



# **FORUM SENTRY™ VERSION 9**

## **API CACHING**



### **Legal Marks**

No portion of this document may be reproduced or copied in any form, or by any means – graphic, electronic, or mechanical, including photocopying, taping, recording, or information retrieval system – without expressed permission from Forum Systems, Inc.

FORUMOS™ Firmware, Forum Systems XMLSec™ WebAdmin, Forum Systems XML Security Appliance™, Forum Sentry™, Forum Presidio™, Forum XWall™ Forum Sentry™ Web Services Gateway, Forum Presidio™ OpenPGP Gateway, Forum FIA Gateway™, Forum XWall Type-PCI™, Forum XWall® Web Services Firewall and Forum XRay™ are trademarks and registered trademarks of Forum Systems, Inc.

All other products are trademarks or registered trademarks of their respective companies.

Copyright © 2002-2024 Forum Systems, Inc. – All Rights Reserved.

Forum Sentry™ API Caching Guide, published May 2024.

D-ASF-SE-01904

## Table of Contents

INTRODUCTION TO THE API CACHING GUIDE .....	1
Audience for the API Caching Guide .....	1
Conventions Used .....	1
API CACHING OVERVIEW .....	1
CACHE POLICY .....	2
Terms and Definitions .....	2
Cache Timeout Directive vs. Expires Header .....	3
SUPPORTED CONTENT POLICIES .....	3
CACHE METER .....	3
Tiles .....	3
Terms and Definitions .....	3
Graph .....	4
Table .....	4
Terms and Definitions .....	4
Enable API Caching .....	4



# INTRODUCTION TO THE API CACHING GUIDE

## Audience for the API Caching Guide

The *Forum Systems Sentry™ Version 9 API Caching Guide* for System Administrators who will:

- Minimize latency between client and server
- Create, read, update or delete Cache Policies
- Monitor cache entries
- Attach Cache Policies to Content Policies

## Conventions Used

A red asterisk ( \* ) aligned with a field term means that this field is required. In this and other documentation, the Web Administration Interface is referred to as the WebAdmin and the Forum XML Security Appliance™ is referred to as the 'device', 'product' or 'system'.

In this document, all data or commands that must be entered or selected are displayed in boldface. Example:

User name: **johnsmith**  
Password: **\*\*\*\*\***

UI screens, which display a STATUS column, represent the following states:

- Green status light = enabled policy.
- Red status light = disabled policy.

Throughout this and other documents in the Documentation Set, repetitive actions such as:

- View / edit a policy.
- Enable / disable a policy.
- Delete a policy.
- Rename a policy.
- Limit display of policies with Search or Max Results fields.

are not shown. For more information, refer to the Common Operations section of the *Forum Systems Sentry™ Version 9 Web-based Administration Guide*.

Customers with plug-in licenses on the system, confirm that your Integration License is visible on the General Info screen under the SUPPORTED FEATURES section. For using the Secure Edge Cache Policies Guide, you should have the Caching Server appear under the SUPPORTED FEATURES.

Any specifications and constraints referenced in this volume appear in the Appendices of this volume.

## API CACHING OVERVIEW

Deeper integration with systems and mobile users leads to ever increasing network traffic. Whenever information is requested by an external partner, a series of on service invocations occur that traverse deep into your legacy back office systems. Such multi-hop invocations happen regardless of whether the information being request actually changes. To reduce network traffic, improve response time and optimize critical business resources, Forum Systems provides caching technology for modern-day contextual information flows enabling business-aware caching capabilities.

## CACHE POLICY

The purpose of the Cache Policy is to associate a cache configuration to a given Content Policy. For example, to optimize fetching resources from a server, one would attach and enable the Cache Policy to a Content Policy.

Location: Gateway → Network Policies → Cache Policies

### Terms and Definitions

TERM	DEFINITION
Name	The identifier for this Cache Policy.
Labels	A tag used to visually organize Cache Policies on the Cache Policy list screen.
Cache Timeout	A directive that defines when the response is considered stale.
Extend Timeout	Extends the cache timeout until the specified time
HTTP Methods	The following HTTP methods are supported: <ul style="list-style-type: none"><li>• The <b>GET</b> method requests a representation of the specified resource.</li><li>• The <b>POST</b> method is used to submit an entity to the specified resource.</li><li>• The <b>Any</b> method is used to disable the method check and allow any methods</li></ul>
Compression Enabled	Determines whether a compressed request is cached. Note: This setting is <u>not</u> the compression of the cached data
Allow URL Query Parameter Caching	Determines whether a transaction should be cached based on if a query parameter is present in the URL.  Checkbox state: <ul style="list-style-type: none"><li>• A <b>True</b> state will always cache a response regardless if query parameter is present.</li><li>• A <b>False</b> state will only cache a response if no query parameter is present.</li></ul>
Enforce Headers	The following enforce headers are supported: <ul style="list-style-type: none"><li>• The <b>Cache-Control</b> is a general-header field that specifies directives for caching. The Cache-Control was introduced in HTTP/1.1. The three directive categories are cache ability, expiration and revalidation/reloading. The use is encouraged to do further independent reading to understand the nuances between each directive.</li><li>• The <b>Expires</b> header defines the date/time in which the response is considered stale. Note, the server determines the date/time.</li><li>• The <b>Pragma</b> header is defined for backwards compatibility with HTTP/1.0. Even though Pragma is similar to Cache-Control, Pragma is not a replacement and should only be used if absolutely necessary.</li></ul>
Content Types	Defines the content-types that are designated to be cached. Checking the <b>Any</b> checkbox will disable this check.
Use Redis	Check this box to have the caching use a Redis Server. When Redis is being used, there is both local caching and redis caching active.
Redis Policy	This is the Redis Policy that points to a Redis Server or Redis Cluster to be used for caching.

## Cache Timeout Directive vs. Expires Header

There is a subtle difference between the Cache Timeout directive and Expires header the reader needs to understand to properly configure a Cache Policy. The server generates the Expires header and if the Cache Policy is configured to enforce the Expires header, Sentry honors the date/time and expires the cache entry accordingly. The Cache Timeout defines how long the cache entry is valid for starting from the time the cache entry is initially stored. With both properties configured, Sentry honors the minimum of between Cache Timeout directive and Expires header.

For example, if server defines a cache entry to expire at 'Wed, 21 Oct 2015 07:28:00 GMT' and the Cache Policy is configured to Cache Timeout 10 seconds and Expires is set to true, the following table demonstrates the cache entry expiration behavior.

Given:

- Expires: Wed, 21 Oct 2015 07:28:00 GMT
- Cache Timeout: 10 seconds

Results:

Initial Time	Request Time	Expired
Wed, 20 Oct 2015 08:11:41 GMT	Wed, 20 Oct 2015 08:11:43 GMT	False
Wed, 21 Oct 2015 08:11:41 GMT	Wed, 21 Oct 2015 08:11:43 GMT	True
Wed, 21 Oct 2015 07:27:49 GMT	Wed, 21 Oct 2015 07:27:59 GMT	True
Wed, 21 Oct 2015 07:27:55 GMT	Wed, 21 Oct 2015 07:28:01 GMT	True

## SUPPORTED CONTENT POLICIES

The following Content Policies are supported:

- XML Policies
- REST Policies
- JSON Policies
- HTML Policies

## CACHE METER

The purpose of the Cache Meter is monitoring and deleting real time cache data. The Cache Meter is broken up into three components: tiles, graph and table. Tiles represent a global snapshot of the overall state of API Caching. The graph depicts a timeline of cache events. The table allows for searching and removal of cache entries.

Location: Diagnostics → Monitoring → Cache Meter

### Tiles

The purpose of the tiles is to give a global snapshot of API Caching.

### Terms and Definitions

TITLE	DEFINITION
Hits	The global number of cache hits. Alternatively, the global number of times the server was not called.
Misses	The global number of cache misses. Alternatively, the global number of time the server was called.
Purges	The global number of times a cache entry is forced cleared.

Hit Rate	The global percentage of hits. Alternatively, Hit Rate describes how often a hit occurs upon a request.
----------	---

**Note:** Hits plus Misses equals the total requests with API Caching attached. Hits divided by total requests equals the Hit Rate.

## Graph

The Cache Meter graph allows for visualization of real time caching data. To change between timing intervals, select the dropdown and choose an appropriate duration.

## Table

The Cache meter table is where the user has fine-grained control of the current state of the cache. The user can search by keywords, sort by columns or remove entries in this component.

## Terms and Definitions

TERM	DEFINITION
Key	The identifier for the given cache entry.
Hits	The number of cache hits for the given cache entry.
Misses	The number of cache misses for the given cache entry.
Hit Rate	The percentage of hits for the given cache entry.
Purges	The number of times a clear occurred for cache entry.
Expiration	The date/time the cache entry is set to expire.

## Enable API Caching

In API Security Gateway, you can currently enable caching for XML, HTTP, JSON and REST Content Policies.

### To configure API Caching:

1. Go to Gateway → Network Policies → Cache Policies.
2. Ensure a Cache Policy exists. If not, create one.
3. Go to Gateway → Content Policies → Click on desired Content Policy (XML, REST, JSON or HTML). If a Content Policy does not exist, create one.
4. Click a Content Policy.
5. Click the Settings tab.
6. Check the Enable Response Caching checkbox.
7. Choose Cache Policy to attach to Content Policy.