



# **FORUM SYSTEMS HANDS-ON TRAINING**

## **LAB 6. DEPLOYING A REST API THROUGH FORUM SENTRY**



# FORUM SYSTEMS

A Crosscheck Networks Company

## 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.

FORUMOST™ 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 © 2002-2014 Forum Systems, Inc. – All Rights Reserved.

Published: September 2014

Forum Systems Hands-on Training – Lab 6. Deploying a REST API Through Forum Sentry  
D-ASF-SE-010029

## Contents

Introduction.....	4
<i>Skill Level</i> .....	4
<i>Prerequisites</i> .....	4
<i>Lab Overview</i> .....	4
Forum Sentry REST Policies .....	5
<i>Network Policies</i> .....	5
<i>Virtual Directories</i> .....	5
<i>Building a REST Policy</i> .....	5
<i>Testing a REST Policy</i> .....	7
Transactions in the Sentry Logs .....	8
Additional Testing and More Reading.....	11
<i>BACK IT UP!</i> .....	11
<i>Additional Tests</i> .....	11
<i>Additional Information</i> .....	11
About Forum Systems.....	12

## Introduction

Lab 6. Deploying a REST API through Forum Sentry

### Skill Level

This lab is beginner skill level. Little to no prior experience with Forum Sentry or SOAPSonar is required.

### Prerequisites

This lab requires the Forum Sentry Training Image, with a licensed copy of SOAPSonar Enterprise Edition.

Refer to the “FS\_Training\_Labs\_v8-1\_Introduction” document for information on the Forum Sentry Training Image and licensing SOAPSonar Enterprise Edition.

The online OpenWeatherMap REST API used in Lab 3 is utilized in this lab. See Lab 3 for the APPID key and information on accessing this REST API.

General knowledge of testing a REST service using SOAPSonar is assumed. This is covered in Lab 3 of this training series.

The Sentry instance used in this lab needs internet access to the online OpenWeatherMap REST API at <http://openweathermap.org/api>. This access is available with the online Sentry Training Image.

### Lab Overview

This lab provides instructions for deploying the online OpenWeatherMap REST API, tested in Lab 3, through Forum Sentry.

Prior to this API being deployed through Sentry, clients access the API directly. A primary function of an API Gateway is to broker the traffic - behaving as a reverse proxy - so that the client cannot access the API directly.

Once the API is deployed through Sentry, the clients should never have any access to the actual service. As far as the client is concerned, the Sentry endpoint is the API.

This lab will provide instructions for deploying a REST API with Forum Sentry. Topics will include:

1. REST Policies
2. Network Policies
3. Reviewing transactions in the Sentry System and Access Logs

## Forum Sentry REST Policies

A REST API is deployed in Sentry through a REST Policy. A REST Policy is a type of Content Policy. Other Content Policies include: XML Policies, JSON Policies, HTML Policies, STS Policies, and OAuth policies.

The key components to a REST Policy are:

1. Network Policies
2. Virtual Directories

## Network Policies

While building a REST Policy in Sentry, you will also build the associated Network Listener and Network Remote Policies. The network policies are essentially the “plumbing” or network framework that are used to get the runtime traffic into Sentry for processing, and then sent out to the remote server. We will be working with two types of network policies in this lab:

1. Listener Policies – Listen on IP/Port and is used to accept incoming traffic
2. Remote Policies – Define where to send the processed request

## Virtual Directories

The virtual directory is built as the REST Policy is built. The virtual directory ties together the network listener and network remote policies. A REST Policy may have multiple Virtual Directories. For instance, one virtual directory may be used for authentication and another for the actual API.

The Virtual Directory is also where many settings for the REST policy are modified including:

1. Virtual Path – The virtual URI for the API (the URL that the clients will use to access the service)
2. Remote Path – The URI for the remote API endpoint
3. Filter Expression – The characters allowed in the request URI following the path
4. Many more – Policy level authentication, the HTTP content-types, HTTP methods allowed, etc...

## Building a REST Policy

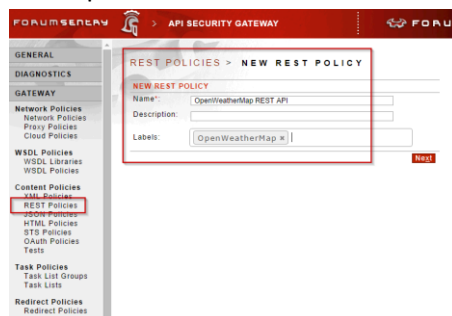
Unlike a WSDL Policy, there is no import required to generate a REST Policy.

In this step we will build a REST Policy and in doing so also create:

1. The network policies for this API
2. The virtual directory for this API

### Follow the steps below to build a REST Policy in Sentry

1. Navigate to the Gateway→Content Policies→REST Policies page and click New
2. Name the policy “OpenWeatherMap REST API” and add the label “OpenWeatherMap”.



3. On the next screen, you are prompted to select existing remote policies or to build new listener and remote policies, and to enter the virtual path. Create new network policies rather than reusing any existing policies. Enter the following information to build the REST Policy for the OpenWeatherMap API:
  - a. Check the "Use Device IP" box
  - b. Set the listener port to: 82
  - c. Set the Virtual Directory Path to: /data/2.5/weather
  - d. Set the Remote Policy Host to: api.openweathermap.org
  - e. Set the Remote Policy Port to: 80

**FORUMSENTRY** > **API SECURITY GATEWAY** > **FORUMSYSTEM**

**REST POLICIES > NEW REST POLICY**

**SET LISTENER POLICY**  
Please specify a listener policy for virtual directory: New Virtual Directory

☐ Select from existing listener policies  
SampleWS-Listener (0.0.0.0:81) [Edit](#)

☒ Create a new HTTP listener policy

Listener Policy Name\*: OpenWeatherMapRESTAPI-Listener

Use Device IP: ☒

Listener IP\*: 172.31.3.22

Listener Port\*: 82

**SET VIRTUAL DIRECTORY PATH**  
Virtual Directory Path: /data/2.5/weather

**SET REMOTE POLICY**  
Please specify a remote network policy

☐ Do not send to remote server

☐ Select from existing remote policies  
SampleWS-Remote (localhost:8080) [Edit](#)

☒ Create a new HTTP remote policy for this remote server

Remote Policy Name\*: OpenWeatherMapRESTAPI-Remote

Remote Policy Host\*: api.openweathermap.org

Remote Policy Port\*: 80

4. After clicking Finish, the REST Policy is built. Notice there is a single Virtual Directory, with the Virtual URI being the Sentry endpoint (what the clients will use) and the Remote URI is the OpenWeatherMap API (where Sentry will send the processed request).

**REST POLICIES > REST POLICY**

**REST POLICY**  
Policy Name: OpenWeatherMap REST API

Virtual Directories | Task Lists | Settings | IDP Rules | Logging

VIRTUAL DIRECTORY	STATUS	VIRTUAL URI	REMOTE URI
New Virtual Directory	●	http://172.31.3.22:82/data/2.5/weather	http://api.openweathermap.org:80/data/2.5/weather

[Enable](#) [Disable](#) [Delete](#) [New](#)

5. Access the Virtual Directory page by clicking the link “New Virtual Directory”
  - a. Rename the Virtual Directory to “OpenWeatherMap REST API”
  - b. Modify the Filter Expression field to a wildcard value of .\* which is required so that additional characters in the URI are allowed by Sentry
  - c. Scroll down and click Save

The screenshot shows the 'REST POLICIES > REST POLICY' configuration page in Sentry. The 'Policy Name' is 'OpenWeatherMap REST API'. Below the tabs (Virtual Directories, Task Lists, Settings, IDP Rules, Logging), the 'VIRTUAL DIRECTORY' section is active. It shows the 'Name' as 'OpenWeatherMap REST API' and the 'Filter Expression' as '.\*', both highlighted with red boxes. Other fields include 'Description', 'Listener Policy' (OpenWeatherMapRESTAPI-Listener), 'Virtual Host', 'Virtual Path' (/data/2.5/weather), 'Virtual URI' (http://172.31.3.22:82/data/2.5/weather.\*), 'Replace Expression' (\$0), and 'Remote Policy' (OpenWeatherMapRESTAPI-Remote). Checkboxes for 'Use virtual host as a regular expression' and 'Enable Virtual Path Case Insensitivity' are present. The 'Send to remote server' checkbox is checked.

6. You have now successfully deployed the OpenWeatherMap REST API through Forum Sentry.

## Testing a REST Policy

Now that OpenWeatherMap REST API has been deployed through Sentry, the clients should only access this service through Sentry.

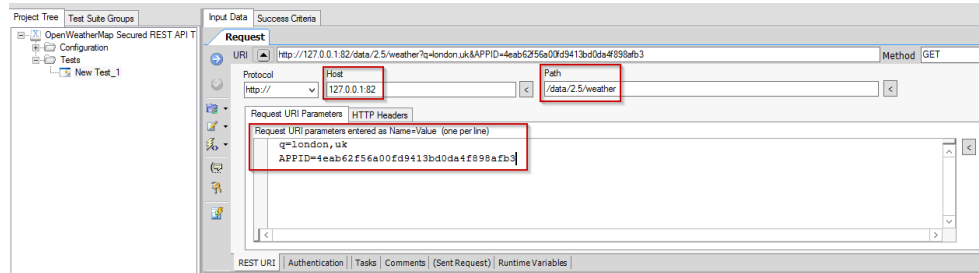
**Follow the steps below to test this Sentry secured API using SOAPSonar.**



1. Launch SOAPSonar and build a new REST Project.
  - a. In the Project QuickStart menu choose Create a REST Project
  - b. Name the project “OpenWeatherMap Secured REST Test Project”

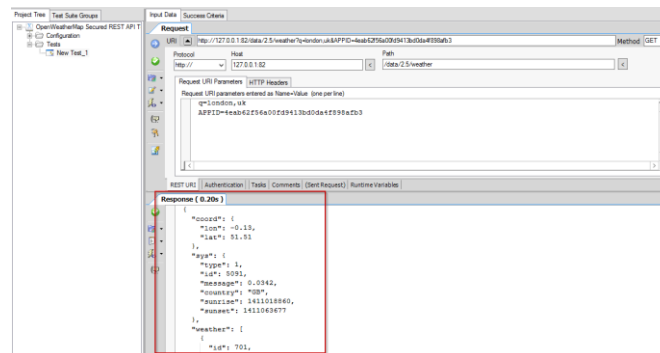
The screenshot shows the 'Project QuickStart' dialog box in SOAPSonar. It has several buttons: 'Create a WSDL SOAP Project', 'Create a REST Project' (highlighted with a red arrow), 'Create a Batch Test Project', 'Create an XML Project', 'Create a JSON Project', and 'Import Proxy Server Test'. Below these, there is a 'Project name' dialog box with the text 'Please enter a name for your project' and a text input field containing 'OpenWeatherMap Secured REST API Test Project'.

2. Build the REST Test case in SOAPSonar using the information listed below. You'll notice the Request URI field will update automatically.
  - a. **Protocol:** HTTP
  - b. **Host:** 127.0.0.1:82 (you can also specify the IP or hostname of the Sentry listener)
  - c. **Path:** /data/2.5/weather
  - d. **Request URI Parameters / Name=Value pairs (one per line):**

```
q=London,uk
appid= 4eab62f56a00fd9413bd0da4f898afb3
```



3. Click the  to commit the settings, and send the request using the . You will receive a JSON response with the weather data requested for the region entered.



## Transactions in the Sentry Logs

Now that you are able to send the OpenWeatherMap REST API calls through Sentry, it is helpful to review a successful transaction at DEBUG level. The runtime traffic can be seen in both the Access and System log. Follow the steps below to enable DEBUG level logging and then review a successful transaction in the Sentry Access and System logs.

**Follow the steps below to review the runtime transactions in the Sentry logs:**

1. Navigate to the Diagnostics→Logging→Settings page
2. Set the log level for the Audit, System, and Access logs to DEBUG and click Save



3. Send another request from SOAPSonar through Sentry
4. Navigate to the Diagnostics→Logging→Internal Logs page
5. Open the Access Log for Today (hint, right click and open the log in another browser tab)

INTERNAL LOGS							
<b>AUDIT LOGS</b>							
<input type="checkbox"/>	Today						Download (175B)
<input type="checkbox"/>	Sep 17, 2014						Download (5.3KB)
<input type="checkbox"/>	Sep 4, 2014						Download (1.6KB)
<b>SYSTEM LOGS</b>							
<input type="checkbox"/>	Today						Download (1.1KB) X
<input type="checkbox"/>	Sep 17, 2014						Download (6.5KB)
<input type="checkbox"/>	Sep 4, 2014						Download (5.2KB)
<b>ACCESS LOGS</b>							
<input type="checkbox"/>	Today						Download (318B) X
<input type="checkbox"/>	Sep 17, 2014						Download (1.8KB)
<input type="checkbox"/>	Sep 4, 2014						Download (1.6KB)

6. Notice that all of the previous runtime transactions are listed, one per line. The line shows the time of the request, the Session ID, the Client IP, the incoming host header, the HTTP method used, the URI (virtual directory triggered), the status code, and the request length for the transaction.

INTERNAL LOGS > ACCESS LOG

SEP 18, 2014

Search:  Search

Refresh:  (0-30 secs)

Filter By Log Level:

2 items found, displaying all items.1

Time	Session	IP	Host Header	Type	URI	Code	Length
00:13:25.267	X000004	127.0.0.1	127.0.0.1:81	POST	/SampleWS.asmx	200	346
00:01:38.573	X000003	127.0.0.1	127.0.0.1:81	POST	/SampleWS.asmx	500	814

- Click the most recent Session ID (at top by default) to jump to the System log.
- The System log will be sorted only showing log messages for this Session (transaction).

INTERNAL LOGS > SYSTEM LOG

SEP 18, 2014

Search:  X000006 Search

Refresh:  (0-30 secs)

Filter By Log Level:

Filter By Policy Name:

31 items found, displaying all items.1

ID	Time	Session	Code	Level	Message
0000F4	00:21:25.326	X000006	08402	D	Document left Communications Layer
0000F3	00:21:25.326	X000006	0840C	D	Sending client a raw response: Status Code: 200 Header Info: ...
0000F2	00:21:25.325	X000006	09334	D	Adding Via header to response

- The most recent log messages are listed at the top of the log by default. Scroll to the bottom to see the first log message for the transaction “Document entered communication layer” and read up from there to see all processing done in Sentry.

0000EB	00:21:25.324	X000006	09211	D	Received an HTTP response: Protocol: HTTP/1.1 Response Code: 200 ...
0000EA	00:21:25.320	X000006	0840B	D	Sending remote server a processed request: Method: POST Remot...
0000E9	00:21:25.319	X000006	09332	D	Adding Via header to request
0000E8	00:21:25.319	X000006	09330	D	Stored header suppressed from proxying - content-length: 377
0000E7	00:21:25.319	X000006	09330	D	Stored header suppressed from proxying - connection: keep-alive
0000E6	00:21:25.319	X000006	09330	D	Stored header suppressed from proxying - host: 127.0.0.1:81
0000E5	00:21:25.319	X000006	09003	D	Connecting to back end server at URL 'http://localhost:8080/SampleS.asmx'
0000E4	00:21:25.318	X000006	0E10B	D	No TaskListener configured, document will not be processed
0000E3	00:21:25.318	X000006	0E20A	D	Document left WSDL validation
0000E2	00:21:25.318	X000006	0E208	D	WSDL message: MultiplySoapIn
0000E1	00:21:25.318	X000006	0E221	D	ACL check skipped - no ACL associated with operation 'Multiply'.
0000E0	00:21:25.318	X000006	0E209	D	Document entered WSDL validation
0000DF	00:21:25.318	X000006	0E207	D	Matched WSDL operation 'Multiply(MultiplySoapIn)'
0000DE	00:21:25.317	X000006	08407	D	Request document: <?xml version="1.0" encoding="utf-8"?> <soap:Envelop...
0000DD	00:21:25.317	X000006	09604	D	Simple decode succeeded
0000DC	00:21:25.317	X000006	09607	D	Decoding a document of 377 bytes
0000DB	00:21:25.316	X000006	09410	D	Message type filter match succeeded - matched filter 'SOAP 1.1 Filter' of type Simple
0000DA	00:21:25.316	X000006	08016	D	ACL check skipped - no ACL associated with virtual directory '/SampleWS.asmx/'
0000D9	00:21:25.316	X000006	08016	D	ACL check skipped - no ACL associated with network policy 'SampleWS-Listener'
0000D8	00:21:25.316	X000006	0915C	D	Processing request for 'WSDL Policy: 'SampleWS''
0000D7	00:21:25.316	X000006	09140	D	Received an HTTP request: Protocol: HTTP/1.1 Scheme: http ...
0000D6	00:21:25.314	X000006	08401	D	Document entered Communications Layer

- Note that the “connecting to remote server” line is very important. This indicates that Sentry has successfully processed the request message and is now forwarding it to the remote server. If there are any errors returned for this transaction, the failure did not occur during request processing but rather either while connecting to the remote server, in processing on the remote server, or in the Sentry processing of the response document.

END

## Additional Testing and More Reading

### BACK IT UP!

It is recommended that you export your REST Policy and/or your full Sentry configuration after completing this lab. To export the REST policy, navigate to the REST Policies page, select the REST policy and use the GDM Export option to export the policy (and all dependencies) as a password encrypted FSG file. This can later be imported on the System→Configuration→Import/Export screen.

To export your full Sentry configuration, navigate to the System→Configuration→Import/Export screen and use the Export option in the center of the page to export the full Sentry configuration file as a password encrypted FSX file. This can later be imported on the same screen.

Backup your SOAPSonar project file by using the File→Save As option. All of your test cases will be saved in an .SSP file.

### Additional Tests

1. A browser can also be used to test this REST API. Simply copy/paste the Request URI from the SOAPSonar test case into the Address bar of the browser.
2. What happens if you send a request into the wrong URI?
3. What happens if you change the HTTP Method to POST?

Later labs will explore how to apply security to this REST API, including: SSL, authentication, and query parameter filtering.

### Additional Information

For more information, review the following Forum Sentry Admin Guides:

1. Network Policies Guide
2. REST Policies Guide

For more information on the OpenWeatherMap API see: <http://openweathermap.org/api>

## About Forum Systems

Forum Systems is the global leader in API and Cloud Security technology with industry-certified, patented, and proven products deployed in the most rigorous and demanding customer environments worldwide. Forum Systems has been an industry leader for over 12 years and has built the core architecture of its technology on the foundation of FIPS 140-2 and NDPP. Forum Systems security-first mindset enables trusted, network edge deployments of its technology for protecting critical enterprise transactions.

Our product technology is purpose-built and designed for mission-critical, enterprise-class scalable solutions where business solutions require the modern day security and identity enforcement protection, while enabling a scalable architecture and low-latency, high-volume throughput.

Forum Systems supports global enterprise customers across industries in commercial, government, and military sectors. Forum Systems technology provides the leading-edge of modern-day cyber-security innovation with integrated identity and SSO features that enable out-of-the box business solutions with point-and-click technology.

Forum's patented; FIPS 140-2 and NDPP certified hardware and virtual products make modern-day business communications secure by actively protecting and accelerating data exchange and API service access across networks and business boundaries. For more information, please visit [www.forumsys.com](http://www.forumsys.com).