



# **Security Assessment Questionnaire API**

User Guide

Version 3.3

November 11, 2020

## Table of Contents

Start Here .....	4
Security Assessment Questionnaire API.....	4
Qualys user account.....	5
URL to Qualys API server.....	6
Making API calls.....	8
Know your portal version.....	10
SAQ Users .....	12
Search users .....	12
Create user.....	16
Update user .....	20
Update users (bulk).....	23
Delete user .....	26
Delete users (bulk).....	28
SAQ Templates .....	31
Search library templates .....	31
Get template details.....	34
Search templates.....	39
Update a template .....	47
Update a template (bulk).....	50
Publish template .....	54
Delete template .....	56

Delete templates (bulk).....	58
Reference: Template.....	60
SAQ Campaigns.....	64
Search campaigns.....	64
Get Campaign details .....	68
Current campaign count .....	71
Get campaign status.....	74
Complete campaign.....	76
Update a campaign .....	78
Delete campaign.....	81
Launch campaign.....	83
Complete campaign.....	86
Cancel campaign .....	88
View campaign summary.....	90
Reference: Campaign .....	92
SAQ Questionnaires .....	96
Search questionnaires .....	96
Appendix .....	100
Appendix A - Error Messages.....	100

# Start Here

## Security Assessment Questionnaire API

Welcome to Qualys Security Assessment Questionnaire (SAQ) API. This user guide is intended for application developers who will use the Qualys SAQ API.

Modules supported

SAQ

Authentication

Authentication to your Qualys account with valid Qualys credentials is required for making Qualys API requests to the Qualys API servers. [Learn more about authentication to your Qualys account](#)

Get API Notifications

We recommend you join our Community and subscribe to our API Notifications RSS Feeds for announcements and discussions.

<https://community.qualys.com/community/developer/notifications-api>

About Qualys

Qualys, Inc. (NASDAQ: QLYS) is a pioneer and leading provider of cloud-based security and compliance solutions. The Qualys Cloud Platform and its integrated Cloud Apps deliver businesses critical security intelligence continuously, enabling them to automate the full spectrum of auditing, compliance and protection for IT systems and web applications on premises, on endpoints and elastic clouds. For more information, please visit [www.qualys.com](http://www.qualys.com)

Qualys and the Qualys logo are proprietary trademarks of Qualys, Inc. All other products or names may be trademarks of their respective companies

## Qualys user account



Authentication with valid Qualys user account credentials is required for making Qualys API requests to the Qualys API servers. These servers are hosted at the Qualys platform, also referred to as the Security Operations Center (SOC), where your account is located. If you need assistance with obtaining a Qualys account, please contact your Qualys account representative. Qualys user accounts that have been enabled with VIP two-factor authentication can be used with the Qualys API, however two-factor authentication will not be used when making API requests. Two-factor authentication is only supported when logging into the Qualys GUI.

## URL to Qualys API server

Qualys maintains multiple Qualys Cloud Platforms. The API server URL that you should use for API requests depends on the platform where your Qualys account is located.

Account location	API server URL
Qualys US Platform 1	<a href="https://qualysapi.qualys.com">https://qualysapi.qualys.com</a>
Qualys US Platform 2	<a href="https://qualysapi.qg2.apps.qualys.com">https://qualysapi.qg2.apps.qualys.com</a>
Qualys US Platform 3	<a href="https://qualysapi.qg3.apps.qualys.com">https://qualysapi.qg3.apps.qualys.com</a>
Qualys EU Platform 1	<a href="https://qualysapi.qualys.eu">https://qualysapi.qualys.eu</a>
Qualys EU Platform 2	<a href="https://qualysapi.qg2.apps.qualys.eu">https://qualysapi.qg2.apps.qualys.eu</a>
Qualys India Platform 1	<a href="https://qualysapi.qg1.apps.qualys.in">https://qualysapi.qg1.apps.qualys.in</a>
Qualys Private Cloud Platform	<a href="https://qualysapi.&lt;customer_base_url&gt;">https://qualysapi.&lt;customer_base_url&gt;</a>

Looking for your API server URL for your account? You can find this easily. Just log in to your Qualys account and go to Help > About. You'll see this information under Security Operations Center (SOC).

**About** Launch Help  

**General Information** >

Identified Services >

Identified OS >

Additional References >

---

### General Information

**Qualys Web Service**

Application Version:	8.9.0.2-2
Online Help Version:	8.9.29-1
SCAP Module Version:	1.2

**Qualys External Scanners**

Security Operations Center (SOC):	64.39.96.0/20 (64.39.96.1-64.39.111.254)
Scanner Version:	9.0.29-1
Vulnerability Signature Version:	2.3.492-2
Scanner Services	3.0.12-1

**Qualys Scanner Appliances**

Security Operations Center (SOC):	- qualysguard.qualys.com:443
	- <b>qualysapi.qualys.com:443</b>
	- dist01.sjdc01.qualys.com:443
	- nochoost.sjdc01.qualys.com:443
	- scanservice1.qualys.com:443
	- all in 64.39.96.0/20

## Making API calls

Curl samples in our API doc

We use curl in our API documentation to show an example how to form REST API calls, and it is not meant to be an actual production example of implementation.

### GET and POST Methods

Qualys API functions allow API users to submit parameters (name=value pairs) using the GET and/or POST method. There are known limits for the amount of data that can be sent using the GET method, and these limits are dependent on the toolkit used. Please refer to the individual descriptions of the API function calls to learn about the supported methods for each function

### Parameters in URLs

API parameters, as documented in this user guide, should be specified one time for each URL. In the case where the same parameter is specified multiple times in a single URL, the last parameter takes effect and the previous instances are silently ignored. URL elements are case sensitive.

### Date format in API Results

The Qualys API has adopted a date/time format to provide consistency and interoperability of the Qualys API with third-party applications. The date format follows standards published in RFC 3339 and ISO 8601, and applies throughout the Qualys API. The date format is: yyyy-mm-ddThh-mm-ssZ This represents a UTC value (GMT time zone).

### URL Encoding in API Code

You must URL encode variables when using the Qualys API. This is standard practice for HTTP communications. If your application passes special characters, like the single quote ('), parentheses, and symbols, they must be URL encoded. For example, the pound (#) character cannot be used as an input parameter in URLs. If “#” is specified, the Qualys API returns an error. To



specify the “#” character in a URL you must enter the encoded value “%23”. The “#” character is considered by browsers and other Internet tools as a separator between the URL and the results page, so whatever follows an un-encoded “#” character is not passed to the Qualys API server and returns an error.

## Know your portal version

/qps/rest/portal/version/

[GET] [POST]

Using the Version API you can find out the installed version of Portal and its sub-modules that are available in your subscription.

Sample XML

### API request

```
curl -u "USERNAME:PASSWORD" -X "GET" -H "Accept: application/xml"
https://qualysapi.qualys.com/qps/rest/portal/version
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="https://qualysapi.qualys.com/qps/xsd/ve
rsion.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <Portal-Version>
      <PortalApplication-VERSION>2.33.0.0-SNAPSHOT-1 DEVELOP
#352 (2018-05-07T22:53:43Z)</PortalApplication-VERSION>
      <WAS-VERSION>6.0.0.0</WAS-VERSION>
      <FIM-VERSION>1.5.1</FIM-VERSION>
      <VM-VERSION>1.0.3</VM-VERSION>
      <CERTVIEW-VERSION>1.1.0.0</CERTVIEW-VERSION>
      <CM-VERSION>1.20.1</CM-VERSION>
      <MDS-VERSION>2.11.7.0</MDS-VERSION>
      <CA-VERSION>2.9.1.0</CA-VERSION>
      <IOC-VERSION>1.1.0</IOC-VERSION>
      <AV2-VERSION>0.1.0</AV2-VERSION>
      <QUESTIONNAIRE-VERSION>2.14.0.4</QUESTIONNAIRE-VERSION>
      <WAF-VERSION>2.7.0.0</WAF-VERSION>
    </Portal-Version>
```

```
</data>  
</ServiceResponse>
```

Sample JSON

### API request

```
curl -u "USERNAME:PASSWORD" -X "GET" -H "Accept: application/json"  
https://qualysapi.qualys.com/qps/rest/portal/version
```

### Response

```
{  
  "ServiceResponse": {  
    "data": [  
      {  
        "Portal-Version": {  
          "PortalApplication-VERSION": "2.33.0.0-SNAPSHOT-1 DEVELOP  
#352 (2018-05-07T22:53:43Z)",  
          "WAS-VERSION": "6.0.0.0",  
          "VM-VERSION": "1.0.3",  
          "CM-VERSION": "1.20.1",  
          "MDS-VERSION": "2.11.7.0",  
          "CA-VERSION": "2.9.1.0",  
          "QUESTIONNAIRE-VERSION": "2.14.0.4",  
          "WAF-VERSION": "2.7.0.0"  
        },  
        ...  
      }  
    ],  
    "responseCode": "SUCCESS",  
    "count": 1  
  }  
}
```

# SAQ Users

## Search users

`/qps/rest/1.0/search/saq/user/`

[GET] [POST]

Search users in the API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" permission, Output includes users within the API user's scope

### Input Parameter

The following input elements are optional and act as filters. When multiple elements are specified, parameters are combined using a logical AND.

Optional	Allowed operators
id (Integer)	EQUALS, IN
uuid (Integer)	EQUALS
firstName (Text)	CONTAINS, EQUALS
lastName (Text)	CONTAINS, EQUALS
company (Text)	CONTAINS, EQUALS
title (Text)	CONTAINS, EQUALS
emailAddress (Text)	CONTAINS, EQUALS

userName           CONTAINS, EQUALS  
(Text)

tags.tag.id        EQUALS, IN  
(Integer)

tags.tag.name      CONTAINS, EQUALS  
(Text)

### Sample - Search user

#### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --  
data-binary @-  
"http://qualysapi.qualys.com/qps/rest/1.0/search/saq/user" < file.xml  
Note: "file.xml" contains the request POST data.
```

#### Request POST data

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceRequest>  
  <filters>  
    <Criteria field="id" operator="EQUALS">400591</Criteria>  
  </filters>  
</ServiceRequest>
```

#### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0/  
/saq/user.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <hasMoreRecords>>false</hasMoreRecords>  
  <data>  
    <User>
```

```
<id>400591</id>
<uuid>2277d4ff-9068-44f2-97d0-ccf35cfa1679</uuid>
<firstName>user</firstName>
<lastName>q</lastName>
<company>abc</company>
<emailAddress>user@abc.com</emailAddress>
<title>dev</title>
<tags>
  <Tag>
    <id>7513412</id>
    <name>Malware Domain Assets</name>
  </Tag>
  <Tag>
    <id>7533412</id>
    <name>fadi's tag</name>
  </Tag>
  <Tag>
    <id>7508422</id>
    <name>Business Units</name>
  </Tag>
  <Tag>
    <id>7520414</id>
    <name>test2</name>
  </Tag>
  <Tag>
    <id>7546412</id>
    <name>Passive Scanning</name>
  </Tag>
  <Tag>
    <id>7553612</id>
    <name>tag1</name>
  </Tag>
  <Tag>
    <id>7553812</id>
    <name>tag2</name>
  </Tag>
</tags>
</User>
</data>
</ServiceResponse>
```

XSD

Qualys Security Assessment Questionnaire API  
SAQ Users

<https://<platform API server>/qps/xsd/1.0/saq/user.xsd>

## Create user

`/qps/rest/1.0/create/saq/user/`

[POST]

Add a new user to the API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" permission

### Input Parameters

Required elements	Allowed operators
firstName (Text)	CONTAINS, EQUALS, NOT EQUALS
lastName (Text)	CONTAINS, EQUALS, NOT EQUALS
company (Text)	CONTAINS, EQUALS, NOT EQUALS
emailAddress (Text)	CONTAINS, EQUALS, NOT EQUALS
Optional elements	
title (Text)	CONTAINS, EQUALS, NOT EQUALS
lastScanned	
tags (List)	CONTAINS, EQUALS, NOT EQUALS
tags.tag.name (Text)	EQUALS, NOT EQUALS



## Sample - Create a user

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST"
--data-binary @-
"http://qualysapi.qualys.com/qps/rest/1.0/create/saq/user" < file.xml
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceRequest>
  <data>
    <User>
      <firstName>user</firstName>
      <lastName>user</lastName>
      <company>abc</company>
      <emailAddress>user@abc.com</emailAddress>
    </User>
  </data>
</ServiceRequest>
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0
/saq/user.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <User>
      <id>401990</id>
      <uuid>734d9b3d-d246-4bb6-a028-5a390b47eb2a</uuid>
      <firstName>user</firstName>
      <lastName>user</lastName>
      <company>qualys</company>
      <emailAddress>user@abc.com</emailAddress>
    </User>
  </data>
</ServiceResponse>
```

## Sample - Create multiple users

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --data-binary @-"http://qualysapi.qualys.com/qps/rest/1.0/create/saq/user" < file.xml  
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceRequest>  
  <data>  
    <User>  
      <firstName>abc-user</firstName>  
      <lastName>xyz-user</lastName>  
      <company>abc</company>  
      <emailAddress>user@abc.com</emailAddress>  
    </User>  
    <User>  
      <firstName>User</firstName>  
      <lastName>pqr</lastName>  
      <company>abc</company>  
      <emailAddress>user2@abc.com</emailAddress>  
    </User>  
  </data>  
</ServiceRequest>
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0/saq/user.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>2</count>  
  <data>  
    <User>  
      <id>404591</id>  
      <uuid>328ea575-fc9d-4562-aa28-c97e7ba5482f</uuid>  
      <firstName>abc-user</firstName>
```

```
    <lastName>xyz-user</lastName>
    <company>abc</company>
    <emailAddress>user@abc.com</emailAddress>
    <userName>user</userName>
  </User>
  <User>
    <id>404592</id>
    <uuid>4f6c65b6-e877-43fe-9b45-50993235f60e</uuid>
    <firstName>User</firstName>
    <lastName>pqr</lastName>
    <company>abc</company>
    <emailAddress>user2@abc.com</emailAddress>
    <userName>user2</userName>
  </User>
</data>
</ServiceResponse>
```

XSD

[<platform API server>/qps/xsd/1.0/saq/user.xsd](#)

## Update user

/qps/rest/1.0/update/saq/user/<id>

[POST]

Update a user in the API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" permission, Output includes users within the API user's scope

### Input Parameters

The "id" (Long) element is required to identify the user of interest. Other elements are optional.

Required elements	Allowed operators
firstName (Text)	CONTAINS, EQUALS, NOT EQUALS
lastName (Text)	CONTAINS, EQUALS, NOT EQUALS
company (Text)	CONTAINS, EQUALS, NOT EQUALS
emailAddress (Text)	CONTAINS, EQUALS, NOT EQUALS
title (Text)	CONTAINS, EQUALS, NOT EQUALS
tags (List)	
Optional elements	
userName	CONTAINS, EQUALS, NOT EQUALS

(Text)

id (integer)	CONTAINS, EQUALS, NOT EQUALS
--------------	------------------------------

uuid (integer)	CONTAINS, EQUALS, NOT EQUALS
----------------	------------------------------

tags.tag.id (integer)	CONTAINS, EQUALS, NOT EQUALS
--------------------------	------------------------------

tags.tag.name (Text)	EQUALS, NOT EQUALS
-------------------------	--------------------

## Sample - Update a user

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST"
--data-binary @-
"https://qualysapi.qualys.com/qps/rest/1.0/update/saq/user/387600" <
file.xml
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceRequest>
  <data>
    <User>
      <id>387600</id>
      <firstName>user</firstName>
      <lastName>user</lastName>
      <company>abc</company>
      <emailAddress>user@abc.com</emailAddress>
    </User>
  </data>
</ServiceRequest>
```

## Response

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0
/saq/user.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <User>
      <id>387600</id>
      <uuid>c54c47ef-57f6-42c8-bdd4-411c5f252234</uuid>
      <firstName>user1</firstName>
      <lastName>user11</lastName>
      <company>abc</company>
      <emailAddress>user1@abc.com</emailAddress>
    </User>
  </data>
</ServiceResponse>
```

XSD

[<platform API server>/qps/xsd/1.0/saq/user.xsd](#)

See [Update users \(bulk\)](#)

## Update users (bulk)

`/qps/rest/1.0/update/saq/user/`

[POST]

Update multiple users in the API user's scope

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" permission, Output includes users within the API user's scope

### Input Parameter

The following input elements are optional and act as filters. When multiple elements are specified, parameters are combined using a logical AND.

Optional	Allowed operators
id (Integer)	EQUALS, IN
uuid (Integer)	EQUALS
firstName (Text)	CONTAINS, EQUALS
lastName (Text)	CONTAINS, EQUALS
company (Text)	CONTAINS, EQUALS
title (Text)	CONTAINS, EQUALS
emailAddress (Text)	CONTAINS, EQUALS
userName (Text)	CONTAINS, EQUALS

tags.tag.id (Integer)	EQUALS, IN
tags.tag.name (Text)	CONTAINS, EQUALS

## Sample - Update users in bulk

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --data-binary @-"https://qualysapi.qualys.com/qps/rest/1.0/update/saq/user" < file.xml  
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceRequest>  
  <filters>  
    <Criteria field="id" operator="EQUALS">403390</Criteria>  
  </filters>  
  <data>  
    <User>  
      <lastName>user-api-renamed</lastName>  
    </User>  
  </data>  
</ServiceRequest>
```

### Response

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



Qualys Security Assessment Questionnaire API  
SAQ Users

```
<User>
  <id>403390</id>
  <uuid>b472f092-fed9-4b0d-8e08-98490e8f0726</uuid>
  <firstName>api-user-renamed</firstName>
  <lastName>user-api-renamed</lastName>
  <company>abc-renamed</company>
  <emailAddress>user@qualys.com</emailAddress>
  <userName>api-user.api-user@abc</userName>
</User>
</data>
</ServiceResponse>
```

XSD

[<platform API server>/qps/xsd/1.0/saq/user.xsd](#)

## Delete user

/qps/rest/1.0/delete/saq/user/<id>

[POST]

Delete a user in API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" permission, Output includes users within the API user's scope

Input

The "id" (Long) element is required to identify the user of interest. Other elements are optional.

Sample - Delete a user

### API request

```
curl -u "USERNAME:PASSWORD" -X "GET" -H "Content-Type: text/xml"
https://qualysapi.qualys.com/qps/rest/1.0/delete/saq/user/401990
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0
/saq/user.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <User>
      <id>401990</id>
    </User>
  </data>
</ServiceResponse>
```

XSD

<https://<platform API server>/qps/xsd/1.0/saq/user.xsd>

See [Delete users \(bulk\)](#)

## Delete users (bulk)

`/qps/rest/1.0/update/saq/user/`

[POST]

Delete multiple users in the API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" permission, Output includes users within the API user's scope

### Input Parameter

The following input elements are optional and act as filters. When multiple elements are specified, parameters are combined using a logical AND.

Optional	Allowed operators
id (Integer)	EQUALS, IN
uuid (Integer)	EQUALS
firstName (Text)	CONTAINS, EQUALS
lastName (Text)	CONTAINS, EQUALS
company (Text)	CONTAINS, EQUALS
title (Text)	CONTAINS, EQUALS
emailAddress (Text)	CONTAINS, EQUALS
userName (Text)	CONTAINS, EQUALS
tags.tag.id	EQUALS, IN

(Integer)

tags.tag.name      CONTAINS, EQUALS  
(Text)

Sample - Delete user in bulk

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST"
--data-binary @-
"https://qualysapi.qualys.com/qps/rest/1.0/delete/saq/user" < file.xml
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<?xml version="1.0" ?>
<ServiceRequest>
  <filters>
    <Criteria field="userName" operator="EQUALS">api-user.api-
user@abc</Criteria>
  </filters>
</ServiceRequest>
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0
/saq/user.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <User>
      <id>403190</id>
    </User>
  </data>
</ServiceResponse>
```

XSD

<http://<platform API server>/qps/xsd/1.0/saq/user.xsd>

# SAQ Templates

## Search library templates

`/qps/rest/1.0/search/saq/librarytemplate/`

[POST]

Search library templates in the API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" permission, Output includes templates within the API user's scope

### Input Parameter

The following input elements are optional and act as filters. When multiple elements are specified, parameters are combined using a logical AND.

Optional	Allowed operators
id (Integer)	EQUALS, IN
uuid (Integer)	EQUALS
name (Text)	CONTAINS, EQUALS
description (Text)	CONTAINS, EQUALS
category (Text)	CONTAINS, EQUALS
familyId (Integer)	EQUALS
revision	EQUALS, IN

(Integer)

isLibrary  
(Boolean)           EQUALS

questionCnt  
(Integer)           EQUALS, IN

state (String)       EQUALS, IN

## Sample - Search library templates

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/search/saq/librarytemplate"  
< file.xml  
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<ServiceRequest>  
  <filters>  
    <Criteria field="name" operator="CONTAINS">Vendor Risk  
Assessment</Criteria> </filters>  
</ServiceRequest>
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0  
/saq/librarytemplate.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <hasMoreRecords>>false</hasMoreRecords>  
  <data>
```



```
<LibraryTemplate>
  <id>10030</id>
  <uuid>9fc12d51-0dbb-416c-933c-9762831f8af7</uuid>
  <name>Vendor Risk Assessment</name>
  <description>Vendor Risk Assessment Questionnaire template
provides a standard approach to all High Risk Vendors</description>
  <familyId>f431d2ba-462c-4d14-a8de-5ee2c4c770ce</familyId>
  <revision>1</revision>
  <isLibrary>true</isLibrary>
  <questionCnt>288</questionCnt>
  <state>RETIRED</state>
</LibraryTemplate>
</data>
</ServiceResponse>
```

XSD

[<platform API server>/qps/xsd/1.0/saq/librarytemplate.xsd](#)

## Get template details

`/qps/rest/1.0/get/saq/template/<id>`

[GET]

Returns details for a template in the API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" permission, Output includes templates within the API user's scope

### Input Parameter

The element "id" (Integer) is required, where "id" identifies a template of interest.

### Sample - Get details of template

#### API request

```
curl -u "USERNAME:PASSWORD" -X GET  
https://qualysapi.qualys.com/qps/rest/1.0/get/saq/template/82969
```

#### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0/  
/saq/template.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <data>  
    <Template>  
      <id>82969</id>  
      <uuid>5d9e0a22-14ff-4961-b0f0-a663c3b7f2f3</uuid>  
      <name>template-from-api-with-scoring</name>  
      <description>template containing scoring fields created  
from the api</description>
```

```

<familyId>75120987-737d-4157-9b3f-d0c2229d8c1d</familyId>
<revision>1</revision>
<isLibrary>>false</isLibrary>
<questionCnt>5</questionCnt>
<state>DRAFT</state>
<elements>
  <sections>
    <list>
      <Section>
        <name>section 1</name>
        <description>section 1
description</description>
        <questions>
          <list>
            <Question>
              <label>0</label>
              <text>Question 0</text>
              <type>booleanQuestion</type>
              <mode>BUTTON</mode>
              <criticality>CRITICAL</critica
lity>
              <requireAttachment>>false</requ
ireAttachment>
              <requireComment>>false</require
Comment>
              <requireAsset>>false</requireAs
set>
              <yesScoringLabel>HIGH</yesScor
ingLabel>
              <noScoringLabel>MEDIUM</noScor
ingLabel>
            </Question>
            <Question>
              <label>1</label>
              <text>QUESTION 1</text>
              <type>booleanQuestion</type>
              <mode>BUTTON</mode>
              <criticality>CRITICAL</critica
lity>
              <requireAttachment>>false</requ
ireAttachment>
              <requireComment>>false</require
Comment>

```

Qualys Security Assessment Questionnaire API  
SAQ Templates

```

set>
ingLabel>
ingLabel>
<requireAsset>>false</requireAs
<yesScoringLabel>HIGH</yesScor
<noScoringLabel>MEDIUM</noScor
</Question>
</list>
</questions>
</Section>
<Section>
<name>section 2</name>
<description>section 2
description</description>
<questions>
<list>
<Question>
<label>3</label>
<text>QUESTION 3</text>
<type>multipleChoiceQuestion</
type>
<mode>MULTI_SELECT</mode>
<criticality>INFO</criticality
>
<requireAttachment>>false</requ
<requireComment>>false</require
Comment>
<requireAsset>>false</requireAs
set>
<answers>
<list>
<Answer>
<title>a</title>
<value>0</value>
<scoringLabel>LOW<
/scoringLabel>
<requireAttachment
>>false</requireAttachment>
<requireComment>fa
lse</requireComment>
<requireAsset>fals
e</requireAsset>
</Answer>

```

```

UM</scoringLabel>
>false</requireAttachment>
lse</requireComment>
e</requireAsset>
</scoringLabel>
>false</requireAttachment>
lse</requireComment>
e</requireAsset>
</Answer>
<Answer>
<title>c</title>
<value>2</value>
<scoringLabel>HIGH
</scoringLabel>
<requireAttachment>
>false</requireAttachment>
lse</requireComment>
e</requireAsset>
</Answer>
</list>
</answers>
</Question>
</list>
</questions>
</Section>
</list>
</sections>
</elements>
<scorings>
<list>
<Scoring>
<label>LOW</label>
<value>0</value>
</Scoring>
<Scoring>
<label>MEDIUM</label>
<value>50</value>
</Scoring>
<Scoring>

```

```
        <label>HIGH</label>  
        <value>100</value>  
    </Scoring>  
  </list>  
</scorings>  
</Template>  
</data>  
</ServiceResponse>
```

XSD

<platform API server>/qps/xsd/1.0/saq/template.xsd

## Search templates

`/qps/rest/1.0/search/saq/template/`

[POST]

Search templates in the API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" and "Access SAQ module" permission, Output includes templates within the API user's scope

### Input Parameter

The following input elements are optional and act as filters. When multiple elements are specified, parameters are combined using a logical AND.

Optional	Allowed operators
id (Integer)	EQUALS, IN
uuid (Integer)	EQUALS
name (Text)	CONTAINS, EQUALS
description (Text)	CONTAINS, EQUALS
category (Text)	CONTAINS, EQUALS
familyId (Integer)	EQUALS
revision (Integer)	EQUALS, IN
isLibrary	EQUALS

(Boolean)

questionCnt  
(Integer)            EQUALS, IN

state (String)        EQUALS, IN

## Sample - Search templates

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/search/saq/template" <  
file.xml  
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<ServiceRequest>  
  <filters>  
    <Criteria field="questionCnt"  
      operator="EQUALS">5</Criteria>  
    <Criteria field="state" operator="EQUALS">DRAFT</Criteria>  
    <Criteria field="name"  
      operator="CONTAINS">scoring</Criteria>  
  </filters>  
</ServiceRequest>
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0/  
/saq/template.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <hasMoreRecords>>false</hasMoreRecords>
```



```

<data>
  <Template>
    <id>82969</id>
    <uuid>5d9e0a22-14ff-4961-b0f0-a663c3b7f2f3</uuid>
    <name>template-from-api-with-scoring</name>
    <description>template containing scoring fields created
from the api</description>
    <familyId>75120987-737d-4157-9b3f-d0c2229d8c1d</familyId>
    <revision>1</revision>
    <isLibrary>>false</isLibrary>
    <questionCnt>5</questionCnt>
    <state>DRAFT</state>
    <elements>
      <sections>
        <list>
          <Section>
            <name>section 1</name>
            <description>section 1
description</description>
            <sections/>
            <questions>
              <list>
                <Question>
                  <label>0</label>
                  <text>Question 0</text>
                  <type>booleanQuestion</type>
                  <mode>BUTTON</mode>
                  <criticality>CRITICAL</critica
lity>
                  <requireAttachment>>false</requ
ireAttachment>
                  <requireComment>>false</require
Comment>
                  <requireAsset>>false</requireAs
set>
                  <yesScoringLabel>HIGH</yesScor
ingLabel>
                  <noScoringLabel>MEDIUM</noScor
ingLabel>
                </Question>
                <Question>
                  <label>1</label>
                  <text>QUESTION 1</text>
                  <type>booleanQuestion</type>

```

Qualys Security Assessment Questionnaire API  
SAQ Templates

```

<mode>BUTTON</mode>
<criticality>CRITICAL</critica
lity>
<requireAttachment>>false</requ
Comment>
<requireComment>>false</require
set>
<requireAsset>>false</requireAs
ingLabel>
<yesScoringLabel>HIGH</yesScor
ingLabel>
<noScoringLabel>MEDIUM</noScor
</Question>
<Question>
<label>2</label>
<text>QUESTION 2</text>
<type>multipleChoiceQuestion</
type>
<mode>MULTI_SELECT</mode>
<criticality>INFO</criticality
>
<requireAttachment>>false</requ
ireAttachment>
Comment>
<requireComment>>false</require
set>
<requireAsset>>false</requireAs
<answers>
<list>
<Answer>
<title>choice
0</title>
<value>0</value>
<requireAttachment
>>false</requireAttachment>
<requireComment>fa
lse</requireComment>
<requireAsset>fals
e</requireAsset>
</Answer>
<Answer>
<title>choice
1</title>

```

Qualys Security Assessment Questionnaire API  
SAQ Templates

```

<value>1</value>
<scoringLabel>MEDI
UM</scoringLabel>
<requireAttachment
>>false</requireAttachment>
<requireComment>fa
lse</requireComment>
<requireAsset>fals
e</requireAsset>
</Answer>
<Answer>
<title>choice
2</title>
<value>2</value>
<scoringLabel>HIGH
</scoringLabel>
<requireAttachment
>>false</requireAttachment>
<requireComment>fa
lse</requireComment>
<requireAsset>fals
e</requireAsset>
</Answer>
</list>
</answers>
</Question>
</list>
</questions>
</Section>
<Section>
<name>section 2</name>
<description>section 2
description</description>
<sections/>
<questions>
<list>
<Question>
<label>3</label>
<text>QUESTION 3</text>
<type>multipleChoiceQuestion</
type>
<mode>MULTI_SELECT</mode>
<criticality>INFO</criticality
>

```

Qualys Security Assessment Questionnaire API  
SAQ Templates

```

ireAttachment>
Comment>
set>
<requireAttachment>>false</requ
<requireComment>>false</require
<requireAsset>>false</requireAs
<answers>
  <list>
    <Answer>
      <title>a</title>
      <value>0</value>
      <scoringLabel>LOW<
/scoreingLabel>
      <requireAttachment
>>false</requireAttachment>
      <requireComment>fa
lse</requireComment>
      <requireAsset>fals
e</requireAsset>
    </Answer>
    <Answer>
      <title>b</title>
      <value>1</value>
      <scoringLabel>MEDI
UM</scoringLabel>
      <requireAttachment
>>false</requireAttachment>
      <requireComment>fa
lse</requireComment>
      <requireAsset>fals
e</requireAsset>
    </Answer>
    <Answer>
      <title>c</title>
      <value>2</value>
      <scoringLabel>HIGH
</scoringLabel>
      <requireAttachment
>>false</requireAttachment>
      <requireComment>fa
lse</requireComment>
      <requireAsset>fals
e</requireAsset>
    </Answer>
  </list>
</answers>

```

```

        </list>
      </answers>
    </Question>
  </list>
</questions>
</Section>
</list>
</sections>
<questions>
  <list>
    <Question>
      <label>4</label>
      <text>QUESTION 4</text>
      <type>booleanQuestion</type>
      <mode>BUTTON</mode>
      <criticality>INFO</criticality>
      <requireAttachment>>false</requireAttachmen

t>
      <requireComment>>false</requireComment>
      <requireAsset>>false</requireAsset>
      <yesScoringLabel>MEDIUM</yesScoringLabel>
      <noScoringLabel>LOW</noScoringLabel>
    </Question>
  </list>
</questions>
</elements>
<scorings>
  <list>
    <Scoring>
      <label>LOW</label>
      <value>0</value>
    </Scoring>
    <Scoring>
      <label>MEDIUM</label>
      <value>50</value>
    </Scoring>
    <Scoring>
      <label>HIGH</label>
      <value>100</value>
    </Scoring>
  </list>
</scorings>
</Template>
</data>
```

```
</ServiceResponse>
```

XSD

<platform API server>/qps/xsd/1.0/saq/template.xsd

## Update a template

/qps/rest/1.0/update/saq/template/<id>

[POST]

Update a template.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", "Questionnaire Invite" and "Questionnaire Template Create" permissions

Template must be within the API user's scope

Input Parameter

The "id" element is required, other elements are optional. See [Reference: Template](#) for descriptions of supported elements.

Sample - Update a template

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/update/saq/template/82971"  
< file.xml  
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceRequest>  
  <data>  
    <Template>  
      <name>test-api-renamed</name>  
    </Template>  
  </data>  
</ServiceRequest>
```

## Response

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0
/saq/template.xsd">
<responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <Template>
      <id>82971</id>
      <uuid>a25a405c-2ef5-40ab-b747-518d37057853</uuid>
      <name>test-api-renamed</name>
      <description>template-api template-api-1-boolean-
question</description>
      <familyId>948192e8-0fbd-4fca-bc50-5412f6ee91ff</familyId>
      <revision>1</revision>
      <isLibrary>>false</isLibrary>
      <questionCnt>0</questionCnt>
      <state>DRAFT</state>
      <elements>
        <sections>
          <list>
            <Section>
              <name>section 1</name>
              <description>description section
1</description>
              <sections/>
              <questions>
                <list>
                  <Question>
                    <label>boolean question
label</label>
                    <text>boolean question
text</text>
                    <type>booleanQuestion</type>
                    <mode>BUTTON</mode>
                    <requireAttachment>>false</requ
ireAttachment>
                    <requireComment>>false</require
Comment>
                    <requireAsset>>false</requireAs
set>
                  </Question>
                </list>
              </questions>
            </Section>
          </list>
        </sections>
      </elements>
    </Template>
  </data>
</ServiceResponse>
```



```
        </questions>  
      </Section>  
    </list>  
  </sections>  
<questions/>  
</elements>  
</Template>  
</data>  
</ServiceResponse>
```

XSD

[platform API server](qps/xsd/1.0/saq/template.xsd)qps/xsd/1.0/saq/template.xsd

See [Update a template \(bulk\)](#)

## Update a template (bulk)

/qps/rest/1.0/update/saq/template/

[POST]

Update templates in bulk.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", "Questionnaire Invite" and "Questionnaire Template Create" permissions

Template must be within the API user's scope

Input Parameter

See [Reference: Template](#) for descriptions of supported elements.

See [Search templates](#) for supported operators for various elements you can use to filter template you want to update.

Sample - Update templates in bulk

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/update/saq/template" <  
file.xml
```

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

### Request POST data

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceRequest>  
  <filters>  
    <Criteria field="name" operator="EQUALS">template-using-  
api</Criteria>  
  </filters>  
  <data>
```

```

    <Template>
      <description>test-api</description>
    </Template>
  </data>
</ServiceRequest>

```

## Response

```

<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0
/saq/template.xsd">
<responseCode>SUCCESS</responseCode>
  <count>2</count>
  <data>
    <Template>
      <id>86969</id>
      <uuid>5038cc3d-87a0-437a-8e74-293d8e77694f</uuid>
      <name>template-using-api</name>
      <description>test-api</description>
      <familyId>6f1b9410-7caf-4b4b-8a24-00226abbc911</familyId>
      <revision>1</revision>
      <isLibrary>>false</isLibrary>
      <questionCnt>9</questionCnt>
      <state>RETIRED</state>
      <elements>
        <sections>
          <list>
            <Section>
              <name>section 1</name>
              <description>section 1
description</description>
              <sections/>
              <questions>
                <list>
                  <Question>
                    <label>0</label>
                    <text>text question</text>
                    <type>textQuestion</type>
                    <mode>SIDE_BY_SIDE</mode>
                    <requireAttachment>>false</requ
ireAttachment>
                    <requireComment>>false</require
Comment>

```

```

set>
    <requireAsset>>false</requireAs
</Question>
<Question>
    <label>1</label>
    <text>formatted
question</text>
    <type>textQuestion</type>
    <mode>FORMATTED</mode>
    <requireAttachment>>false</requ
ireAttachment>
    <requireComment>>true</requireC
omment>
    <requireAsset>>false</requireAs
set>
    </Question>
    ...
    <Question>
    <label>8</label>
    <text>Section 2 numeric
question</text>
    <type>numericQuestion</type>
    <requireAttachment>>false</requ
ireAttachment>
    <requireComment>>false</require
Comment>
    <requireAsset>>false</requireAs
set>
    </Question>
    </list>
    </questions>
    </Section>
    </list>
    </sections>
    <questions/>
</elements>
<scorings>
    <list>
    <Scoring>
    <label>LOW</label>
    <value>0</value>
    </Scoring>
    <Scoring>
    <label>MEDIUM</label>

```

```
        <value>50</value>
      </Scoring>
      <Scoring>
        <label>HIGH</label>
        <value>100</value>
      </Scoring>
    </list>
  </scorings>
</Template>
</data>
</ServiceResponse>
```

XSD

<qps/xsd/1.0/saq/template.xsd>

## Publish template

/qps/rest/1.0/publish/saq/template/<id>

[POST]

Publish a template based on its ID.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", "Questionnaire Invite" and "Questionnaire Template Publish" permissions

Template must be within the API user's scope

Input Parameter

The "id" (Long) element is required.

Sample - Publish a template

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/publish/saq/template" <  
file.xml
```

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

### Request POST data

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceRequest>  
  <data>  
    <Template>  
      <id>82968</id>  
    </Template>  
  </data>  
</ServiceRequest>
```

## Response

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0/
/saq/template.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <Template>
      <id>82968</id>
    </Template>
  </data>
</ServiceResponse>
```

XSD

[platform API server](http://platform API server/qps/xsd/1.0/publish/saq/template.xsd)qps/xsd/1.0/publish/saq/template.xsd

## Delete template

/qps/rest/1.0/delete/saq/template/<id>

[POST]

Delete a template in API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", "Questionnaire Invite" and "Questionnaire Template Delete" permissions

Template must be within the API user's scope

Input Parameter

The "id" (Long) element is required.

Sample - Delete a template

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "GET" --  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/delete/saq/template/58969"
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://localhost:8383/qps/xsd/1.0/saq/t  
emplate.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <data>  
    <Template>  
      <id>58969</id>  
    </Template>  
  </data>  
</ServiceResponse>
```



XSD

<https://<platform API server>qps/xsd/1.0/saq/template.xsd>

See [Delete templates \(bulk\)](#)

## Delete templates (bulk)

/qps/rest/1.0/delete/saq/template/

[POST]

Delete multiple templates in the API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", "Questionnaire Invite" and "Questionnaire Template Delete" permissions

Template must be within the API user's scope

Input Parameter

All elements for the search operation are supported. See [Search templates](#).

Sample - Delete template in bulk

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/delete/saq/template" <  
file.xml
```

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

### Request POST data

```
<?xml version="1.0" ?>  
<ServiceRequest>  
  <filters>  
    <Criteria field="category" operator="EQUALS">VENDOR</Criteria>  
  </filters>  
</ServiceRequest>
```

## Response

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0
/saq/template.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>3</count>
  <data>
    <Template>
      <id>54611</id>
    </Template>
    <Template>
      <id>54612</id>
    </Template>
    <Template>
      <id>54806</id>
    </Template>
  </data>
</ServiceResponse>
```

XSD

[platform API server](http://platform API server/qps/xsd/1.0/saq/template.xsd)qps/xsd/1.0/saq/template.xsd

## Reference: Template

A reference of template elements is provided below. Required or Optional indicates whether the element is required to create a template.

### Template Elements

Element	Description
id (Long)	(Optional) ID of the template.
uuid (String)	(Optional) Unique ID of the template.
name (String)	(Optional) Name of the template.
description (String)	(Optional) Description of the template.
category (String)	(Required) Category of the template.
familyId (String)	(Required) Family identifier of the template.
revision (Long)	(Optional) Version of the template.
isLibrary (Boolean)	(Optional) True if the template is created from a library template.
questionCnt (Long)	(Required) Count of questions in a template.
state (String)	(Optional) State of the template (Published, Draft)
elements (List)	(Required) List of Template element (sections, questions, ...)
elements.sections (List)	(Required) List of Template section, a section can contains other sections and questions.

Qualys Security Assessment Questionnaire API  
SAQ Templates

elements.questions (List)	(Required) List of the template questions.
elements.scorings (List)	(Required) List of the template scorings.
Scoring.label (String)	(Optional) Scoring label.
Scoring.value (Integer)	(Optional) Scoring value.
Section.name (String)	(Required) Name of the section.
Section.description (String)	Description of the section.
Question.label (String)	(Optional) Label assigned to the question.
Question.text (String)	(Optional) Description of the question.
Question.type (String)	(Optional) The question type (booleanQuestion, numericQuestion, textQuestion, ...)
Question.mode (String)	(Required) The question mode (Button, Radio, Combo)
Question.criticality (String)	(Required) The question criticality (INFO, CRITICAL, ...)
Question.requireAttachment (Boolean)	(Required) True if the Attachment required flag is set to yes.
Question.requireComment (Boolean)	(Required) True if the Comment required flag is set to yes.
Question.requireAsset (Boolean)	(Required) True if the Asset required flag is set to yes.
Question.yesScoringLabel (String)	(Required) The question yes scoring label, used for Boolean question only.
Question.noScoringLabel (String)	(Required for Boolean question) The question no scoring label, used for

Boolean question only.

Question.minCommentLength (Integer)	(Required) The question min comment length, the value of this field will be stored if the requireComment field is set to true.
Question.maxCommentLength (Integer)	(Required) The question max comment length, the value of this field will be stored if the requireComment field is set to true.
Question.dateRange (Date)	(Required for date question) The question date range, used for a date question only.
Question.minAnswerLength (Integer)	(Required for text question) The question minimum question length, used for a text question only.
Question.maxAnswerLength (Integer)	(Required for text question) The question minimum question length, used for a text question only.
Answer.title (String)	(Required) Title of the answer.
Answer.value (String)	(Required) Value of the answer.
Answer.requireAttachment (Boolean)	(Required) True if the Attachment required flag is set to yes.
Answer.requireComment (Boolean)	(Required) True if the Comment required flag is set to yes.
Answer.requireAsset (Boolean)	(Required) True if the Asset required flag is set to yes.
Answer.scoringLabel (String)	(Required) The answer scoring label (only for the dropdown, multiselect, single select)
Answer.minCommentLength	(Required) The Answer min comment

Qualys Security Assessment Questionnaire API  
SAQ Templates

(Integer)

length, the value of this field will be stored if the requireComment field is set to true.

Answer.maxCommentLength  
(Integer)

(Required) The Answer max comment length, the value of this field will be stored if the requireComment field is set to true.

# SAQ Campaigns

## Search campaigns

`qps/rest/1.0/search/saq/questionnairecampaign/`

[POST]

Search for campaigns using specified filters.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" permission, Output includes campaigns within the API user's scope.

Campaign must be within the API user's scope.

### Input Parameter

The following input elements are optional and act as filters. When multiple elements are specified, parameters are combined using a logical AND.

Optional	Allowed operators
id (Integer)	(Optional) Id of the campaign.  Supports "EQUALS" and "IN" operators. <Criteria field="id" operator="EQUALS">1234</Criteria>
name (string)	(Optional) name of the campaign.  Supports "EQUALS" and "CONTAINS" operators. <Criteria field="name" operator="EQUALS">campaign 1</Criteria>
status (string)	(Optional) Status of the campaign. The values are ACTIVE, INACTIVE, COMPLETE, CANCELED.



```
Supports "EQUALS" and "IN" operators. <Criteria  
field="status" operator="EQUALS">ACTIVE</Criteria>
```

workflowType (string) (Optional) Workflow type of the campaign. The values are VERY\_HIGH, HIGH, MEDIUM, LOW.

```
Supports "EQUALS" and "IN" operators. <Criteria  
field="workflowType"  
operator="EQUALS">LOW</Criteria>
```

## Sample - Search campaigns

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST"--  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/search/saq/questionnairecam  
paign/"<file.xml  
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<ServiceRequest>  
<filters>  
  <Criteria field="name" operator="EQUALS">NCSC Basic Cyber Security  
Controls (BCSC) Version 1.0</Criteria>  
</filters>  
</ServiceRequest>
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0  
/saq/questionnairecampaign.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <hasMoreRecords>>false</hasMoreRecords>
```

Qualys Security Assessment Questionnaire API  
SAQ Campaigns

```
<data>
  <QuestionnaireCampaign>
    <id>565002</id>
    <name>NCSC Basic Cyber Security Controls (BCSC) Version
1.0</name>
    <workflowType>SIMPLE</workflowType>
    <status>CANCELED</status>
    <comments>demo api cancel </comments>
    <created>2019-10-21T04:58:50Z</created>
    <updated>2019-10-23T10:45:26Z</updated>
    <launchDate>2019-10-21T04:58:57Z</launchDate>
    <template>
      <id>624001</id>
      <name>NCSC Basic Cyber Security Controls (BCSC)
Version 1.0</name>
      <questionCnt>135</questionCnt>
    </template>
    <user>
      <id>70596886</id>
      <uuid>3d124658-8f4f-771d-806d-1e18a04a1acf</uuid>
      <firstName>api-user</firstName>
      <lastName>api-user</lastName>
      <emailAddress>user@email.com</emailAddress>
      <userName>user</userName>
      <title>dev</title>
      <tags>
        <Tag>
          <id>101271291</id>
          <name>Unassigned Business Unit</name>
        </Tag>
      </tags>
    </user>
    <campaignNotification>
      <reminderNotification>
        <enabled>false</enabled>
        <interval>1</interval>
        <noOfDays>1</noOfDays>
      </reminderNotification>
      <overdueNotification>
        <enabled>false</enabled>
        <interval>1</interval>
        <noOfDays>1</noOfDays>
      </overdueNotification>
      <dueDateNotification>
```

Qualys Security Assessment Questionnaire API  
SAQ Campaigns

```
        <enabled>false</enabled>
        <interval>1</interval>
<noOfDays>1</noOfDays>
        </dueDateNotification>
    </campaignNotification>
</QuestionnaireCampaign>
</data>
</ServiceResponse>
```

XSD

<platform API server>/qps/xsd/1.0/saq/questionnairecampaign.xsd

## Get Campaign details

/qps/rest/1.0/get/saq/questionnairecampaign/<id>

[GET]

Returns details for a campaign in the API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" permission, Output includes campaigns within the API user's scope.

Campaign must be within the API user's scope.

### Input Parameter

The element "id" (Integer) is required, where "id" identifies a campaign of interest.

### Sample - Get details of campaign

#### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "GET"--  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/get/saq/questionnairecampai  
gn/640002"
```

#### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0  
/saq/questionnairecampaign.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <data>  
    <QuestionnaireCampaign>  
      <id>640002</id>  
      <name>Test-Campaign-KT</name>
```

Qualys Security Assessment Questionnaire API  
SAQ Campaigns

```
<workflowType>FULL</workflowType>
<status>COMPLETE</status>
<created>2019-10-23T05:47:16Z</created>
<updated>2019-10-24T05:47:50Z</updated>
<launchDate>2019-10-23T05:47:50Z</launchDate>
<template>
  <id>722010</id>
  <name>Test-Template-KT</name>
  <questionCnt>19</questionCnt>
</template>
<user>
  <id>70596886</id>
  <uuid>3d124658-8f4f-771d-806d-1e18a04a1acf</uuid>
  <firstName>john</firstName>
  <lastName>doe</lastName>
  <emailAddress>user@email.com</emailAddress>
  <userName>user</userName>
  <title>QA</title>
  <tags>
    <Tag>
      <id>101271291</id>
      <name>Unassigned Business Unit</name>
    </Tag>
  </tags>
</user>
<reviewer>
  <id>71077391</id>
  <uuid>831b7266-08f5-4a10-afdd-ab9453075355</uuid>
  <firstName>api-user</firstName>
  <lastName>api-user</lastName>
  <company>afko</company>
  <emailAddress>user@email.com</emailAddress>
  <userName>user</userName>
  <title>reviewer</title>
</reviewer>
<approver>
  <id>89375990</id>
  <uuid>76fcec21-f674-4e17-a868-801f7a92c35c</uuid>
  <firstName>john</firstName>
  <lastName>doe</lastName>
  <company>afko</company>
  <emailAddress>user@email.com</emailAddress>
  <userName>user</userName>
</approver>
```

```
<campaignNotification>
  <reminderNotification>
    <enabled>>false</enabled>
    <interval>1</interval>
    <noOfDays>1</noOfDays>
  </reminderNotification>
  <overdueNotification>
    <enabled>>false</enabled>
    <interval>1</interval>
    <noOfDays>1</noOfDays>
  </overdueNotification>
  <dueDateNotification>
    <enabled>>false</enabled>
    <interval>1</interval>
    <noOfDays>1</noOfDays>
  </dueDateNotification>
</campaignNotification>
</QuestionnaireCampaign>
</data>
</ServiceResponse>
```

XSD

[platform API server](https://platform-api-server/qps/xsd/1.0/saq/questionnairecampaign.xsd)/qps/xsd/1.0/saq/questionnairecampaign.xsd

## Current campaign count

/qps/rest/1.0/count/saq/questionnairecampaign/

[GET] [POST]

Returns the total number of campaigns in the API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module" permissions.

Campaign must be within the API user's scope.

### Input Parameter

The following input elements are optional and act as filters. When specified POST must be used. Multiple filters are combined using a logical AND.

Optional	Allowed operators
id (Integer)	(Optional) Id of the campaign.  Supports "EQUALS" and "IN" operators. For example, <Criteria field="id" operator="EQUALS">1234</Criteria>
name (string)	(Optional) name of the campaign.  Supports "EQUALS" and "CONTAINS" operators. For example, <Criteria field="name" operator="EQUALS">campaign 1</Criteria>
status (string)	(Optional) Status of the campaign. The values are ACTIVE, INACTIVE, COMPLETE, CANCELED.  Supports "EQUALS" and "IN" operators. For example, <Criteria field="status"

```
operator="EQUALS">ACTIVE</Criteria>
```

**workflowType** (string) (Optional) Workflow type of the campaign. The values are SIMPLE, REVIEWABLE, FULL.

Supports "EQUALS" and "IN" operators. For example,  
<Criteria field="workflowType"  
operator="EQUALS">FULL</Criteria>

## Sample - Get campaign count

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST"--  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/count/saq/questionnairecamp  
aign/"<file.xml  
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceRequest>  
  <filters>  
    <Criteria field="status" operator="EQUALS">COMPLETE</Criteria>  
  </filters>  
</ServiceRequest>
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0  
/saq/questionnairecampaign.xsd">  
<responseCode>SUCCESS</responseCode>  
<count>35</count>  
</ServiceResponse>
```



XSD

[platform API server](https://platform-api-server.com/qps/xsd/1.0/saq/questionnairecampaign.xsd)/qps/xsd/1.0/saq/questionnairecampaign.xsd

## Get campaign status

qps/rest/1.0/status/saq/questionnairecampaign/<id>

[GET]

Get status of a campaign.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS" permission, Output includes campaigns within the API user's scope.

Campaign must be within the API user's scope.

Input Parameter

The "id" (Long) element is required.

Sample - Get status of a campaign

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "GET" --  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/status/saq/questionnairecam  
paign/640002"
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0  
/saq/questionnairecampaign.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <data>  
    <QuestionnaireCampaign>  
      <id>926001</id>  
      <name>Demo-campaign-api-1</name>  
      <status>ACTIVE</status>
```

Qualys Security Assessment Questionnaire API  
SAQ Campaigns

```
</QuestionnaireCampaign>  
</data>  
</ServiceResponse>
```

XSD

[<platform API server>](#)/qps/xsd/1.0/saq/questionnairecampaign.xsd

## Complete campaign

/qps/rest/1.0/complete/saq/questionnairecampaign/<id>

[POST]

Mark a campaign complete. Only active campaigns can be marked complete.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", and "Questionnaire Campaign Complete" permissions.

Campaign must be within the API user's scope.

### Input Parameter

The "id" (Long) element is required.

### Sample - Complete a campaign

#### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --data-binary @"https://qualysapi.qualys.com/qps/rest/1.0/complete/saq/questionnairecampaign/684001"
```

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="qualysapi.qualys.com/qps/xsd/1.0/saq/questionnairecampaign.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <QuestionnaireCampaign>
      <id>684001</id>
      <name>Test10_vinod</name>
      <status>COMPLETE</status>
    </QuestionnaireCampaign>
  </data>
</ServiceResponse>
```

XSD

[platform API server](https://platform-api-server/qps/xsd/1.0/saq/questionnairecampaign.xsd)/qps/xsd/1.0/saq/questionnairecampaign.xsd

## Update a campaign

/qps/rest/1.0/update/saq/questionnairecampaign/<id>

[POST]

Update a campaign.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have “API ACCESS”, “Access SAQ module”, and “Questionnaire Campaign Update” permissions.

Campaign must be within the API user’s scope.

Input Parameter

The “id” element is required, other elements are optional. See [Reference: Campaign](#) for descriptions of supported elements.

Sample - Update a campaign

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST"--  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/update/saq/questionnairecam  
paign/" <file.xml  
Note: "file.xml" contains the request POST data.
```

### Request POST data

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceRequest>  
  <data>  
    <QuestionnaireCampaign>  
      <id>120802</id>  
      <workflowType>FULL</workflowType>  
      <approver>  
        <id>352088</id>  
      </approver>
```

```
<reviewer>
  <id>352088</id>
</reviewer>
<recipients>
  <add>
    <User>
      <id>339590</id>
      <assigned>New </assigned>
    </User>
    <User>
      <id>339591</id>
      <assigned>New </assigned>
    </User>
  </add>
</recipients>
<campaignNotification>
  <overdueNotification>
    <enabled>true</enabled>
    <interval>1</interval>
    <noOfDays>5</noOfDays>
  </overdueNotification>
  <dueDateNotification>
    <enabled>false</enabled>
  </dueDateNotification>
</campaignNotification>
</QuestionnaireCampaign>
</data>
</ServiceRequest>
```

## Response

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0
/saq/questionnairecampaign.xsd">
<responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <QuestionnaireCampaign>
      <id>120802</id>
      <name>Test API demo 1</name>
      <workflowType>FULL</workflowType>
      <status>INACTIVE</status>
      <dueDate>2019-10-23T17:08:00Z</dueDate>
```

```
<created>2019-10-21T10:24:16Z</created>
<updated>2019-10-21T10:26:56Z</updated>
<template>
  <id>122601</id>
</template>
<user>
  <id>337590</id>
</user>
<reviewer>
  <id>352088</id>
</reviewer>
<approver>
  <id>352088</id>
</approver>
<campaignNotification>
  <reminderNotification>
    <enabled>true</enabled>
    <interval>2</interval>
    <noOfDays>2</noOfDays>
  </reminderNotification>
  <overdueNotification>
    <enabled>true</enabled>
    <interval>1</interval>
    <noOfDays>5</noOfDays>
  </overdueNotification>
  <dueDateNotification>
    <enabled>false</enabled>
    <interval>1</interval>
    <noOfDays>1</noOfDays>
  </dueDateNotification>
</campaignNotification>
</QuestionnaireCampaign>
</data>
</ServiceResponse>
```

XSD

[platform API server](https://platform-api-server/qps/xsd/1.0/saq/questionnairecampaign.xsd)/qps/xsd/1.0/saq/questionnairecampaign.xsd



## Delete campaign

/qps/rest/1.0/delete/saq/questionnairecampaign/<id>

[POST]

Delete a campaign in API user's scope.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", and "Questionnaire Campaign Delete" permissions.

Campaign must be within the API user's scope.

Input Parameter

The "id" (Long) element is required.

Sample - Delete a campaign

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST"--  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/delete/saq/questionnairecam  
paign/952401"
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0  
/saq/questionnairecampaign.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <data>  
    <QuestionnaireCampaign>  
      <id>952401</id>  
    </QuestionnaireCampaign>  
  </data>
```

```
</ServiceResponse>
```

XSD

[<platform API server>](#)/qps/xsd/1.0/saq/questionnairecampaign.xsd

## Launch campaign

/qps/rest/1.0/launch/saq/template/<id>

[POST]

Launch a new campaign. You can launch only campaign that are inactive.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", and "Questionnaire Campaign Launch" permissions.

Campaign must be within the API user's scope.

Input Parameter

The "id" (Long) element is required.

Sample - Launch a campaign

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/launch/saq/questionnairecam  
paign/640002"
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0  
/saq/questionnairecampaign.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <data>  
    <QuestionnaireCampaign>  
      <id>640002</id>  
      <name>Test-Campaign-KT</name>  
      <workflowType>FULL</workflowType>
```

Qualys Security Assessment Questionnaire API  
SAQ Campaigns

```
<status>COMPLETE</status>
<created>2019-02-01T05:47:16Z</created>
<updated>2019-02-01T05:47:50Z</updated>
<launchDate>2019-02-01T05:47:50Z</launchDate>
<template>
  <id>722010</id>
  <name>Test-Template-KT</name>
  <questionCnt>19</questionCnt>
</template>
<user>
  <id>70596886</id>
  <uuid>3d124658-8f4f-771d-806d-1e18a04a1acf</uuid>
  <firstName>api-user</firstName>
  <lastName>api-user</lastName>
  <emailAddress>user@email.com</emailAddress>
  <userName>user</userName>
  <title>dev</title>
  <tags>
    <Tag>
      <id>101271291</id>
      <name>Unassigned Business Unit</name>
    </Tag>
  </tags>
</user>
<reviewer>
  <id>71077391</id>
  <uuid>831b7266-08f5-4a10-afdd-ab9453075355</uuid>
  <firstName>api-user</firstName>
  <lastName>api-user</lastName>
  <company>afko</company>
  <emailAddress>user@email.com</emailAddress>
  <userName>user</userName>
  <title>reviewer</title>
</reviewer>
<approver>
  <id>89375990</id>
  <uuid>76fcec21-f674-4e17-a868-801f7a92c35c</uuid>
  <firstName>john</firstName>
  <lastName>doe</lastName>
  <company>afko</company>
  <emailAddress>user@email.com</emailAddress>
  <userName>user</userName>
</approver>
<campaignNotification>
```

Qualys Security Assessment Questionnaire API  
SAQ Campaigns

```
<reminderNotification>
  <enabled>>false</enabled>
  <interval>1</interval>
  <noOfDays>1</noOfDays>
</reminderNotification>
<overdueNotification>
  <enabled>>false</enabled>
  <interval>1</interval>
  <noOfDays>1</noOfDays>
</overdueNotification>
<dueDateNotification>
  <enabled>>false</enabled>
  <interval>1</interval>
  <noOfDays>1</noOfDays>
</dueDateNotification>
</campaignNotification>
</QuestionnaireCampaign>
</data>
</ServiceResponse>
```

XSD

[platform API server/qps/xsd/1.0/saq/questionnairecampaign.xsd](http://platform API server/qps/xsd/1.0/saq/questionnairecampaign.xsd)

## Complete campaign

/qps/rest/1.0/complete/saq/questionnairecampaign/<id>

[POST]

Mark a campaign complete. Only active campaigns can be marked complete.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", and "Questionnaire Campaign Complete" permissions.

Campaign must be within the API user's scope.

Input Parameter

The "id" (Long) element is required.

Sample - Complete a campaign

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --data-binary @-"https://qualysapi.qualys.com/qps/rest/1.0/complete/saq/questionnairecampaign/684001"
```

```
<?xml version="1.0" encoding="UTF-8"?>
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="qualysapi.qualys.com/qps/xsd/1.0/saq/questionnairecampaign.xsd">
  <responseCode>SUCCESS</responseCode>
  <count>1</count>
  <data>
    <QuestionnaireCampaign>
      <id>684001</id>
      <name>Test10_vinod</name>
      <status>COMPLETE</status>
    </QuestionnaireCampaign>
  </data>
</ServiceResponse>
```

XSD

[platform API server](https://platform-api-server.com/qps/xsd/1.0/saq/questionnairecampaign.xsd)/qps/xsd/1.0/saq/questionnairecampaign.xsd

## Cancel campaign

/qps/rest/1.0/cancel/saq/questionnairecampaign/<id>

[POST]

Mark a campaign cancelled. Only active campaigns can be marked cancelled.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", and "Questionnaire Campaign Cancel" permissions.

Campaign must be within the API user's scope.

### Input Parameter

The "id" (Long) and "comments" (Comments for cancelling the campaign) elements are required.

### Sample - Cancel a campaign

#### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/cancel/saq/questionnairecam  
paign/565002"
```

#### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0  
/saq/questionnairecampaign.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <data>  
    <QuestionnaireCampaign>  
      <id>565002</id>
```



Qualys Security Assessment Questionnaire API  
SAQ Campaigns

```
<name>NCSC Basic Cyber Security Controls (BCSC) Version  
1.0</name>  
  <status>CANCELED</status>  
</QuestionnaireCampaign>  
</data>  
</ServiceResponse>
```

XSD

[platform API server](https://platform-api-server/qps/xsd/1.0/saq/questionnairecampaign.xsd)/qps/xsd/1.0/saq/questionnairecampaign.xsd

## View campaign summary

/qps/rest/1.0/summary/saq/questionnairecampaign/<id>

[POST]

View details of a campaign.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", and "Questionnaire Campaign View" permissions.

Campaign must be within the API user's scope.

Input Parameter

The "id" (Long) and "comments" (Comments for cancelling the campaign) elements are required.

Sample - Cancel a campaign

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST"--  
data-binary @-  
"https://qualysapi.qualys.com/qps/rest/1.0/summary/saq/questionnaireca  
mpaign/684001"
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0  
/saq/questionnairecampaign.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>1</count>  
  <data>  
    <QuestionnaireCampaign>  
      <name>Campaign_RevBP</name>  
      <workflowType>REVIEWABLE</workflowType>
```

```
<template>
  <name>ASQTe</name>
</template>
<reviewer>
  <id>71077391</id>
  <userName>reviewer.r.39b9xw@qualys</userName>
</reviewer>
<summary>
  <count>2</count>
  <list>
    <QuestionnaireSummary>
      <state>Closed</state>
      <value>1</value>
    </QuestionnaireSummary>
    <QuestionnaireSummary>
      <state>In Progress</state>
      <value>1</value>
    </QuestionnaireSummary>
  </list>
</summary>
</QuestionnaireCampaign>
</data>
</ServiceResponse>
```

XSD

[platform API server/qps/xsd/1.0/saq/questionnairecampaign.xsd](http://platform API server/qps/xsd/1.0/saq/questionnairecampaign.xsd)

## Reference: Campaign

A reference of campaign elements is provided below. Required or Optional indicates whether the element is required to create a campaign.

### Create Campaign Elements

Element	Description
name (Text)	(Required) Name of the campaign.
template.id (Long)	(Required) ID of the template that is used for creating the campaign.
workflowType (Text)	(Required) Workflow type of the campaign. The values are: SIMPLE, REVIEWABLE and FULL.  - SIMPLE (2-stage): workflow sends the questionnaire to user for information gathering.  - REVIEWABLE (3-stage): workflow sends the answered questionnaire to a reviewer for review.  - FULL (4-stage): workflow sends the answered questionnaire to a reviewer and an approver.
due date (Date)	(Required) Due date for the recipients to complete the campaign.
reviewer.id (Long)	(Optional) ID of the campaign reviewer. Required if the workflow type is Reviewable or Full.
approver.id (Long)	(Optional) ID of the approver. Required if the workflow type is Full.

campaignNotification	List of notification types for the campaign (reminder, due date and overdue).
reminderNotification.enabled (Boolean)	(Optional) True if notification enabled flag is set to true for the campaign.
reminderNotification.noOfDays (Integer)	(Optional) Number of days after which first reminder notification will be sent.
reminderNotification.interval (Integer)	(Optional) The interval in days after which next reminder notification will be sent.
dueDateNotification.enabled (Boolean)	(Optional) True if due date notification enabled flag is set to true for the campaign.
dueDateNotification.noOfDays (Integer)	(Optional) Number of days after which first reminder notification for due date of campaign will be sent.
dueDateNotification.interval (Integer)	(Optional) The interval in days after which the next reminder notification for campaign due date will be sent.
overdueNotification.enabled (Boolean)	(Optional) True if overdue notification enabled flag is set to true for the campaign.
overdueNotification.noOfDays (Integer)	(Optional) Number of days after which first reminder notification for campaign getting overdue will be sent.
overdueNotification.interval (Integer)	(Optional) The interval in days after which the next reminder notification for campaign getting overdue will be sent.

## Update Campaign Elements

Qualys Security Assessment Questionnaire API  
SAQ Campaigns

Element	Description
id (Long)	(Required) Name of the campaign.
name (Text)	(Optional) Name of the campaign.
template.id (Long)	(Optional) ID of the template that is used for creating the campaign.
workflowType (Text)	<p>(Optional) Workflow type of the campaign. The values are: SIMPLE, REVIEWABLE and FULL.</p> <ul style="list-style-type: none"> <li>- SIMPLE (2-stage): workflow sends the questionnaire to user for information gathering.</li> <li>- REVIEWABLE (3-stage): workflow sends the answered questionnaire to a reviewer for review.</li> <li>- FULL (4-stage): workflow sends the answered questionnaire to a reviewer and an approver.</li> </ul>
due date (Date)	(Optional) Due date for the recipients to complete the campaign.
reviewer.id (Long)	(Optional) ID of the campaign reviewer. Required if the workflow type is Reviewable or Full.
approver.id (Long)	(Optional) ID of the approver. Required if the workflow type is Full.
campaignNotification	List of notification types for the campaign (reminder, due date and overdue).
reminderNotification.enabled (Boolean)	(Optional) True if notification enabled flag is set to true for the campaign.

Qualys Security Assessment Questionnaire API  
SAQ Campaigns

reminderNotification.noOfDays (Integer)	(Optional) Number of days after which first reminder notification will be sent.
reminderNotification.interval (Integer)	(Optional) The interval in days after which next reminder notification will be sent.
dueDateNotification.enabled (Boolean)	(Optional) True if due date notification enabled flag is set to true for the campaign.
dueDateNotification.noOfDays (Integer)	(Optional) Number of days after which first reminder notification for due date of campaign will be sent.
dueDateNotification.interval (Integer)	(Optional) The interval in days after which the next reminder notification for campaign due date will be sent.
overdueNotification.enabled (Boolean)	(Optional) True if overdue notification enabled flag is set to true for the campaign.
overdueNotification.noOfDays (Integer)	(Optional) Number of days after which first reminder notification for campaign getting overdue will be sent.
overdueNotification.interval (Integer)	(Optional) The interval in days after which the next reminder notification for campaign getting overdue will be sent.

# SAQ Questionnaires

## Search questionnaires

`/qps/rest/1.0/search/saq/questionnairecampaign/`

[POST]

Search questionnaires of a campaign.

Permissions - User must have the Security Assessment Questionnaire (SAQ) module enabled, User must have "API ACCESS", "Access SAQ module", and "Questionnaire Campaign" permissions.

Campaign must be within the API user's scope.

### Input Parameter

The following input elements are optional and act as filters. When multiple elements are specified, parameters are combined using a logical AND.

Optional	Allowed operators
id (Long)	(Optional) Id of the questionnaire.  Supports "EQUALS" and "IN" operators. <Criteria field="id" operator="EQUALS">1234</Criteria>
campaign.id (Long)	(Optional) Id of the campaign.  Supports "EQUALS" and "IN" operators. <Criteria field="campaign.id" operator="EQUALS">4527</Criteria>
state (string)	(Optional) State of the questionnaire. The values are INFORMATION_GATHERING, APPROVAL, REVIEW, CANCELED, INCOMPLETE, CLOSED.



Supports "EQUALS" and "IN" operators. <Criteria field="state" operator="EQUALS">REVIEW</Criteria>

risk (string) (Optional) Risk value assigned to the questionnaire. The values are VERY\_HIGH, HIGH, MEDIUM, LOW.

Supports "EQUALS" and "IN" operators. <Criteria field="risk" operator="LOW">REVIEW</Criteria>

## Sample - Search questionnaires

### API request

```
curl -u "USERNAME:PASSWORD" -H "content-type: text/xml" -X "POST" --data-binary @-"https://qualysapi.qualys.com/qps/rest/1.0/search/saq/questionnaire/"
```

### Request POST data

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceRequest>  
  <filters>  
    <Criteria field="state" operator="EQUALS">APPROVAL</Criteria>  
  </filters>  
</ServiceRequest>
```

### Response

```
<?xml version="1.0" encoding="UTF-8"?>  
<ServiceResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://qualysapi.qualys.com/qps/xsd/1.0/saq/questionnaire.xsd">  
  <responseCode>SUCCESS</responseCode>  
  <count>2</count>  
  <hasMoreRecords>>false</hasMoreRecords>  
  <data>  
    <Questionnaire>  
      <id>707802</id>  
      <title>Test-campaign-demo1 - responder r</title>
```

Qualys Security Assessment Questionnaire API  
SAQ Questionnaires

```
<dueDate>2019-10-28T00:00:00Z</dueDate>
<assignedDate>2019-10-23T06:31:48Z</assignedDate>
<createdBy>
  <id>70596886</id>
  <uuid>3d124658-8f4f-771d-806d-1e18a04a1acf</uuid>
  <firstName>api-user</firstName>
  <lastName>api-user</lastName>
  <company>afko</company>
  <emailAddress>user@email.com</emailAddress>
  <userName>user</userName>
  <title>QA</title>
  <tags>
    <Tag>
      <id>101271291</id>
      <name>Unassigned Business Unit</name>
    </Tag>
  </tags>
</createdBy>
<updatedBy>
  <id>89375990</id>
  <uuid>76fcec21-f674-4e17-a868-801f7a92c35c</uuid>
  <firstName>api-user</firstName>
  <lastName>api-user</lastName>
  <company>afko</company>
  <emailAddress>user@email.com</emailAddress>
  <userName>user</userName>
  <title>dev</title>
</updatedBy>
<assignedTo>
  <id>89375990</id>
  <uuid>76fcec21-f674-4e17-a868-801f7a92c35c</uuid>
  <firstName>john</firstName>
  <lastName>doe</lastName>
  <company>afko</company>
  <emailAddress>user@email.com</emailAddress>
  <userName>user</userName>
</assignedTo>
<responder>
  <id>71077390</id>
  <uuid>97318a7b-85a8-466d-8071-597cde68e1b4</uuid>
  <firstName>resonder1</firstName>
  <lastName>doe</lastName>
  <company>afko</company>
  <emailAddress>user@email.com</emailAddress>
```

Qualys Security Assessment Questionnaire API  
SAQ Questionnaires

```
    <userName>user</userName>
    <title>responder</title>
  </responder>
  <originalAssignedTo>
    <id>71077390</id>
    <uuid>97318a7b-85a8-466d-8071-597cde68e1b4</uuid>
    <firstName>responder1</firstName>
    <lastName>doe</lastName>
    <company>afko</company>
    <emailAddress>user@email.com</emailAddress>
    <userName>user</userName>
    <title>tag1</title>
  </originalAssignedTo>
  <campaign>
    <id>656002</id>
  </campaign>
  <maxPossibleScore>0</maxPossibleScore>
  <score>0</score>
  <state>APPROVAL</state>
  <progress>
    <total>14</total>
    <unanswered>0</unanswered>
    <answered>14</answered>
    <delegated>0</delegated>
    <qaacceptedQuestions>14</qaacceptedQuestions>
  </progress>
</Questionnaire>
</data>
</ServiceResponse>
```

XSD

[platform API server](https://platform-api-server/qps/xsd/1.0/saq/questionnaire.xsd)/qps/xsd/1.0/saq/questionnaire.xsd

# Appendix

## Appendix A - Error Messages

This appendix describes the types of error messages returned from SAQ API requests.

### Template Elements

Error Message	Resolution
<b>Element</b>	
url: Invalid URL format (<value>).	URL format must be as follows:  http://<baseUrl>/rest/1.0/?parameters
Url: Element is required	Element "Url" is required.
uris.<field>: Invalid URL format (<value>).	For the uri.<field> sub element, specify a URL like http://domain.name/base/url/?parameters
uris.<field>: Length of the field must not be greater than 2048 characters. (<value>).	For the uri.<field> sub element, the maximum field length is 2048 characters.
Attribute.category: Element is required.	The element Attribute.category is required.
Attribute.category: Invalid value (<value>).	Element Attribute.category must be set to one of these values: Business Function, Business Location, Business Description.

Attribute.value: Element is required.	Provide a value for the attribute in the Attribute.value element: function, location or description.
The attribute length cannot be greater than 64 characters.	The value for this attribute cannot exceed 64 characters.
The attribute length cannot be greater than 2048 characters.	The value for this attribute cannot exceed 2048 characters.
<element>: Element must not be set.	This element does not apply to this request.
set: Element must contain at least one child.	The set element requires at least one sub element.
headers: Length of all headers cannot exceed 2048 characters.	The values of all headers cannot exceed 2048 characters.
At least one of the following elements must be set: set, add, remove.	For an “update” request you must set at least one of these elements: set, add or remove.
UrlEntry: Element is required.	The element UrlEntry must be provided.
UrlEntry: Invalid URL format (value).	Specify a URL like <a href="http://domain.name/base/url/?parameters">http://domain.name/base/url/?parameters</a>
<parent>: Length of all [URLs, regular	The list of entries for a given type shall not exceed 2048 characters.

expressions] cannot exceed 2048 characters	
UrlEntry: Only regular expressions are accepted for this element.	You must provide regular expressions for the element postDataBlackList.
tags.<element>: Element must not be set.	The tags element does not apply for this request
tags.set: Element must contain at least one child.	At least one sub element must be provided for the element tag.set.
Tag.id: Element is required.	Provide a value for the element Tag.id
Tag.id: Invalid value (value).	Value must be an integer set at least to 1.
Tag: Tag specified by ID <id> does not exist or is not available.	Provide a value for the element id that corresponds to a valid tag.

### Criteria

Criteria: Field is required.	Specify the name of the criteria to search against.
Criteria: Invalid criteria (<field name>).	Please search against one of the following criteria: %s.
Criteria: Invalid operator for criteria '<field>' (<operator>).	Allowed operations for this criteria are: %s.

Criteria: Value is required for criteria '<field>'. Specify a value for a field name for search criteria.

Criteria: Invalid value format for criteria '<field>': <value>.

Boolean (true, false).

Date and Time in UTC format

Enumeration (allowed options separated by comma).

Other: Specify criteria value(s) as <type>.

### Authorization

You are not authorized to access the application through the API. You must be granted the API Access permission in your roles and scopes.

No data shall be passed for this operation. The POST request does not specify a data element.

User is not authorized to perform this operation on specified object(s). You must be granted access to these objects in your user scope.

Operation %s does not support search filters. Do not provide search filters for this operation.

### Report Storage Limit

Your [subscription|user] storage limit of <NB> Mb has been reached. Delete existing reports and try again.

