>2707382279</ID>
        <TITLE><![CDATA[MY_ORA_WEB_RECORD]]></TITLE>
        <IP_SET>
          <IP>10.10.10.28</IP>
        </IP_SET>
        <INSTALLATION_PATH>/u01/app/oracle/middleware</INSTALLATION_PATH>
        <AUTO_DISCOVER>1</AUTO_DISCOVER>
        <CREATED>
          <DATETIME>2016-05-10T05:23:34Z</DATETIME>
          <BY>qualys_user</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2016-05-10T05:23:34Z</DATETIME>
        </LAST_MODIFIED>
      </AUTH_ORACLE_WEBLOGIC>
    </AUTH_ORACLE_WEBLOGIC_LIST>
  </RESPONSE>
</AUTH_ORACLE_WEBLOGIC_LIST_OUTPUT>
```

DTD:

```
<!-- QUALYS AUTH_ORACLE_WEBLOGIC_LIST_OUTPUT DTD -->
<!ELEMENT AUTH_ORACLE_WEBLOGIC_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_ORACLE_WEBLOGIC_LIST|ID_SET)?,
                    WARNING_LIST?, GLOSSARY?)>
<!ELEMENT AUTH_ORACLE_WEBLOGIC_LIST (AUTH_ORACLE_WEBLOGIC+)>

<!ELEMENT AUTH_ORACLE_WEBLOGIC (ID, TITLE, IP_SET, INSTALLATION_PATH,
                                AUTO_DISCOVER, DOMAIN?, NETWORK_ID?,
                                CREATED, LAST_MODIFIED, COMMENTS?)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT INSTALLATION_PATH (#PCDATA)>
<!ELEMENT DOMAIN (#PCDATA)>
<!ELEMENT AUTO_DISCOVER (#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 Oracle WebLogic Server authentication records

Use these parameters:

Parameter	Description
action=create|update	(Required) POST method may be used.
title={value}	(Required for create request) The title for the new Oracle WebLogic Server record. The title must be unique and must contain 255 characters (ascii).
installation_path={value}	(Required for create request) The directory where the Oracle WebLogic Server is installed (i.e. Home directory). Example: /u01/app/oracle/middleware
auto_discover={0|**1**}	(Optional) When not specified we will use auto discovery to find all domains for you. Specify **auto_discover=0** and we will not auto discover domains.
	auto_discover=0 must be specified with the **domain** parameter in the same request.
domain={value}	(Optional) A single Oracle WebLogic Server domain name. Example: website
	The **domain** parameter must be specified with **auto_discover=0** in the same request.
ips={value}	(Required for create request) The IP address(es) for the Unix hosts where Oracle WebLogic servers are installed. Multiple entries are comma separated. (Optional for update request) IPs specified will overwrite existing IPs in the record, and existing IPs will be removed.
ids={value}	(Required for update request; invalid for create request) The IDs of the Oracle WebLogic Server authentication records that you want to update. Multiple IDs are comma separated
add_ips={value}	(Optional for update request; invalid for create request) Add IPs to the IPs list for this record. Multiple IPs/ranges are comma separated.
remove_ips={value}	(Optional for update request; invalid for create request) Remove IPs from the IPs list for this record. Multiple IPs/ranges are comma separated.

Parameter	Description
comments={value}	(Optional) User-defined comments. The comments may include a maximum of 1999 characters (ascii); if comments have 2000 or more characters an error is returned and comments are not saved. Tags (such as <script>) cannot be included.
network_id={value}	(Optional and valid when the networks feature is enabled) The network ID for the record.

API request (create record without Auto Discover):

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -X "POST" -d
"action=create&installation_path=/u01/app/oracle&auto_discover=0&domain=w
ww.qualys.com&ips=10.10.10.23&title=WEB_ORA_CREATE"
"https://qualysapi.qualys.com/api/2.0/fo/auth/oracle_weblogic/"
```

XML output:

```
<!DOCTYPE BATCH_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
  <RESPONSE>
    <DATETIME>2016-05-10T13:30:49Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>2707632279</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

API request (create record with Auto Discover):

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -X "POST" -d
"action=create&installation_path=/u01/app/oracle&auto_discover=1&ips=10.1
0.10.23&title=ABC_ORA"
"https://qualysapi.qualys.com/api/2.0/fo/auth/oracle_weblogic/"
```

XML output:

```
<!DOCTYPE BATCH_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
  <RESPONSE>
```

```
    <DATETIME>2016-05-10T13:42:46Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Created</TEXT>
        <ID_SET>
          <ID>2707642279</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

API request (update record):

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -X "POST" -d
"action=update&ids=2707632279&installation_path=/u01/app/oracle/update&au
to_discover=0&domain=www.qualystest.com&ips=10.10.10.23"
"https://qualysapi.qualys.com/api/2.0/fo/auth/oracle_weblogic/"
```

XML output:

```
<!DOCTYPE BATCH_RETURN SYSTEM
"https://qualysapi.qualys.com/api/2.0/batch_return.dtd">
<BATCH_RETURN>
  <RESPONSE>
    <DATETIME>2016-05-10T13:32:36Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Updated</TEXT>
        <ID_SET>
          <ID>2707632279</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Delete Oracle WebLogic Server authentication records

Use these parameters:

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

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: Curl" -X "POST" -d
"action=delete&ids=2707632279"
"https://qualysapi.qualys.com/api/2.0/fo/auth/oracle_weblogic/"
```

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>2016-05-10T13:33:41Z</DATETIME>
    <BATCH_LIST>
      <BATCH>
        <TEXT>Successfully Deleted</TEXT>
        <ID_SET>
          <ID>2707632279</ID>
        </ID_SET>
      </BATCH>
    </BATCH_LIST>
  </RESPONSE>
</BATCH_RETURN>
```

Authentication Records List

We added Oracle WebLogic Server record IDs to the XML output returned by the Authentication Record List API v2 (/api/2.0/fo/auth/?action=list).

API request:

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

XML output:

```
...
<AUTH_RECORDS_OUTPUT>
  <RESPONSE>
    <DATETIME>2015-06-25T17:05:08Z</DATETIME>
    <AUTH_RECORDS>
      <AUTH_UNIX_IDS>
        <ID_SET>
          <ID>2143096540</ID>
        </ID_SET>
      </AUTH_UNIX_IDS>
      <AUTH_ORACLE_WEBLOGIC_IDS>
        <ID_SET>
          <ID>2135896540</ID>
```

```
          <ID>2135906540</ID>
          <ID>2136086540</ID>
        </ID_SET>
      </AUTH_ORACLE_WEBLOGIC_IDS>
    </AUTH_RECORDS>
  </RESPONSE>
</AUTH_RECORDS_OUTPUT>
```

DTD update:

We added the AUTH_ORACLE_WEBLOGIC_IDS element to the authentication record list output DTD (/api/2.0/fo/auth/auth_records.dtd).

```
...
<!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_MYSQL_IDS?, AUTH_TOMCAT_IDS?,
                        AUTH_ORACLE_WEBLOGIC_IDS?)>
...
<!ELEMENT AUTH_ORACLE_WEBLOGIC_IDS (ID_SET)>
...
```

PC - Unix Authentication Supports CheckPoint Firewall Sub-Type

You can now create and manage authentication records for CheckPoint Firewall. You'll use the Unix authentication resource (/api/2.0/fo/auth/unix) with the new sub-type "checkpoint_firewall". This works in the same way as Cisco IOS, which is also a sub-type of Unix authentication.

Good to Know

- The Checkpoint Firewall record type is only available in accounts with PC (Policy Compliance) and is only supported for compliance scans.
- An IP address in the Checkpoint Firewall record cannot also exist in Unix or Cisco IOS records.
- User permissions for this API are the same as other authentication record APIs. Want to know more? No problem, check out the Qualys API v2 User Guide - just log in to your account and go to Help > Resources.

Create / Update Checkpoint Firewall authentication records

Use these parameters:

Parameter	Description
action=create|update	(Required) POST method may be used.
sub_type={value}	(Required for Cisco IOS and Checkpoint Firewall records; Invalid for Unix record) To create a Cisco IOS record, specify **sub_type=cisco**. To create a Checkpoint Firewall record, specify **sub_type=checkpoint_firewall**.
ids={value}	(Required for update request; Invalid for create request) Update only authentication records with certain IDs and/or ID ranges. Multiple entries are comma separated. One or more IDs/ranges may be specified. An ID range entry is specified with a hyphen (for example, 1359-1407). Valid IDs are required.
echo_request={**0**|1}	(Optional) Show (echo) the request's input parameters (names and values) in the XML output. When not specified, parameters are not included in the XML output. Specify 1 to include parameters in the XML output.
title={value}	(Required for create request) Specifies a title for the authentication record. The title must be unique and may include a maximum of 255 characters (ascii).

Parameter	Description
comments={value}	(Optional) Specifies user defined notes about the authentication record. The comments may include a maximum of 1999 characters (ascii); if comments have 2000 or more characters an error is returned and comments are not saved. Tags (such as <script>) cannot be included; if tags are included an error is returned and the request fails.
{login credentials}	Define login credentials for authentication to target hosts.
enable_password={value}	(Optional for Cisco IOS record; Invalid for Unix and Checkpoint Firewall records) The password required for executing the "enable" command on the target hosts. The password may include 1-31 characters (ascii). Note: The pooled credentials feature is not supported if the "enable" command requires a password and it is specified using the **`enable_password`** parameter.
expert_password={value}	(Optional for Checkpoint Firewall record; Invalid for Unix and Cisco IOS records) The password required for executing the "expert" command on the target hosts. The password may include 1-31 characters (ascii).
port={value}	(Optional for compliance scanning: Invalid for vulnerability scanning) Specifies custom ports to be used to perform authentication and compliance assessment (control testing).
{target hosts}	Define the target hosts for authentication.
use_agentless_tracking=[**`0`**|1}	Specify "1" to enable Agentless Tracking.
agentless_tracking_path={value}	(Valid for a Unix record only). The pathname where you would like the service to store the host ID file on each host. This is required to enable Agentless Tracking for a Unix record; this is not valid for a Windows record.
network_id={value}	(Optional and valid when the networks feature is enabled) The network ID for the record.

For complete information on login credentials, custom port settings and target hosts, please refer to the Qualys API V2 User Guide. Log in to your account and go to Help > Resources to download it.

Samples

API request (create new Checkpoint Firewall record):

```
curl -H "X-Requested-With: curl" -u "USERNAME:PASSWORD" -X "POST"
"https://qualysapi.qualys.com/api/2.0/fo/auth/unix/?action=create&sub
_type=checkpoint_firewall&title=My+Checkpoint+Firewall+Record&usernam
e=John&password=abc12345&cleartext_password=1&expert_password=chocola
te&ips=10.10.10.10-10.10.10.12"
```

XML output:

```
<BATCH_RETURN>
    <RESPONSE>
        <DATETIME>2015-06-03T17:08:34Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Created</TEXT>
                <ID_SET>
                    <ID>137296922</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

API request (update record for any Unix-type):

```
curl -H "X-Requested-With: curl" -u "USERNAME:PASSWORD" -X "POST"
"https://qualysapi.qualys.com/api/2.0/fo/auth/unix/?action=update&ids
=2136616540&add_ips=10.10.10.200-10.10.10.240"
```

XML output:

```
<BATCH_RETURN>
    <RESPONSE>
        <DATETIME>2015-06-03T17:14:28Z</DATETIME>
        <BATCH_LIST>
            <BATCH>
                <TEXT>Successfully Updated</TEXT>
                <ID_SET>
                    <ID>2136616540</ID>
                </ID_SET>
            </BATCH>
        </BATCH_LIST>
    </RESPONSE>
</BATCH_RETURN>
```

PC - Exception API - Support for Truncation Limit

The Exception API (/api/2.0/fo/compliance/exception/) now supports truncation limit and also provides options to filter exceptions by exception number.

Parameter	Description
action=list	(Required) GET or POST method may be used.
exception_numbers= {value}	(Optional) Show a specific exception by specifying a valid exception number. Multiple entries are comma separated. An exception number range is specified with a hyphen (for example, 289-292).
exception_number_min= {value}	(Optional) Show only exceptions that have a exception number greater than or equal to the specified value.
exception_number_max= {value}	(Optional) Show only exceptions that have exception number less than or equal to the specified value.
truncation_limit={value}	(Optional) Specify the maximum number of exceptions to be listed per request. When not specified, the truncation limit is set to 1000 records. You may specify a value less than the default (1-999) or greater than the default (1001-1000000).

Sample Requests

Sample 1: Let us restrict the output (number of exceptions displayed) using the truncation_limit parameter to 2.

API request:

```
curl -H -u "USERNAME:PASSWORD" Curl Sample "X-Requested-With:"
"https://qualysapi.qualys.com/api/2.0/fo/compliance/exception/?action=lis
t&truncation_limit=2"
```

XML output:

```
<!DOCTYPE EXCEPTION_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/compliance/exception/exception_l
ist_output.dtd">
<EXCEPTION_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2016-04-25T11:31:24Z</DATETIME>
    <!-- keep-alive for EXCEPTION_LIST_OUTPUT  -->
    <EXCEPTION_LIST>
      <EXCEPTION>
        <EXCEPTION_NUMBER>412</EXCEPTION_NUMBER>
        <HOST>
          <IP_ADDRESS>10.10.26.26</IP_ADDRESS>
```

```
    </HOST>
    <TECHNOLOGY>
      <ID>45</ID>
      <NAME><![CDATA[Red Hat Enterprise Linux 6.x]]></NAME>
    </TECHNOLOGY>
    <POLICY>
      <ID>211210</ID>
      <NAME><![CDATA[RHEL 6.x _merge API test]]></NAME>
    </POLICY>
    <CONTROL>
      <CID>1071</CID>
      <STATEMENT><![CDATA[Status of the 'Minimum Password Length'
      setting]]></STATEMENT>
      <CRITICALITY>
        <VALUE>5</VALUE>
        <LABEL><![CDATA[URGENT]]></LABEL>
      </CRITICALITY>
    </CONTROL>
    <ASSIGNEE><![CDATA[Reader User]]></ASSIGNEE>
    <STATUS>Rejected</STATUS>
    <ACTIVE>0</ACTIVE>
    <REOPEN_ON_EVIDENCE_CHANGE>1</REOPEN_ON_EVIDENCE_CHANGE>
    <EXPIRATION_DATE>N/A</EXPIRATION_DATE>
    <MODIFIED_DATE>2016-02-16T10:26:49Z</MODIFIED_DATE>
  </EXCEPTION>
  <EXCEPTION>
    <EXCEPTION_NUMBER>413</EXCEPTION_NUMBER>
    <HOST>
      <IP_ADDRESS>10.10.26.26</IP_ADDRESS>
    </HOST>
    <TECHNOLOGY>
      <ID>45</ID>
      <NAME><![CDATA[Red Hat Enterprise Linux 6.x]]></NAME>
    </TECHNOLOGY>
    <POLICY>
      <ID>211210</ID>
      <NAME><![CDATA[RHEL 6.x _merge API test]]></NAME>
    </POLICY>
    <CONTROL>
      <CID>1072</CID>
      <STATEMENT><![CDATA[Status of the 'Minimum Password Age'
      setting]]></STATEMENT>
      <CRITICALITY>
        <VALUE>5</VALUE>
        <LABEL><![CDATA[URGENT]]></LABEL>
      </CRITICALITY>
    </CONTROL>
    <ASSIGNEE><![CDATA[Unit Manager Manager]]></ASSIGNEE>
    <STATUS>Expired</STATUS>
```

```
        <ACTIVE>0</ACTIVE>
        <REOPEN_ON_EVIDENCE_CHANGE>0</REOPEN_ON_EVIDENCE_CHANGE>
        <EXPIRATION_DATE>2016-02-16T18:30:00Z</EXPIRATION_DATE>
        <MODIFIED_DATE>2016-02-16T10:34:21Z</MODIFIED_DATE>
      </EXCEPTION>
    </EXCEPTION_LIST>
    <WARNING>
      <CODE>1980</CODE>
      <TEXT>2 record limit exceeded. Use URL to get next batch of
      results.</TEXT>
<URL><![CDATA[https://qualysapi.qualys.com/api/2.0/fo/compliance/
 exception/?action=list&truncation_limit=2&exception_number_min=414]]>
 </URL>
    </WARNING>
  </RESPONSE>
</EXCEPTION_LIST_OUTPUT>
```

Sample 2: Show only exceptions that have a exception number greater than or equal to 414.

API request:

```
curl -H -u "USERNAME:PASSWORD" Curl Sample"X-Requested-With:"
"https://qualysapi.qualys.com/api/2.0/fo/compliance/exception/?action=lis
t&truncation_limit=2&exception_number_min=414"
```

XML output:

```
<!DOCTYPE EXCEPTION_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/compliance/exception/exception_l
ist_output.dtd">
<EXCEPTION_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2016-04-25T11:34:28Z</DATETIME>
    <!-- keep-alive for EXCEPTION_LIST_OUTPUT  -->
    <EXCEPTION_LIST>
      <EXCEPTION>
        <EXCEPTION_NUMBER>414</EXCEPTION_NUMBER>
        <HOST>
          <IP_ADDRESS>10.10.26.26</IP_ADDRESS>
        </HOST>
        <TECHNOLOGY>
          <ID>45</ID>
          <NAME><![CDATA[Red Hat Enterprise Linux 6.x]]></NAME>
        </TECHNOLOGY>
        <POLICY>
          <ID>211210</ID>
          <NAME><![CDATA[RHEL 6.x _merge API test]]></NAME>
        </POLICY>
        <CONTROL>
```

```
        <CID>1073</CID>
        <STATEMENT><![CDATA[Status of the 'Maximum Password Age' setting
        (expiration) / Accounts having the 'password never expires' flag
        set]]></STATEMENT>
        <CRITICALITY>
          <VALUE>5</VALUE>
          <LABEL><![CDATA[URGENT]]></LABEL>
        </CRITICALITY>
      </CONTROL>
      <ASSIGNEE><![CDATA[Auditor User]]></ASSIGNEE>
      <STATUS>Approved</STATUS>
      <ACTIVE>0</ACTIVE>
      <REOPEN_ON_EVIDENCE_CHANGE>0</REOPEN_ON_EVIDENCE_CHANGE>
      <EXPIRATION_DATE>N/A</EXPIRATION_DATE>
      <MODIFIED_DATE>2016-02-16T10:26:49Z</MODIFIED_DATE>
    </EXCEPTION>
    <EXCEPTION>
      <EXCEPTION_NUMBER>423</EXCEPTION_NUMBER>
      <HOST>
        <IP_ADDRESS>10.10.10.28</IP_ADDRESS>
      </HOST>
      <TECHNOLOGY>
        <ID>1</ID>
        <NAME><![CDATA[Windows XP desktop]]></NAME>
      </TECHNOLOGY>
      <POLICY>
        <ID>200271</ID>
        <NAME><![CDATA[All Technology + All Ag policy]]></NAME>
      </POLICY>
      <CONTROL>
        <CID>1073</CID>
        <STATEMENT><![CDATA[Status of the 'Maximum Password Age' setting
        (expiration) / Accounts having the 'password never expires' flag
        set]]></STATEMENT>
        <CRITICALITY>
          <VALUE>5</VALUE>
          <LABEL><![CDATA[URGENT]]></LABEL>
        </CRITICALITY>
      </CONTROL>
      <ASSIGNEE><![CDATA[Auditor User]]></ASSIGNEE>
      <STATUS>Approved</STATUS>
      <ACTIVE>1</ACTIVE>
      <REOPEN_ON_EVIDENCE_CHANGE>1</REOPEN_ON_EVIDENCE_CHANGE>
      <EXPIRATION_DATE>2016-08-03T00:00:00Z</EXPIRATION_DATE>
      <MODIFIED_DATE>2016-04-25T08:19:38Z</MODIFIED_DATE>
    </EXCEPTION>
  </EXCEPTION_LIST>
  <WARNING>
    <CODE>1980</CODE>
```

```
      <TEXT>2 record limit exceeded. Use URL to get next batch of
      results.</TEXT>
<URL><![CDATA[https://qualysapi.qualys.com/api/2.0/fo/compliance/
exception/?action=list&truncation_limit=2&exception_number_min=425]]>
</URL>
    </WARNING>
  </RESPONSE>
</EXCEPTION_LIST_OUTPUT>
```

Sample 3: Show only exceptions that have a exception number greater than 414 and equal to 414,415, and 416.

API request:

```
curl -H -u "USERNAME:PASSWORD" Curl Sample "X-Requested-With:"
"https://qualysapi.qualys.com/api/2.0/fo/compliance/exception/?action=lis
t&truncation_limit=2&exception_number_min=414&exception_numbers=414,415,4
16"
```

XML output:

```
<!DOCTYPE EXCEPTION_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/compliance/exception/exception_l
ist_output.dtd">
<EXCEPTION_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2016-04-25T11:37:02Z</DATETIME>
    <EXCEPTION_LIST>
      <EXCEPTION>
        <EXCEPTION_NUMBER>414</EXCEPTION_NUMBER>
        <HOST>
          <IP_ADDRESS>10.10.26.26</IP_ADDRESS>
        </HOST>
        <TECHNOLOGY>
          <ID>45</ID>
          <NAME><![CDATA[Red Hat Enterprise Linux 6.x]]></NAME>
        </TECHNOLOGY>
        <POLICY>
          <ID>211210</ID>
          <NAME><![CDATA[RHEL 6.x _merge API test]]></NAME>
        </POLICY>
        <CONTROL>
          <CID>1073</CID>
          <STATEMENT><![CDATA[Status of the 'Maximum Password Age' setting
          (expiration) / Accounts having the 'password never expires' flag
          set]]></STATEMENT>
          <CRITICALITY>
            <VALUE>5</VALUE>
            <LABEL><![CDATA[URGENT]]></LABEL>
          </CRITICALITY>
```

```
        </CONTROL>
        <ASSIGNEE><![CDATA[Auditor User]]></ASSIGNEE>
        <STATUS>Approved</STATUS>
        <ACTIVE>0</ACTIVE>
        <REOPEN_ON_EVIDENCE_CHANGE>0</REOPEN_ON_EVIDENCE_CHANGE>
        <EXPIRATION_DATE>N/A</EXPIRATION_DATE>
        <MODIFIED_DATE>2016-02-16T10:26:49Z</MODIFIED_DATE>
      </EXCEPTION>
    </EXCEPTION_LIST>
  </RESPONSE>
</EXCEPTION_LIST_OUTPUT>
```

Sample 4:You can use multiple criteria such as show only exceptions that have a exception number greater than 414 and equal to 414,415, and 416, but not greater than 450.

API request:

```
curl -H -u "USERNAME:PASSWORD" Curl Sample "X-Requested-With: "
"https://qualysapi.qualys.com/api/2.0/fo/compliance/exception/?action=lis
t&truncation_limit=2&exception_number_min=414&exception_numbers=414,415,4
16&exception_number_max=450"
```

XML output:

```
!DOCTYPE EXCEPTION_LIST_OUTPUT SYSTEM
"https://qualysapi.qualys.com/api/2.0/fo/compliance/exception/exception_l
ist_output.dtd">
<EXCEPTION_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2016-04-25T11:38:29Z</DATETIME>
    <EXCEPTION_LIST>
      <EXCEPTION>
        <EXCEPTION_NUMBER>414</EXCEPTION_NUMBER>
        <HOST>
          <IP_ADDRESS>10.10.26.26</IP_ADDRESS>
        </HOST>
        <TECHNOLOGY>
          <ID>45</ID>
          <NAME><![CDATA[Red Hat Enterprise Linux 6.x]]></NAME>
        </TECHNOLOGY>
        <POLICY>
          <ID>211210</ID>
          <NAME><![CDATA[RHEL 6.x _merge API test]]></NAME>
        </POLICY>
        <CONTROL>
          <CID>1073</CID>
          <STATEMENT><![CDATA[Status of the 'Maximum Password Age' setting
          (expiration) / Accounts having the 'password never expires' flag
```

```
            set]]></STATEMENT>
            <CRITICALITY>
              <VALUE>5</VALUE>
              <LABEL><![CDATA[URGENT]]></LABEL>
            </CRITICALITY>
          </CONTROL>
          <ASSIGNEE><![CDATA[Auditor User]]></ASSIGNEE>
          <STATUS>Approved</STATUS>
          <ACTIVE>0</ACTIVE>
          <REOPEN_ON_EVIDENCE_CHANGE>0</REOPEN_ON_EVIDENCE_CHANGE>
          <EXPIRATION_DATE>N/A</EXPIRATION_DATE>
          <MODIFIED_DATE>2016-02-16T10:26:49Z</MODIFIED_DATE>
        </EXCEPTION>
      </EXCEPTION_LIST>
    </RESPONSE>
  </EXCEPTION_LIST_OUTPUT>
```

PC - Support Agent IPs in Compliance Policy

Now you can report on agent host compliance by adding agent host IPs in your compliance policy. Managers and Auditors have permission to add agent IPs to policies, view and report on agent IPs.

Policy List

Updates to Policy List API (<platformURL>/api/2.0/fo/compliance/policy/)

Changes to XML output:

- new elements: POLICY/LAST_EVALUATED and POLICY/INCLUDE_AGENT_IPS

DTD update (policy_list_output.dtd)

```
...
<!ELEMENT POLICY (ID, TITLE, CREATED?, LAST_MODIFIED?, LAST_EVALUATED?,
STATUS?, ASSET_GROUP_IDS?, INCLUDE_AGENT_IPS?, CONTROL_LIST?)>
<!ELEMENT LAST_EVALUATED (DATETIME)>
<!ELEMENT INCLUDE_AGENT_IPS (#PCDATA)>
...
```

Sample request:

```
 curl -u "USERNAME:PASSWORD" -H "X-Requested-With: cmdline curl" -d
"action=list&ids=991742279"
"https://qualysapi.qualys.com/api/2.0/fo/compliance/policy/"
```

Sample XML:

```
<POLICY_LIST_OUTPUT>
  <RESPONSE>
    <DATETIME>2016-05-18T20:52:47Z</DATETIME>
    <POLICY_LIST>
      <POLICY>
        <ID>991742279</ID>
        <TITLE><![CDATA[Agent Enable for Windows 7]]></TITLE>
        <CREATED>
          <DATETIME>2016-05-12T18:20:04Z</DATETIME>
          <BY>acme_ab</BY>
        </CREATED>
        <LAST_MODIFIED>
          <DATETIME>2016-05-13T19:07:51Z</DATETIME>
          <BY>acme_ab</BY>
        </LAST_MODIFIED>
        <LAST_EVALUATED>
          <DATETIME>2016-05-17T11:23:26Z</DATETIME>
        </LAST_EVALUATED>
```

```
        <STATUS><![CDATA[active]]></STATUS>
        <INCLUDE_AGENT_IPS>1</INCLUDE_AGENT_IPS>
...
```

Exception List

Updates to Exception List API (<platformURL>/api/2.0/fo/compliance/exception/)

Changes to XML output:

- agent IPs are listed with tracking method AGENT if PC Agent license is enabled (Manager and Auditor only)

- new element: HOST/TRACKING_METHOD

DTD update (exception_list_output.dtd):

```
...
<!ELEMENT HOST (IP_ADDRESS, TRACKING_METHOD, NETWORK?)>
<!ELEMENT TRACKING_METHOD (#PCDATA)>
...
```

Sample request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: cmdline curl" -D
headers.15
"https://qualysapi.qualys.com/api/2.0/fo/compliance/exception/?action=lis
t&details=All"
```

Sample XML:

```
...
      </EXCEPTION>
      <EXCEPTION>
        <EXCEPTION_NUMBER>243</EXCEPTION_NUMBER>
        <HOST>
          <IP_ADDRESS>10.100.14.168</IP_ADDRESS>
          <TRACKING_METHOD>AGENT</TRACKING_METHOD>
        </HOST>
        <TECHNOLOGY>
          <ID>37</ID>
          <NAME><![CDATA[Windows 7]]></NAME>
        </TECHNOLOGY>
        <POLICY>
          <ID>991742279</ID>
          <NAME><![CDATA[Agent Enable for Windows 7]]></NAME>
        </POLICY>
        <CONTROL>
          <CID>1048</CID>
         <STATEMENT><![CDATA[Status of the 'Shutdown: Clear virtual memory
```

```
pagefile' setting]]></STATEMENT>
          <CRITICALITY>
            <VALUE>4</VALUE>
            <LABEL><![CDATA[CRITICAL]]></LABEL>
          </CRITICALITY>
        </CONTROL>
        <ASSIGNEE><![CDATA[Jill Smith]]></ASSIGNEE>
        <STATUS>Pending</STATUS>
        <ACTIVE>1</ACTIVE>
        <REOPEN_ON_EVIDENCE_CHANGE>0</REOPEN_ON_EVIDENCE_CHANGE>
        <EXPIRATION_DATE>N/A</EXPIRATION_DATE>
        <MODIFIED_DATE>2016-05-16T22:32:52Z</MODIFIED_DATE>
        <HISTORY_LIST>
          <HISTORY>
            <USER><![CDATA[Jill Smith (acme_js15)]]></USER>
            <COMMENT><![CDATA[exception comments]]></COMMENT>
            <INSERTION_DATE>2016-05-16T22:32:52Z</INSERTION_DATE>
          </HISTORY>
        </HISTORY_LIST>
      </EXCEPTION>
...
```

List Posture Info

Updates to List Posture Info API (<platformURL>/api/2.0/fo/compliance/posture/info/)

Request parameters to support host filtering (default setting in bold):

- New parameters: tag_set_by (id or name), tag_include_selector (**any** or all), tag_exclude_selector (**any** or all), tag_set_include, tag_set_exclude

Changes to XML output:

- agent IPs are listed with tracking method AGENT if PC Agent license is enabled (Manager and Auditor only)

- no changes to posture info list DTD (posture_info_list_output.dtd)

Sample request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: demo 2" -D headers.15
"https://qualysapi.qualys.com/api/2.0/fo/compliance/posture/info/?action=
list&policy_id=991742279&tag_set_by=name&tag_include_selector=all&tag_set
_include=Cloud_Agent"
```

Sample XML:

```
...
    <GLOSSARY>
      <HOST_LIST>
        <HOST>
          <ID>28762882279</ID>
          <IP>10.0.203.59</IP>
          <TRACKING_METHOD>AGENT</TRACKING_METHOD>
          <DNS><![CDATA[101854-t450]]></DNS>
          <NETBIOS><![CDATA[101854-T450]]></NETBIOS>
          <OS><![CDATA[Microsoft Windows 7 Professional 6.1.7601 Service
Pack 1 Build 7601]]></OS>
          <PERCENTAGE><![CDATA[75.00% (3 of 4)]]></PERCENTAGE>
        </HOST>
...
```

Policy Report

Updates to XML output:

- agent IPs are listed with tracking method AGENT if PC Agent license is enabled (Manager and Auditor only)

- no changes were made to compliance_policy_report.dtd

Sample XML:

```
...
  <RESULTS>
    <HOST_LIST>
      <HOST>
        <IP><![CDATA[10.10.2.67]]></IP>
        <TRACKING_METHOD><![CDATA[AGENT]]></TRACKING_METHOD>
        <DNS><![CDATA[100194-d9010]]></DNS>
        <NETBIOS><![CDATA[100194-D9010]]></NETBIOS>
        <OPERATING_SYSTEM><![CDATA[Microsoft Windows 7 Professional
6.1.7601 Service Pack 1 Build 7601]]></OPERATING_SYSTEM>
        <LAST_SCAN_DATE>2016-04-24T05:15:57Z</LAST_SCAN_DATE>
        <TOTAL_PASSED>908</TOTAL_PASSED>
        <TOTAL_FAILED>6</TOTAL_FAILED>
        <TOTAL_ERROR>32</TOTAL_ERROR>
        <TOTAL_EXCEPTIONS>1</TOTAL_EXCEPTIONS>
        <ASSET_TAGS>
          <ASSET_TAG><![CDATA[Business Units]]></ASSET_TAG>
          <ASSET_TAG><![CDATA[My AG]]></ASSET_TAG>
          <ASSET_TAG><![CDATA[Cloud Agent]]></ASSET_TAG>
        </ASSET_TAGS>
...
```

Compliance Scorecard Report

Updates to XML output:

- agent IPs are listed with tracking method AGENT if PC Agent license is enabled (Manager and Auditor only)

- element added: HOST/TRACKING_METHOD

DTD update (compliance_scorecard_report.dtd):

```
...
<!ELEMENT HOST (IP_ADDRESS, TRACKING_METHOD, NETBIOS, DNS, NETWORK?,
ASSET_GROUP_NAME?, ASSET_TAG_NAME?, TECHNOLOGY, NUMBER_OF_POLICIES,
PASSED_TOTAL?, PASSED_CHANGED?, FAILED_TOTAL?, FAILED_CHANGED?,
ERROR_TOTAL?, ERROR_CHANGED?, COMPLIANCE, NETWORK?)>
<!ELEMENT TRACKING_METHOD (#PCDATA)>
...
```

Sample XML:

```
...
      <CHANGED_TO_PASS>
        <HOST>
          <IP_ADDRESS><![CDATA[10.0.203.59]]></IP_ADDRESS>
          <TRACKING_METHOD><![CDATA[AGENT]]></TRACKING_METHOD>
          <NETBIOS><![CDATA[101854-T450]]></NETBIOS>
          <DNS><![CDATA[101854-t450]]></DNS>
          <ASSET_TAG_NAME><![CDATA[Cloud Agent]]></ASSET_TAG_NAME>
          <TECHNOLOGY>Windows 7</TECHNOLOGY>
          <NUMBER_OF_POLICIES>1</NUMBER_OF_POLICIES>
          <PASSED_TOTAL>3</PASSED_TOTAL>
          <PASSED_CHANGED>3</PASSED_CHANGED>
          <COMPLIANCE>100%</COMPLIANCE>
        </HOST>
...
```

Control Pass/Fail Report

Updates to XML output:

- agent IPs are included with tracking method AGENT if PC Agent license is enabled (Manager and Auditor only)

- element added: ASSET_TAGS

DTD update (control_pass_fail_report.dtd):

```
...
<!ELEMENT FILTERS (POLICY, CID, REFERENCE, CONTROL, CRITICALITY?,
ASSET_GROUP, ASSET_TAGS?, DISPLAY, SORT_BY, POLICY_MODIFIED)>
```

```
...
<!ELEMENT ASSET_TAGS (INCLUDED_TAG_SELECTOR*, INCLUDED_TAGS*,
EXCLUDED_TAG_SELECTOR*, EXCLUDED_TAGS*)>
<!ELEMENT INCLUDED_TAG_SELECTOR (#PCDATA)>
<!ELEMENT INCLUDED_TAGS (ASSET_TAG_NAME*)>
<!ELEMENT EXCLUDED_TAG_SELECTOR (#PCDATA)>
<!ELEMENT EXCLUDED_TAGS (ASSET_TAG_NAME*)>
<!ELEMENT ASSET_TAG_NAME (#PCDATA)>
...
```

Sample XML:

```
...
    <FILTERS>
      <POLICY><![CDATA[Agent Enable for Windows 7]]></POLICY>
      <CID>1048</CID>
      <REFERENCE><![CDATA[]]></REFERENCE>
      <CONTROL><![CDATA[Status of the 'Shutdown: Clear virtual memory
pagefile' setting]]></CONTROL>
      <CRITICALITY>
        <LABEL><![CDATA[CRITICAL]]></LABEL>
        <VALUE>4</VALUE>
      </CRITICALITY>
      <ASSET_GROUP>
        <TITLE><![CDATA[-]]></TITLE>
      </ASSET_GROUP>
      <ASSET_TAGS>
        <INCLUDED_TAG_SELECTOR><![CDATA[any]]></INCLUDED_TAG_SELECTOR>
        <INCLUDED_TAGS>
          <ASSET_TAG_NAME><![CDATA[Cloud Agent]]></ASSET_TAG_NAME>
        </INCLUDED_TAGS>
      </ASSET_TAGS>
      <DISPLAY><![CDATA[Passed, Failed and Error]]></DISPLAY>
      <SORT_BY><![CDATA[IP Address]]></SORT_BY>
      <POLICY_MODIFIED><![CDATA[N/A]]></POLICY_MODIFIED>
    </FILTERS>
...
```

Individual Host Compliance Report

Updates to XML output:

- agent IPs are included with tracking method AGENT if PC Agent license is enabled (Manager and Auditor only)

- element added ASSET_TAGS/ASSET_TAG

DTD update (individual_host_compliance_report.dtd):

```
...
<!ELEMENT FILTERS (POLICY, ASSET_GROUP, ASSET_TAGS?, IP_ADDRESS, DISPLAY,
CRITICALITY_FILTER?, SORT_BY, POLICY_MODIFIED)>
...
<!ELEMENT ASSET_TAGS (INCLUDED_TAG_SELECTOR*, INCLUDED_TAGS*,
EXCLUDED_TAG_SELECTOR*, EXCLUDED_TAGS*) >
<!ELEMENT INCLUDED_TAG_SELECTOR (#PCDATA)>
<!ELEMENT INCLUDED_TAGS (ASSET_TAG_NAME*)>
<!ELEMENT EXCLUDED_TAG_SELECTOR (#PCDATA)>
<!ELEMENT EXCLUDED_TAGS (ASSET_TAG_NAME*)>
<!ELEMENT ASSET_TAG_NAME (#PCDATA)>
...
```

Sample XML:

```
    <FILTERS>
      <POLICY><![CDATA[Agent Enable for Windows 7]]></POLICY>
      <ASSET_GROUP><![CDATA[]]></ASSET_GROUP>
      <ASSET_TAGS>
        <INCLUDED_TAGS>
          <ASSET_TAG_NAME><![CDATA[Cloud Agent]]></ASSET_TAG_NAME>
        </INCLUDED_TAGS>
      </ASSET_TAGS>
      <IP_ADDRESS><![CDATA[]]></IP_ADDRESS>
      <DISPLAY><![CDATA[Passed, Failed and Error]]></DISPLAY>
<CRITICALITY_FILTER><![CDATA[UNDEFINED,alert(&quot;xss&quot;),MEDIUM,SERI
OUS,CRITICAL,URGENT]]></CRITICALITY_FILTER>
      <SORT_BY><![CDATA[Order]]></SORT_BY>
      <POLICY_MODIFIED><![CDATA[05/13/2016 at 11:07:51 (GMT-
0800)]]></POLICY_MODIFIED>
    </FILTERS>
  </HEADER>
  <RESULTS>
    <TOTAL_CONTROLS>4</TOTAL_CONTROLS>
    <TOTAL_FAILED>1</TOTAL_FAILED>
    <PERCENTAGE_FAILED>(25%)</PERCENTAGE_FAILED>
    <TOTAL_PASSED>3</TOTAL_PASSED>
    <PERCENTAGE_PASSED>(75%)</PERCENTAGE_PASSED>
    <TOTAL_ERROR>0</TOTAL_ERROR>
    <PERCENTAGE_ERROR></PERCENTAGE_ERROR>
    <HOST>
      <TRACKING_METHOD><![CDATA[AGENT]]></TRACKING_METHOD>
      <IP><![CDATA[10.0.203.59]]></IP>
      <DNS><![CDATA[101854-t450]]></DNS>
      <NETBIOS><![CDATA[101854-T450]]></NETBIOS>
...
```

PC - Posture API Always Returns Status

The Posture API (/api/2.0/fo/compliance/posture/info/) returns posture info status in all cases. Previously the status was not returned when the default truncation limit was used.

Sample using default truncation limit

API request:

```
curl -u "USERNAME:PASSWORD" -H "X-Requested-With: curl demo" -d
"action=list&policy_id=2275&status=Error"
"https://qualysapi.qualys.com/api/2.0/fo/compliance/posture/info/"
```

XML output:

```
...
<INFO_LIST>
      <INFO>
        <ID>2973124</ID>
        <HOST_ID>1633074</HOST_ID>
        <CONTROL_ID>1151</CONTROL_ID>
        <TECHNOLOGY_ID>1</TECHNOLOGY_ID>
        <INSTANCE></INSTANCE>
        <STATUS>Error</STATUS>
        <POSTURE_MODIFIED_DATE>2012-10-
16T23:02:47Z</POSTURE_MODIFIED_DATE>
      </INFO>
      <INFO>
        <ID>2973247</ID>
        <HOST_ID>1633074</HOST_ID>
        <CONTROL_ID>1517</CONTROL_ID>
        <TECHNOLOGY_ID>1</TECHNOLOGY_ID>
        <INSTANCE></INSTANCE>
        <STATUS>Error</STATUS>
        <POSTURE_MODIFIED_DATE>2012-10-
16T23:02:48Z</POSTURE_MODIFIED_DATE>
      </INFO>
      <INFO>
        <ID>3933971</ID>
        <HOST_ID>1839957</HOST_ID>
        <CONTROL_ID>1071</CONTROL_ID>
        <TECHNOLOGY_ID>2</TECHNOLOGY_ID>
        <INSTANCE></INSTANCE>
        <STATUS>Error</STATUS>
        <POSTURE_MODIFIED_DATE>2012-10-
17T21:34:18Z</POSTURE_MODIFIED_DATE>
      </INFO>
      <INFO>
```

```
          <ID>3933972</ID>
          <HOST_ID>1839957</HOST_ID>
          <CONTROL_ID>1072</CONTROL_ID>
          <TECHNOLOGY_ID>2</TECHNOLOGY_ID>
          <INSTANCE></INSTANCE>
          <STATUS>Error</STATUS>
          <POSTURE_MODIFIED_DATE>2012-10-
17T21:34:18Z</POSTURE_MODIFIED_DATE>
      </INFO>
...
```

PC - New UDC Reporting Option

We've added a new user defined control (UDC) reporting option to help you better manage your risk and compliance. This option allows you to pass or fail the control in cases where it returns error code 2 "item not found" (e.g. scan did not find file, registry key, or setting within a file or registry key). When enabled we'll add a check box to the policy where you can set the status you prefer. Good to Know - If this option is enabled, the Ignore Errors setting is not applied when "item not found" error is returned.

UDCs that support the new Reporting Option

ImportableControl.xsd

We've added the IGNORE_ITEM_NOT_FOUND. Set this option if you want to pass or fail the control when it returns error code 2 "item not found".

```
...
<xs:element ref="COMMENT" minOccurs="0" maxOccurs="1" />
<xs:element ref="IGNORE_ERROR" maxOccurs="1" />
<xs:element ref="IGNORE_ITEM_NOT_FOUND" minOccurs="0" maxOccurs="1" />
<xs:element ref="SCAN_PARAMETERS" maxOccurs="1" />
...
<xs:element name="IGNORE_ITEM_NOT_FOUND">
 <xs:simpleType>
 <xs:restriction base="xs:integer">
 <xs:enumeration value="0" />
 <xs:enumeration value="1" />
 </xs:restriction>
 </xs:simpleType>
</xs:element>
...
```

Sample UDC XML

```
<CONTROL_LIST total="1">
    <CONTROL>
        <CHECK_TYPE>Unix File/Directory Existence</CHECK_TYPE>
        <CATEGORY>
            <ID>6</ID>
            <NAME><![CDATA[Integrity and Availability]]></NAME>
        </CATEGORY>
        <SUB_CATEGORY>
            <ID>1027</ID>
            <NAME><![CDATA[Auditing/Logging]]></NAME>
        </SUB_CATEGORY>
        <STATEMENT><![CDATA[Unix - file / dir existence - 1
fail]]></STATEMENT>
        <CRITICALITY>
            <LABEL><![CDATA[UNDEFINED]]></LABEL>
```

```
            <VALUE>0</VALUE>
        </CRITICALITY>
        <COMMENT><![CDATA[]]></COMMENT>
        <IGNORE_ERROR>0</IGNORE_ERROR>
        <IGNORE_ITEM_NOT_FOUND>1</IGNORE_ITEM_NOT_FOUND>
        <SCAN_PARAMETERS>
            <FILE_PATH><![CDATA[/trips]]></FILE_PATH>
            <DATA_TYPE>Boolean</DATA_TYPE>
            <DESCRIPTION><![CDATA[tese]]></DESCRIPTION>
        </SCAN_PARAMETERS>
        <TECHNOLOGY_LIST total="40">
            <TECHNOLOGY>
                <ID>3</ID>
                <TECH_NAME><![CDATA[Red Hat Enterprise Linux
3/4]]></TECH_NAME>
                <RATIONALE><![CDATA[/trips]]></RATIONALE>
                <DATAPOINT>
                    <CARDINALITY>no cd</CARDINALITY>
                    <OPERATOR>no op</OPERATOR>
                    <DEFAULT_VALUES total="1">
                        <DEFAULT_VALUE>true</DEFAULT_VALUE>
                    </DEFAULT_VALUES>
                </DATAPOINT>
            </TECHNOLOGY>
            <TECHNOLOGY>
                <ID>4</ID>
                <TECH_NAME><![CDATA[Solaris 9.x]]></TECH_NAME>
                <RATIONALE><![CDATA[/trips]]></RATIONALE>
                <DATAPOINT>
                    <CARDINALITY>no cd</CARDINALITY>
                    <OPERATOR>no op</OPERATOR>
                    <DEFAULT_VALUES total="1">
                        <DEFAULT_VALUE>true</DEFAULT_VALUE>
                    </DEFAULT_VALUES>
                </DATAPOINT>
            </TECHNOLOGY>
            <TECHNOLOGY>
                <ID>5</ID>
                <TECH_NAME><![CDATA[HPUX 11.iv1]]></TECH_NAME>
                <RATIONALE><![CDATA[/trips]]></RATIONALE>
                <DATAPOINT>
                    <CARDINALITY>no cd</CARDINALITY>
                    <OPERATOR>no op</OPERATOR>
                    <DEFAULT_VALUES total="1">
                        <DEFAULT_VALUE>true</DEFAULT_VALUE>
                    </DEFAULT_VALUES>
                </DATAPOINT>
            </TECHNOLOGY>
...
```

UDCs that support the new Reporting Option

Windows Control Types	Unix Control Types
Registry Key Existence	File/Directory Existence
Registry Value Existence	File/Directory Permission
Registry Value Content Check	File Content Check
Registry Permission	File Integrity Check
File/Directory Existence	
File/Directory Permission	
File Integrity Check	
List Group Members	