# Qualys API Limits

**July 10, 2014**

## Overview

The Qualys API enforces limits on the API calls a customer can make based on their subscription settings, starting with Qualys version 6.5. The limits apply to the use of all Qualys APIs except "session" V2 API (session login/logout). Default API control settings are provided by the service. Note these settings may be customized per subscription by Qualys Support.

This document describes the API limits, how they are implemented by the service, and how you can track API usage and view recent API calls, including blocked calls, within the Qualys user interface.

## API Control Settings

Important!  All API controls are applied on a subscription basis.

Concurrency Limit per Subscription (per API): The maximum number of concurrent API call instances allowed within the subscription for each API. Default is 2.

Rate Limit per Subscription (per API): The maximum number of API calls allowed within the subscription per day (or a customized period, in seconds) for each API. The rate limit is calculated based on the settings for rate limit count and period.

- Rate Limit Count per Subscription (per API): The maximum number of API calls allowed within the subscription during the configured rate limit period. Default is 300.

- Rate Limit Period per Subscription (in seconds, per API): The period of time, in seconds, that defines a window when API calls are counted within the subscription for each API. Default is 86400 seconds (24 hours). The window starts from the moment each API call is received by the service and extends backwards 24 hours (or some customized period, in seconds).

Partners have the ability to view the subscription settings by logging into Qualys Admin and editing the subscription account. The API limits settings appear in the Subscription section under API Control.

## Implementation

When an API call is received, the service first checks the concurrency limit; and if the concurrency limit has been exceeded the API call is blocked and an error is returned. In the case where the concurrency limit has not been exceeded, the service checks the rate limit; and if the rate limit has been exceeded the API call is blocked and an error is returned.

Next we will describe how the API limits are implemented.

## Concurrency Limit

The API concurrency is calculated each time an API call is received and checked against the concurrency limit (2 by default) for the subscription.

When a user makes an API for an API that has 2 concurrent API call instances already running, then the new API call is blocked, a Concurrency Limit Exceeded error is reported in the XML output (see Errors Returned in XML Output), and an entry is added to the Qualys Activity (see Activity Log) like this: “API blocked (concurrency): asset_group.php”

Example: A subscription has the default API control settings and there are multiple users. A user makes 2 asset_group.php API calls and both API call instances are running. The asset_group.php API concurrency limit has been reached, so it’s not possible for any user to make another successful asset_group.php API call until at least 1 asset_group.php API call instance completes. There must be a maximum of 1 asset_group.php API call instance running at the time the user makes a new asset_group.php API call.

## Rate Limit

The rate count and period are calculated dynamically each time an API call is received. The rate period represents a rolling window when API calls are counted.

The default of 300 works out to an average of 1 API call every 5 minutes over the default 24 hour rate limit period plus more, but the user may distribute the quota of 300 API calls arbitrarily within that time window as follows:

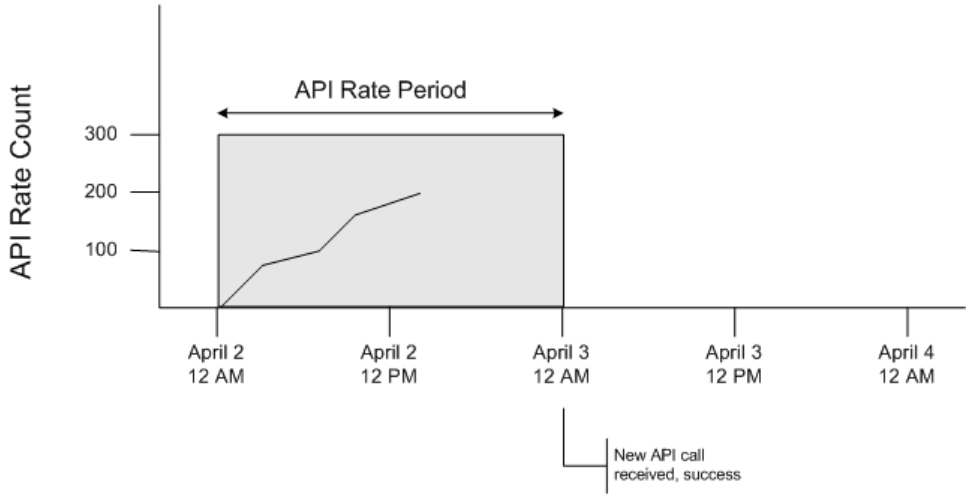
- If 300 API calls are received in a 5 minute period and none are blocked by any API limiting rules, then you need to wait 23 hours and 55 minutes before making the next call to the API. During the wait period API calls will be blocked by the rate limiting rule.

- If 1 API call is made each minute and every API call completes in less than 1 minute, the first API call will run in 4 hours and 48 minutes. Subsequent API calls will be blocked by the rate limiting rule until another 19 hours and 12 minutes have passed.

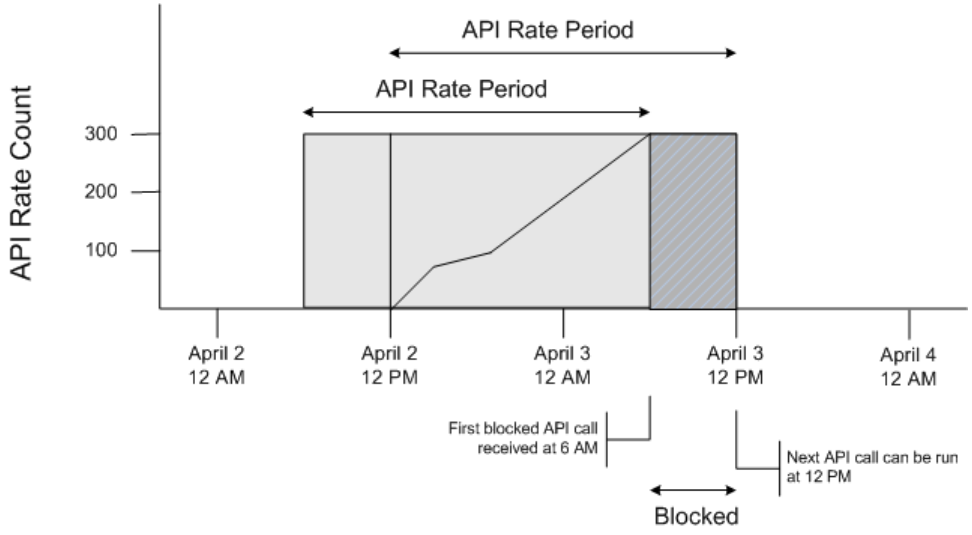
When a user makes an API call for an API that is blocked due to exceeding the rate limit, a Rate Limit Exceeded error appears in the XML output (see Errors Returned in XML Output), and an entry is added to the Qualys Activity Log (see Activity Log) like this: “API blocked (rate): asset_group.php”

Please see the examples on the next page.

Example 1: The diagram below shows the API call history for a particular API for a subscription with the default API limits. An API call was received on April 3 at 12 AM. The service calculated the API rate period by creating a window that extends backwards 24 hours from the time the API call was received to April 2 at 12 AM. The total number of API calls received in the window is 200 so the API call instance received on April 3 at 12 AM runs successfully.




Example 2: The diagram below shows the API call history for a particular API for a subscription with the default API limits. 300 API calls were received starting April 2 at 12 PM. The first blocked API call was received on April 3 at 6 AM. Users could not run API calls for 6 hours. The next time an API can be received and run is April 3 at 12 PM, assuming there is a maximum of 1 API call instance currently running at that time.



## Errors Returned in XML Output

Each API call returns an informational message in the XML output when the API call was blocked because the concurrency limit or rate limit has been exceeded for the API being called. Please note if an API call was blocked, only one error is returned. In the case where the concurrency limit has been exceeded, a Concurrency Limit Exceeded error will be reported (and a Rate Limit Exceeded error will not be reported).

### Concurrency Limit Exceeded Error

An API call returns this error in the XML output in the case where a user makes an API call and the total number of concurrent API instances, which are currently running, exceeds the limit for the subscription.

For a V1 API, the error will appear like this:

```
<GENERIC_RETURN>
  <API name="asset_group_list.php" username="acme_es1" at="2009-04-12T14:52:39Z" />
  <RETURN status="FAILED" number="1999">
    This API cannot be run again until 1 currently running API instance has finished.
  </RETURN>
</GENERIC_RETURN>
```

For a V2 API, the error will appear like this:

```
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2009-04-12T14:52:39Z </DATETIME>
    <CODE>1960</CODE>
    <TEXT> This API cannot be run again until 1 currently running API instance has finished.</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>CALLS_TO_FINISH</KEY>
        <VALUE>2</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

### Rate Limit Exceeded Error

An API call returns this error in the XML output in the case where a user makes an API call and the rate limit for the API, as defined for the subscription, has already been reached. In other words, the rate limit count (maximum number of API call instances) has already been reached for the rate limit period.

For a V1 API, the error will appear like this:

```
<GENERIC_RETURN>
  <API name="asset_group_list.php" username="acme_es1" at="2009-04-12T14:52:39Z " />
  <RETURN status="FAILED" number="1999">
    This API cannot be run again for another 2 days, 23 hours, 57 minutes and 54 seconds.
  </RETURN>
</GENERIC_RETURN>
```

For a V2 API, the error will appear like this:

```
<SIMPLE_RETURN>
  <RESPONSE>
    <DATETIME>2009-04-12T14:52:39Z </DATETIME>
    <CODE>1965</CODE>
```

```
    <TEXT> This API cannot be run again for another 2 days, 23 hours, 57 minutes and 54 seconds.</TEXT>
    <ITEM_LIST>
      <ITEM>
        <KEY>SECONDS_TO_WAIT</KEY>
        <VALUE>68928</VALUE>
      </ITEM>
    </ITEM_LIST>
  </RESPONSE>
</SIMPLE_RETURN>
```

# API Usage

Your subscription's API usage and quota information is exposed in the HTTP response headers generated by Qualys APIs (all APIs except "session" V2 API).

## HTTP Response Headers

The HTTP response headers generated by Qualys APIs are described below.

Note: The HTTP status code "OK" (example: "HTTP/1.1 200 OK") is returned in the header for normal (not blocked) API calls. The HTTP status code "Conflict" (example: "HTTP/1.1 409 Conflict") is returned for API calls that were blocked.

| HEADER | DESCRIPTION |
|---|---|
| X-RateLimit-Limit | Maximum number of API calls allowed in any given time period of <number-sec> seconds, where <number-sec> is the value of X-RateLimit-Window-Sec. |
| X-RateLimit-Window-Sec | Time period (in seconds) during which up to <number-limit> API calls are allowed, where <number-limit> is the value of X-RateLimit-Limit. |
| X-RateLimit-Remaining | Number of API calls you can make right now before reaching the rate limit <number-limit> in the last <number-sec> seconds. |
| X-RateLimit-ToWait-Sec | The wait period (in seconds) before you can make the next API call without being blocked by the rate limiting rule. |
| X-ConcurrencyLimit-Limit | Number of API calls you are allowed to run simultaneously. |
| X-ConcurrencyLimit-Running | Number of API calls that are running right now (including the one identified in the current HTTP response header). |

## Sample HTTP Response Headers

### Sample 1: Normal API call (API call not blocked)

Returned from API call using HTTP authentication.

```
HTTP/1.1 200 OK
Date: Fri, 22 Apr 2011 00:13:18 GMT
Server: qweb
X-RateLimit-Limit: 15
X-RateLimit-Window-Sec: 360
X-ConcurrencyLimit-Limit: 3
X-ConcurrencyLimit-Running: 1
X-RateLimit-ToWait-Sec: 0
X-RateLimit-Remaining: 4
Transfer-Encoding: chunked
Content-Type: application/xml
```

### Sample 2: API Call Blocked - Rate Limit exceeded

Returned from API call using HTTP authentication.

```
HTTP/1.1 409 Conflict
Date: Fri, 22 Apr 2011 00:13:18 GMT
Server: qweb
X-RateLimit-Limit: 15
X-RateLimit-Window-Sec: 360
X-ConcurrencyLimit-Limit: 3
X-ConcurrencyLimit-Running: 1
X-RateLimit-ToWait-Sec: 181
X-RateLimit-Remaining: 0
Transfer-Encoding: chunked
Content-Type: application/xml
```

### Sample 3: API V2 Call Blocked - Concurrency Limit exceeded

Returned from API V2 call using API V2 session authentication.

```
HTTP/1.1 409 Conflict
Date: Fri, 22 Apr 2011 00:13:18 GMT
Server: qweb
Expires: Mon, 24 Oct 1970 07:30:00 GMT
Cache-Control: post-check=0,pre-check=0
Pragma: no-cache
X-RateLimit-Limit: 15
X-RateLimit-Window-Sec: 360
X-ConcurrencyLimit-Limit: 3
X-ConcurrencyLimit-Running: 3
Transfer-Encoding: chunked
Content-Type: application/xml
```

Note: In the case where the concurrency limit has been reached, no information about rate limits will appear in the HTTP headers.

## Activity Log

The Qualys Activity Log shows details about user activities including actions taken using the user interface and the API. To view this section, log into your QualysGuard account and go to Users > Activity Log.

User Permissions: The Activity Log entries you see depends on your user role. For Managers, this list includes actions performed by all users in the subscription. For Unit Managers, this list includes actions performed by all users within their business unit. For Scanners and Readers, this list includes the user's own actions only. Auditors view compliance actions performed by all users.

Look at the Details column to see whether an API limit was exceeded.

**Concurrency Limit Exceeded:** For an API call that exceeded the Concurrency Limit, the details entry is in the format: "API blocked (concurrency): <API name>"

**Rate Limit Exceeded:** For an API call that exceeded the Rate Limit, the details entry is the format: "API blocked (rate): <API name>"

Go to **Filters** > **Recent API Calls** to view the API Processes list. The list includes APIs subject to the API limits (all APIs except “session” V2 API). By default the service displays API calls submitted (by users) and/or updated (by the service) in the past week.

The value “-“ appears for User Login when you are restricted from viewing this user information due to your account settings. The value “-“ appears if these three conditions are met: 1) Your user role is Unit Manager, Scanner or Reader, 2) The user who performed the API call is not in your business unit, and 3) Your subscription has this user permission selected: “Restrict view of user information for users outside of business unit”. To view this permission setting, go to Setup > User Permissions.

The value for State will be one of the following: Queued, Running, Expired, Finished (means API call completed successfully), Blocked (Rate) or Blocked (Concurrency).

User Permissions: Managers view all API calls performed by all users in the subscription. Unit Managers, Scanners and Readers view all vulnerability management API calls; and these sub-accounts view compliance and WAS API calls when the corresponding modules are enabled for their account. For example, if a Scanner's account has the compliance module enabled and it does not have the WAS module enabled, then this user will view vulnerability management API calls and compliance API calls. Auditors view only compliance API calls for compliance reporting and host management.