

Version 1.0

Monitoring the Layer 7 Gateway



1.800.681.9377
info@layer7tech.com
www.layer7tech.com



Copyright © 2005-2012 Layer 7 Technologies Inc.

The Layer 7 Installation and Maintenance Manuals, the Layer 7 Policy Manager User Manual, the Layer 7 Policy Authoring User Manual, the SecureSpan™ XML VPN Client User Manual, and the Layer 7 Enterprise Service Manager User Manual are the copyright of Layer 7 Technologies Inc. All rights reserved.

SecureSpan and CloudSpan are trademarks of Layer 7 Technologies Inc. (registration pending), and is protected by law in Canada, the United States, and other countries.

All other trademarks and tradenames belong to their respective owners.

Layer 7 Technologies Inc. reserves the right to change the information in this Manual without notice. The content in this Manual is confidential. No part of this Manual may be copied, transmitted, or saved for non-personal purposes without the written permission of Layer 7 Technologies Inc.

Contents

| | |
|--|-----------|
| List of Figures | iv |
| List of Tables | iv |
| Chapter One: Introduction | 1 |
| How to Use This Document | 1 |
| Overview | 1 |
| Chapter Two: Monitoring Overview | 3 |
| Introduction | 3 |
| Available Monitoring Methods and Stakeholders | 3 |
| Monitoring Using the Policy Manager | 3 |
| Monitoring Using Remote Shell Command Line | 4 |
| Monitoring Using SNMP | 4 |
| Monitoring Using ILOM | 4 |
| Monitoring Using WSDM | 5 |
| Monitoring by Logging and Auditing | 5 |
| Monitoring Using Splunk | 5 |
| Monitoring Using Nagios | 6 |
| Conclusion | 6 |
| Chapter Three: Basic Monitoring | 7 |
| Introduction | 7 |
| Basic Monitoring Tools Included | 7 |
| Gathering Basic Information about a Gateway Service | 7 |
| Service Metrics | 9 |
| Gateway Audit Events | 11 |
| Conclusion | 13 |
| Chapter Four: Advanced Monitoring | 15 |
| Introduction | 15 |
| Using the Linux Command Line | 15 |
| Accessing the Gateway via SSH | 15 |
| Viewing Current Load with “vmstat” | 16 |
| Checking Java Processes with “ps” | 17 |
| Command Line Policy Migration Toolkit | 18 |
| Understanding Policies vs. Services | 19 |
| Using the Command Line Policy Migration Toolkit | 20 |
| Listing Services | 20 |
| Listing References | 21 |
| Creating a Map File | 22 |
| Testing the Map File | 24 |
| Reusing Map Files | 25 |
| Creating a New Service and Importing the Policy into the Target Server | 25 |

| | |
|--|-----------|
| Enabling and Disabling a Service | 27 |
| Modifying Service Properties..... | 28 |
| Conclusion | 29 |
| Chapter Five: General SNMP Monitoring..... | 31 |
| Introduction | 31 |
| About SNMP MIBs | 31 |
| Standard System MIB | 31 |
| UCD Extended MIB..... | 31 |
| Layer 7 MIB | 32 |
| SNMP Events to Monitor..... | 32 |
| Conclusion | 33 |
| Chapter Six: Monitoring Using Tactical Agent..... | 35 |
| Introduction | 35 |
| The Basic Solution | 35 |
| The Extended Solution..... | 35 |
| Deploying the Tactical SNMP Agent..... | 37 |
| Deploying the Agent..... | 37 |
| Adding the Service to the Gateway | 38 |
| Conclusion | 39 |
| Chapter Seven: Monitoring the Gateway Using WSDM | 41 |
| Introduction | 41 |
| Setting Up the Layer 7 Gateway | 41 |
| Step 1: Enable the Gateway WSDM Subscription Tool..... | 41 |
| Step 2: Link Notification Service to Subscription Manager..... | 43 |
| Step 3: Configure Notification Settings | 43 |
| Step 4: Subscribe Services..... | 44 |
| Subscribing a Service to the Gateway WSDM Subscription Manager | 45 |
| Unsubscribing a Service from the Gateway WSDM Subscription Manager | 46 |
| Renewing a Service to the Gateway WSDM Subscription Manager | 46 |
| Conclusion | 46 |
| Chapter Eight: Enterprise Logging and Auditing..... | 47 |
| Introduction | 47 |
| Components of an Enterprise Logging/Auditing System | 47 |
| Gathering Audit Data from the Gateway..... | 48 |
| Using the Audit Sink Policy | 49 |
| Manipulating/Storing the Audit Data..... | 49 |
| Sample Audit Sink Policy | 50 |
| Conclusion | 51 |
| Chapter Nine: Monitoring Using Splunk | 53 |
| Introduction | 53 |
| What is Splunk? | 53 |
| Setting Up the Splunk Server | 53 |

| | |
|---|-----------|
| Step 1: Download Splunk | 53 |
| Step 2: Install Splunk..... | 54 |
| Configuring the Server Firewall | 54 |
| Step 3: Set Up the Splunk Web Console | 55 |
| Setting Up the Splunk Forwarder | 55 |
| Step 1: Download the Splunk Forwarder..... | 56 |
| Step 2: Install the Splunk Forwarder | 56 |
| Step 3: Index the Gateway Log Files..... | 56 |
| Step 4: Configure *nix on the Gateway | 57 |
| Downloading *nix | 57 |
| Installing *nix..... | 57 |
| Step 5: Start Using Splunk | 58 |
| Troubleshooting | 58 |
| Check if Splunk Forwarder is working | 58 |
| Restart Splunk Forwarder | 58 |
| Retrieving Data for Gateway Service Statistics | 58 |
| Installing the WSDM Auto-Subscription Tool..... | 59 |
| Step 1: Install the Gateway Command Line Migration Tool | 59 |
| Step 2: Set up the WSDM Subscription Manager and WSDM Metrics Notifier..... | 59 |
| Step 3: Copy the WSDM auto-subscription file | 59 |
| Step 4: Configure the WSDM Auto-Subscription Tool..... | 60 |
| Step 5: Run script manually or create cron job..... | 62 |
| Step 6: Monitor the log file..... | 62 |
| Understanding the WSDM Auto-Subscription Tool..... | 62 |
| Third Party Licensing..... | 63 |
| Conclusion | 63 |
| Chapter Ten: Nagios Integration..... | 65 |
| Introduction | 65 |
| Setting Up Nagios..... | 65 |
| Downloading Nagios | 65 |
| Creating an Account..... | 65 |
| Installing Nagios..... | 66 |
| Installing the Plugins..... | 67 |
| Starting Nagios..... | 68 |
| Configuring Nagios..... | 68 |
| Main Configuration | 68 |
| SNMP Configuration | 69 |
| Loading Configuration Changes into Nagios | 70 |
| Using Perl Scripts as Nagios Commands..... | 70 |
| Conclusion | 70 |

List of Figures

| | |
|--|----|
| Figure 1: Published Service Properties dialog..... | 8 |
| Figure 2: Published Service Properties – [HTTP/FTP] tab | 8 |
| Figure 3: Policy Manager Dashboard | 9 |
| Figure 4: Dashboard - Message rate (requests/sec) | 10 |
| Figure 5: Dashboard - Selection metrics..... | 10 |
| Figure 6: Dashboard - Cluster Status..... | 11 |
| Figure 7: Gateway Audit Events..... | 12 |
| Figure 8: Audit record details | 12 |
| Figure 9: Layer 7 Gateway main menu | 16 |
| Figure 10: vmstat running every 2 seconds | 16 |
| Figure 11: vmstat showing a Gateway processing 650 requests/sec..... | 17 |
| Figure 12: Running the "ps -afux grep java" command..... | 18 |
| Figure 13: Published Service Properties..... | 19 |
| Figure 14: Displaying the published services on a Gateway | 21 |
| Figure 15: Publishing the Gateway Management Service | 42 |
| Figure 16: Publishing the WSDM Subscription Service | 42 |
| Figure 17: Creating a policy for WSDM Notification | 42 |
| Figure 18: Subscribing a service to the WSDM Subscription Manager | 45 |
| Figure 19: Unsubscribing a service from the WSDM Subscription Manager..... | 46 |
| Figure 20: Renewing a service to the WSDM Subscription Manager | 46 |
| Figure 21: Audit Sink Properties..... | 48 |
| Figure 22: Sample audit sink policy | 50 |
| Figure 23: The Splunk manager console | 55 |
| Figure 24: Shell script sequence diagram for retrieveListServices.sh..... | 63 |

List of Tables

| | |
|---|----|
| Table 1: New metric data available per Layer 7 Gateway published services | 36 |
| Table 2: WSDM notification settings..... | 43 |
| Table 3: Explanation of elements for working with WSDM Subscription Manager | 44 |
| Table 4: Customizing the config.pl script..... | 60 |
| Table 5: Customizing the retrieveListServices.cfg script | 61 |

Chapter One: Introduction

Appliance monitoring is an integral part of enterprise business processes. For many businesses, it is essential to monitor hardware and services available on the network. High availability of hardware and services to both internal employees and external entities is important for running a prospering business.

Working with the Layer 7 Gateway as part of your infrastructure requires some degree of monitoring the appliance's daily operations or Gateway virtual machine that you have deployed. It is important to know whether your Gateway is processing messages in a manner that meets internal service agreements and whether each service is available to accept messages.

There are several ways to monitor the Gateway appliance itself and the services that it hosts. This document will provide insight into the mechanisms that are available to monitor the Layer 7 Gateway and its services.

How to Use This Document

This document is a reference guide that does not need to be read chronologically. However, it is recommended that you read Chapter 2 of this manual, as this chapter introduces the concepts behind monitoring the Layer 7 Gateway and it provides a high level view of the monitoring process. After reading this chapter, you can reference individual chapters to refer to monitoring methods that are of specific interest.

Overview

The Layer 7 Gateway provides many options for appliance monitoring, such as creating reports for Gateway load trends, SNMP connectivity to the Gateway, command line tools to view Gateway statuses, migrate policies, and configuring services between Gateway environments. In addition, administrators who prefer GUI tools can use the Gateway's Dashboard to view current load and status of services accessed in real time.

The Layer 7 Gateway has many options for monitoring as well as integration with your enterprise operations and service level agreements.

Chapter Two: Monitoring Overview

Introduction

Whether the Layer 7 Gateway is integrated into your IT infrastructure as a hardware appliance in the rack or as a virtual machine in a cloud environment, it can be monitored to ensure proper operation.

This chapter outlines the monitoring integration points available on the Layer 7 Gateway and its various sample tools.

Before you start, you must ensure that the physical appliance is operational and that the Gateway processes are working correctly.

Available Monitoring Methods and Stakeholders

It is common for different enterprise divisions to have varying levels of responsibilities in machine monitoring. Each team in an organization may be responsible for a different aspect of the Gateway and each will use different tools to monitor the data that interests them. For example, administrators have access to configure and manage the Gateway, but may not have access to the hardware itself, while the Operations team have access to the physical hardware of the Gateway, but are not able to remotely access the Gateway's file system.

The Layer 7 Gateway can be monitored by using any of the tools listed below.

- The Policy Manager's Dashboard
- The Gateway's remote shell console via SSH
- SNMP, the Oracle Integrated Lights Out Manager (ILOM)
- WSDM

It can also be monitored via logging and auditing.

There are different tools available that integrate with these access methods in help monitor the performance of the Layer 7 hardware and services.

Monitoring Using the Policy Manager

Administrators of the Layer 7 Gateway that use the Policy Manager can monitor many aspects of the Gateway, including using the Dashboard to view real-time processing statistics of message data. A graph in the Dashboard presents a quick insight into a particular services performance. This graph exhibits how many requests a service

has processed in the last hour, day, or to an adjustable time period. The Dashboard also contains log files from Layer 7 Gateways that are available on the cluster nodes, and audit logs with data related to published services. This allows administrators to see if the Gateway is processing the messages smoothly overall.

For more information on monitoring using the Policy Manager, see Chapter 3, “Basic Monitoring”.

Monitoring Using Remote Shell Command Line

Administrators who have access to the root shell of the Layer 7 Gateway are able to use Linux command line tools to monitor its performance. Command line tools such as *vmstat*, *ps*, and *net-stat* can show how much processing power is being consumed, whether or not ports are available on the Gateway, and if sockets are being consumed and released at a normal rate. Log files can be viewed via the command line to help diagnose problems that arise during the runtime of a Gateway.

For more information on monitoring using the command line, see Chapter 4, “Advanced Monitoring”.

Monitoring Using SNMP

SNMP provides another method to monitor the Layer 7 Gateway. By using SNMP, you can monitor Gateway hardware appliance features such as CPU usage, temperatures, and fan functionality. The Layer 7 Gateway exploits the Berkeley net-SNMP MIB features for hardware monitoring.

SNMP can also be used to monitor services published on the Layer 7 Gateway. Information such as the number of policy violations in a published service, the number of requests in a policy in the last hour, and the number of failed routes in a policy are just some of the data available about services via SNMP.

For more information on using SNMP with the Gateway, see Chapter 5, “General SNMP Monitoring”.

Monitoring Using ILOM

For customers that use the Layer 7 Gateway hardware appliance hosted on an Oracle/Sun server, ILOM (Integrated Lights Out Management) is available for monitoring. ILOM is hosted on its own Ethernet port and runs on power separate from the rest of the server. It runs on its own CPU and is separated from the rest of the hardware appliance. ILOM provides a web-based user interface that can be accessed by administrators to view hardware information, monitor hardware events, and configure hardware based notifications. For example, an administrator can access the ILOM management console to configure alert emails that are to be sent when a server loses power, or if a CPU cooling fan stops working.

Since ILOM is separate from the hardware server, you can still access the ILOM management console and diagnose hardware problems even if the hardware hosting the Gateway stops working.

ILOM is an Oracle/Sun proprietary hardware management feature.

Monitoring Using WSDM

The Layer 7 Gateway provides a method of managing and monitoring the status of Gateway services using WSDM (Web Services Distributed Management). This allows the subscription of a Gateway service to an internal service manager that monitors the service (the service to monitor is specified in the subscription request). The Layer 7 Gateway Subscription manager gathers metrics data on that service and sends this gathered information to an endpoint (local or external) at regular intervals (configurable in a Gateway cluster property). The subscription manager is sophisticated enough to send metrics data about a subscribed service only if the data is actually changing (that is, only if the service is being consumed within interval times set).

For more information on using WSDM with the Gateway, see Chapter 7, “Monitoring the Gateway using WSDM”.

Monitoring by Logging and Auditing

The logging and auditing features of the Layer 7 Gateway can be used to monitor service health. Error messages and specific audit messages are output to log files on the Layer 7 Gateway. These log files can be accessed to help diagnose problems that arise with the Gateway and with services hosted on the Gateway.

Messages that are audited in Gateway policy are added to the audit log. This audit log is viewable in the Policy Manager.

For more information on logging and auditing, see Chapter 8, “Enterprise Logging and Auditing”.

Monitoring Using Splunk

Several third-party products that can integrate with the Layer 7 Gateway and can help in gathering data are available. The data may include monitor machine state and statistics, and presents the data about the Gateway in formats such as charts, graphs, and reports. One particular product is *Splunk*, which presents data, tracks appliance performance trends, and monitors the Gateway. Splunk integrates easily with the Layer 7 Gateway and allows for monitoring the Gateway's processes and statuses in a Web interface.

For more information on using Splunk with the Gateway, see Chapter 9, “Monitoring Using Splunk”.

Monitoring Using Nagios

Nagios is a tool for monitoring mission critical areas of network infrastructures such as the Layer 7 Gateway. Its flexible infrastructure allows different methods of monitoring statistics produced by the Gateway and the components it relies upon. By using Nagios, the Gateway can be easily tied into an organization-wide monitoring infrastructure. Nagios can be used to obtain a broad range of information from the Gateway, such as the amount of Free Memory at the Operating System level to the number of requests handled by a specific Gateway service.

For more information on using Nagios with the Gateway, see Chapter 10, “Nagios Integration”.

Conclusion

There are many different ways to monitor the Layer 7 Gateway to satisfy the needs of the various parties within an organization. Published services and associated policies as well as other relevant information can be gathered from the Gateway by using the command line and an SSH connection to the appliance. By using SNMP tooling, SNMP can be enabled on the Gateway to allow it to be monitored. The Integrated Lights Out Manager (iLOM) is useful for administrators who are monitoring the hardware.

Chapter Three: Basic Monitoring

Introduction

Built into every Layer 7 Gateway is the basic monitoring functionality. The Layer 7 Policy Manager also provides many tools in gathering basic information, such as details about a published service, the policy that is tied to the service, and basic performance statistics and auditing information.

Basic Monitoring Tools Included

The Layer 7 Policy Manager provides several tools for basic monitoring “outside of the box”. These are just some of the monitoring tools available:

- The services published on the Gateway
- The service OIDs that identify each individual service
- The service endpoints and endpoint conflict resolution
- Policy versions
- Service metrics
- Gateway and cluster status

Gathering Basic Information about a Gateway Service

Basic information about a published service in the Layer 7 Policy Manager is available simply by right-clicking the published service and choosing one of the following options from the context menu.

- **Active Policy Assertions:** Displays the active policy for this published service.
- **Service Properties:** Reveals underlying information related to the service. This is described in more detail below.
- **Revision History:** Shows the automatically versioned edited copies of the service policy.

Right-click any published service and go to **Service Properties** to view the dialog as shown in Figure 1.

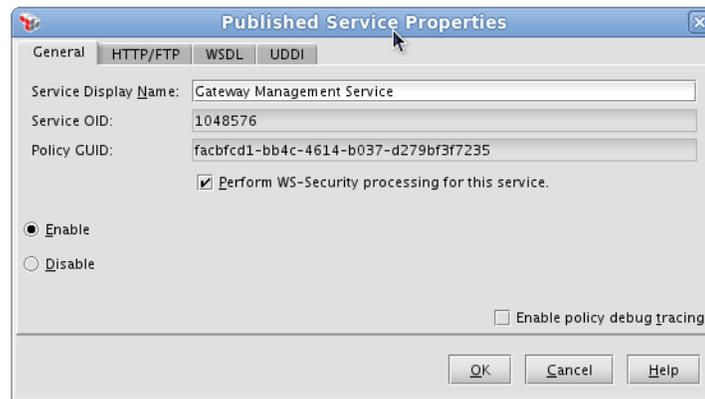


Figure 1: Published Service Properties dialog

The **[General]** tab shows basic information about a service, such as the Service OID number, whether the service is enabled or not, and if WS-Security processing is enabled for this service.

The **[HTTP/FTP]** tab shows the service endpoint, as well as HTTP request methods that are accepted for processing by this service.

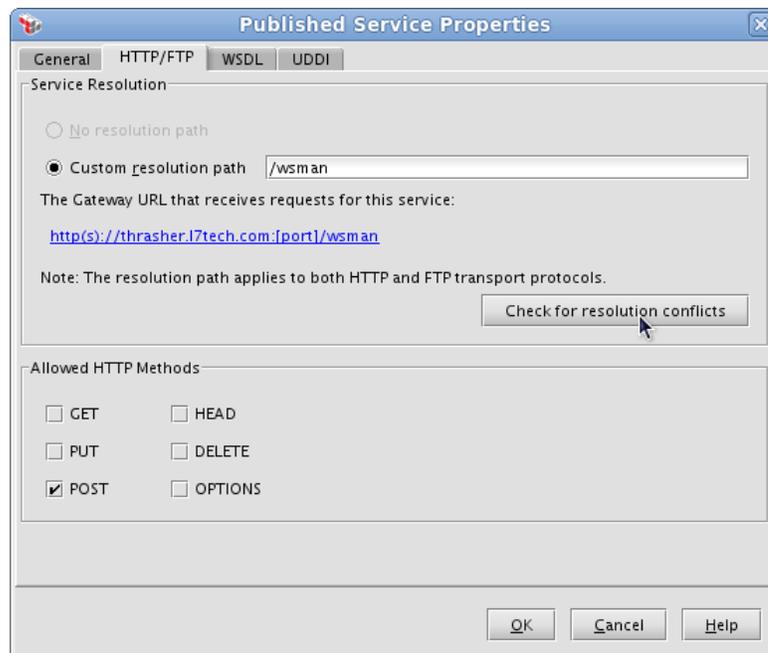


Figure 2: Published Service Properties – [HTTP/FTP] tab

The details presented in this dialog are helpful in basic information gathering and monitoring of services available on the Gateway.

If a particular service is experiencing problems, use this dialog first to see which URLs are resolvable in order to test the service.

The next step in troubleshooting is using the Active Policy Assertions option in the context menu. You can see the active policy that processes the requests sent to this service.

For more information about the Published Service Properties dialog, see “Service Properties” in the *Layer 7 Policy Manager User Manual*.

Service Metrics

The ability to view the service metrics from the Layer 7 Gateway is another basic monitoring feature. In the Policy Manager, select **View > Dashboard** (in the browser client: **Monitor > Dashboard**). This opens the Layer 7 Gateway Dashboard window.

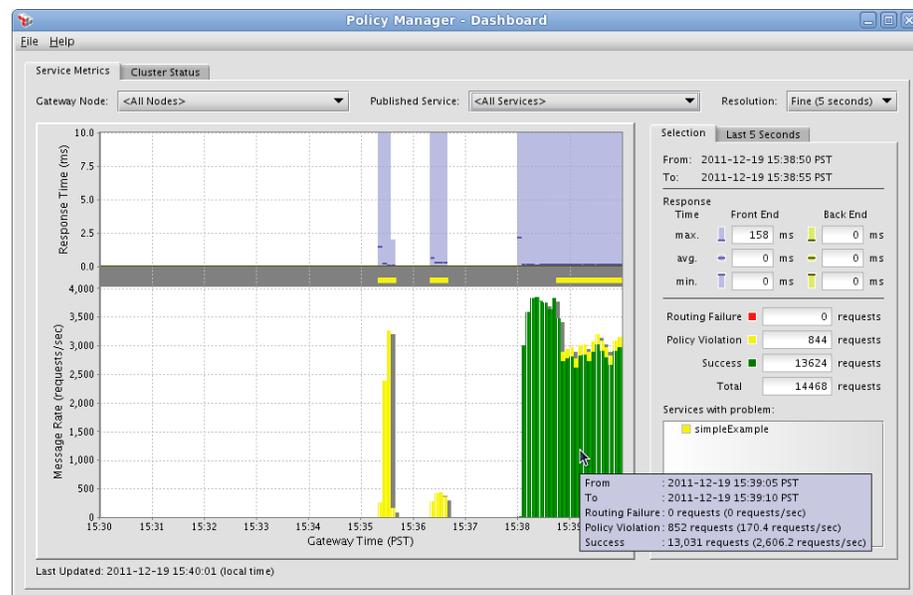


Figure 3: Policy Manager Dashboard

The Policy Manager Dashboard has many features that are helpful in monitoring services hosted on the Gateway. The rate at which messages are being processed is one particular area of interest when monitoring services. Figure 4 shows how this message rate appears in the Dashboard with a real-time graph. Viewing the general information about all services is the default behavior, but you can also drill down for data on a specific service or on a specific Gateway node.

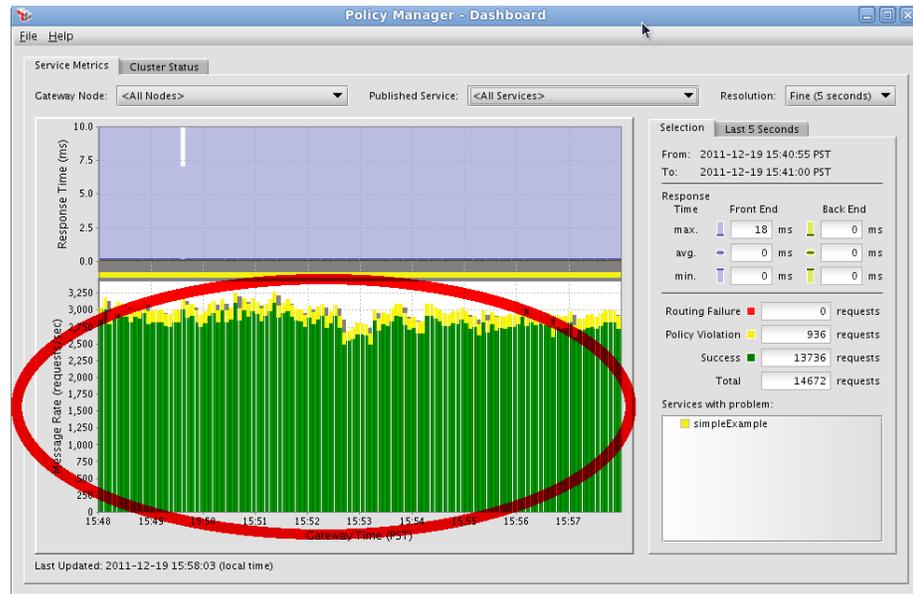


Figure 4: Dashboard - Message rate (requests/sec)

At the right of the Dashboard in Figure 5 are the service metric statistics. When you click on a time in the message rate graph, the numbers shown are for that particular moment in time, which correlates to the information listed in the [Selection] tab. These metrics are helpful to analyze requests that are being processed, and if there are any policies that are experiencing policy violations. Routing failures will appear in red in the message rate graph if they occur.

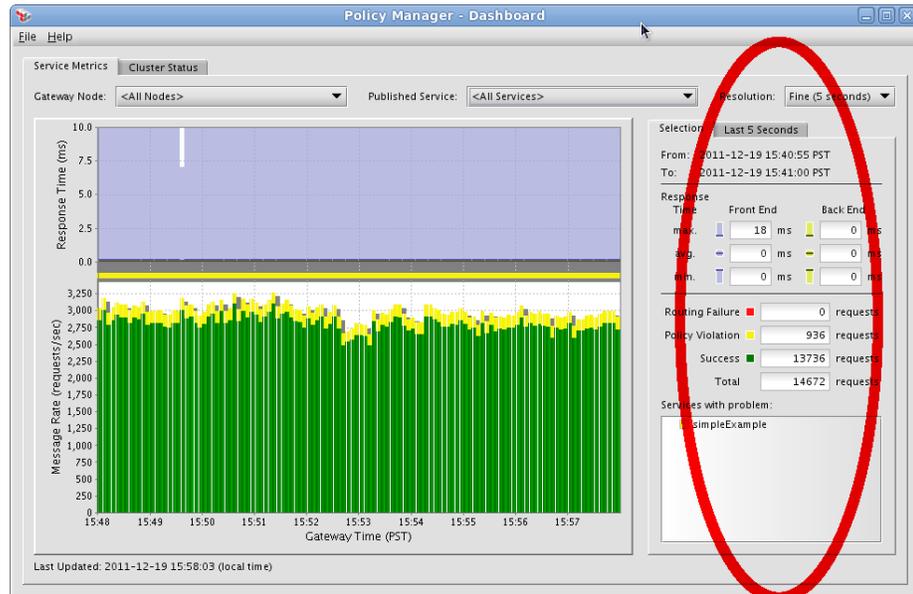


Figure 5: Dashboard - Selection metrics

The selection metrics are also helpful in identifying large front or back-end response times. A large back-end response time indicates that the Gateway is waiting on back-

end services to process incoming requests. A large front-end response time indicates that the Gateway itself is taking a long time to process requests, which in turn, could be caused by large back-end response times.

It is important to note that for performance testing and performance tuning, paying close attention to back-end response times can help to improve service performance. Gateways with large front-end response times may appear to be performing poorly, but the actual cause may be the delays in back-end services and systems that the Gateway relies on to process messages. Caching data (such as credentials and authentication or authorization) in a policy is preferable to improve performance and remove backend latency.

The **[Cluster Status]** tab on the Dashboard shows which cluster nodes are available and operational while monitoring a Gateway cluster. This tab provides a quick overview of the health of the cluster nodes and it is helpful for analyzing problems that may have cluster-related symptoms.

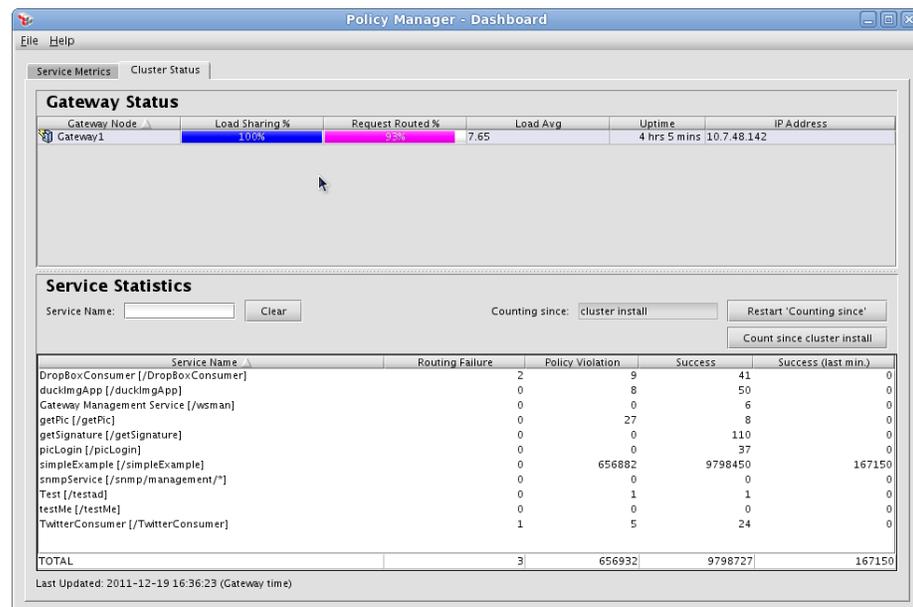


Figure 6: Dashboard - Cluster Status

The tab also shows service statistics that can help pinpoint services experiencing routing failures, policy violations, or, how many successful requests have passed through a published service.

For more information about the Dashboard, see Chapter 6, “Analyzing Gateway Performance” in the *Layer 7 Policy Manager User Manual*.

Gateway Audit Events

Audit events are important pieces of information while monitoring the Layer 7 Gateway. In the Policy Manager, select **View > Gateway Audit Events** (browser client:

Monitor > Gateway Audit Events). The “Gateway Audit Events” window opens with some default information and search options.

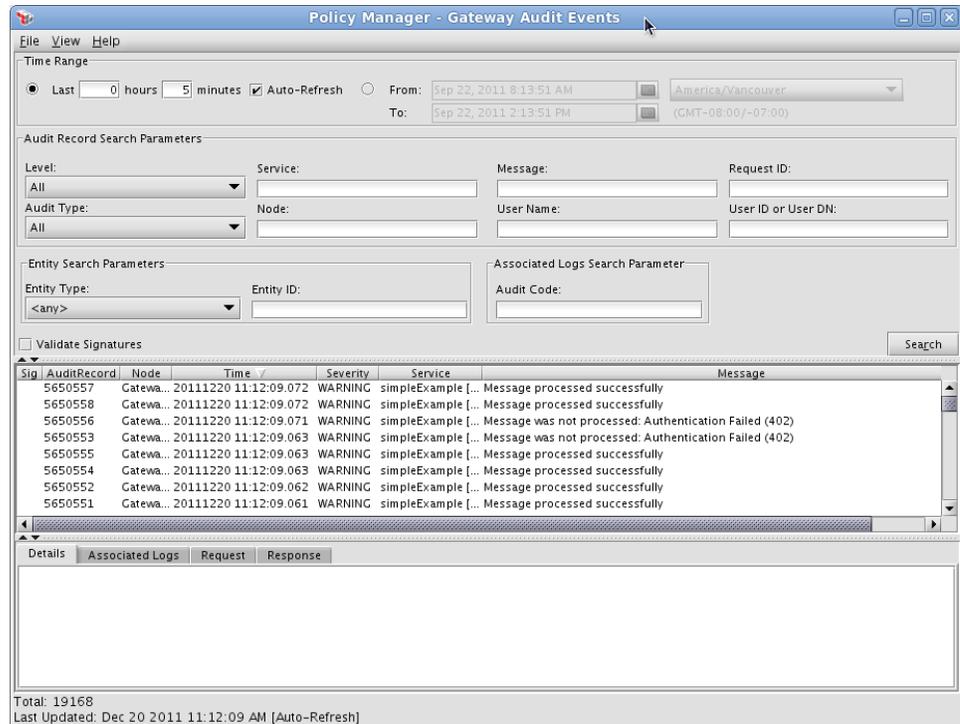


Figure 7: Gateway Audit Events

If auditing is enabled for the Gateway, use the Gateway Audit Events window to view audit records. Within the audit event viewer, you can select a particular audit record to view further details.

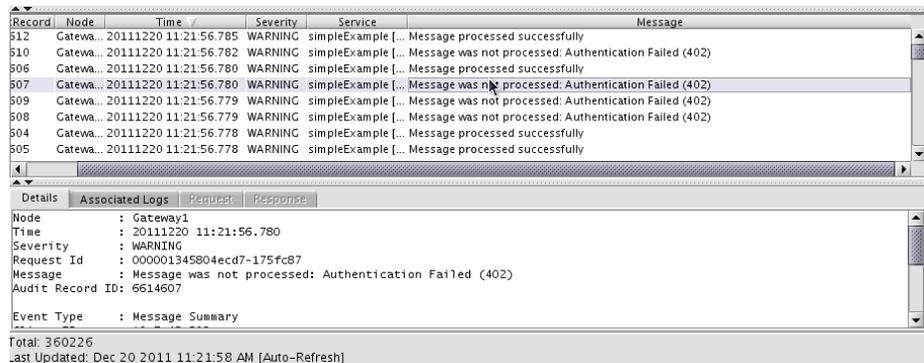


Figure 8: Audit record details

Messages that are being processed correctly can be viewed at a glance in the Audit records. To gather diagnostic information, you can view specific details in the Details and Associated Logs tabs.

For more information, see “Gateway Audit Events” in the *Layer 7 Policy Manager User Manual*.

Conclusion

Basic monitoring of the Gateway assists in confirming that the Gateway and its cluster nodes are operational and whether messages are being processed correctly by the published services and policies.

The Policy Manager is a key tool for viewing information about the services and cluster status of the Gateway. You can view which services are available and gather information about those services from the service properties.

The Dashboard within the Policy Manager provides visual graphing of message processing rates, as well as information about front and back-end latency for services published on the Gateway. You can check the status of services on the cluster by viewing the **[Cluster Status]** tab of the Dashboard.

If auditing is enabled, the Gateway Audit Event viewer can help diagnose message processing errors.

Chapter Four: Advanced Monitoring

This chapter describes how to monitor and administer the Layer 7 Gateway using a remote shell and command line tools.

Introduction

For those who have access to the command line on the Layer 7 Gateway, several command line tools are available to help you monitor and check the health status of the Gateway. By using a terminal, command line programs such as *vmstat* and *ps* are helpful for monitoring the Gateway. Layer 7 Technologies offers a Command Line Migration Toolkit specifically designed for monitoring and manipulating Gateway resources in a Dev/QA/Production environment.

Using the Linux Command Line

The following are monitoring procedures that can be performed through a Linux command line.

Accessing the Gateway via SSH

Some Gateway administrators are granted access to the command line via SSH when working remotely. With SSH access, you can log in to the Gateway and execute command line programs to view the current status of the Gateway. You can view which services are running, and install new Layer 7 modular and custom assertions when available.

```
[jsmith@smith ~]$ ssh ssgconfig@dev1.acmecorp.com  
ssgconfig@dev1.acmecorp.com's password:
```

Log in as the *ssgconfig* user to view the Gateway main menu.

```

Welcome to the Layer 7 Gateway

This user account allows you to configure the appliance
What would you like to do?

1) Configure system settings
2) Display Layer 7 Gateway configuration menu
3) Use a privileged shell (root)
4) Change the Master Passphrase
5) Display Remote Management configuration menu
6) Manage HSM
7) Display Enterprise Service Manager configuration menu
8) Display Patch Management menu
9) Display Log View menu
R) Reboot the SSG appliance (apply the new configuration)
X) Exit (no reboot)

make a selection: 1
  
```

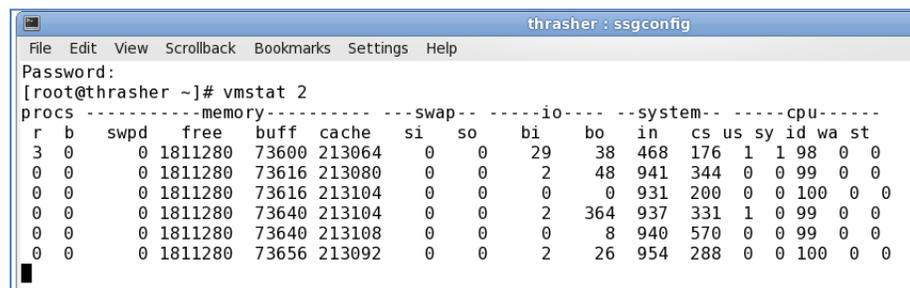
Figure 9: Layer 7 Gateway main menu

Option 3 “Use a privileged shell” is a particularly useful feature. This opens a command line with root privileges where you can enter Linux commands.

For a complete description of each option on the Gateway main menu, see “Accessing the Gateway Configuration Interface” in the *Layer 7 Installation Maintenance Manual*.

Viewing Current Load with “vmstat”

After logging into the Layer 7 Gateway with an SSH client, you can view the current load by executing the `vmstat` command. Executing `vmstat 2` will continue to show `vmstat` information every 2 seconds.



```

thrasher : ssgconfig
File Edit View Scrollback Bookmarks Settings Help
Password:
[root@thrasher ~]# vmstat 2
procs -----memory----- --swap-- --io-- --system-- --cpu-----
r b swpd free buff cache si so bi bo in cs us sy id wa st
3 0 0 1811280 73600 213064 0 0 29 38 468 176 1 1 98 0 0
0 0 0 1811280 73616 213080 0 0 2 48 941 344 0 0 99 0 0
0 0 0 1811280 73616 213104 0 0 0 0 931 200 0 0 100 0 0
0 0 0 1811280 73640 213104 0 0 2 364 937 331 1 0 99 0 0
0 0 0 1811280 73640 213108 0 0 0 8 940 570 0 0 99 0 0
0 0 0 1811280 73656 213092 0 0 2 26 954 288 0 0 100 0 0
  
```

Figure 10: `vmstat` running every 2 seconds

Running `vmstat` displays basic system information such as:

- amount of free memory
- load on the CPU
- which resources are occupying the CPU time

The example in Figure 10 shows that because the CPU is idle 98% to 100% of the time, the Gateway is also idle and therefore no messages are currently being

processed. If many messages are being processed, more CPU cycles will be reflected in the “us” (user time) column under the CPU indicator in *vmstat*.

```
procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r  b   swpd   free   buff  cache   si   so    bi   bo    in   cs  us  sy  id  wa  st
 8  0     0 1558632 75348 273192    0    0   526 2052 2918 28624 57 36  5  1  0
11  0     0 1543504 75388 282316    0    0   546 10826 2994 29512 53 36  7  5  0
10  0     0 1507172 75424 292800    0    0   586  7356 3104 30069 51 41  6  2  0
11  1     0 1497376 75468 297812    0    0   572 3892 3124 30998 56 34  7  2  0
 6  1     0 1468484 75496 299976    0    0   130 3390 2964 30035 54 38  6  2  0
```

Figure 11: *vmstat* showing a Gateway processing 650 requests/sec

In Figure 11 above, the “us” column rises to over 50, indicating that approximately 50% of the CPU is being used to run non-kernel code. This indicates the workload on the Gateway to process the incoming messages. This is useful information to you, as an administrator, to see that the Gateway is indeed processing messages and that the amount of CPU time is approximately 50%.

Using other monitoring tooling available, such as the Dashboard in the Gateway Policy Manager, you can correlate the CPU usage to the number of services you have published that are processing messages. This will help you in understanding how much CPU is being consumed for a certain amount of traffic. Logging this information over time helps provide data that can be analyzed in deciding when you need to scale your Gateway environment.

If you are administering a Gateway appliance, and receive a notification that the gateway is not functioning properly, the *vmstat* data on the command line can help you investigate the issue. The *vmstat* data will detect if the gateway is processing messages and how much memory is available, if the user CPU time is very large, and how much paging is occurring. This can all be helpful information when diagnosing problems or just when periodically checking gateway health.

Checking Java Processes with “ps”

The *ps* command can be used for checking the Java processes that are driving the Gateway message processing engine. This helps you ensure that the Java processes running correctly, which is a good indication that the Gateway is available and running properly.

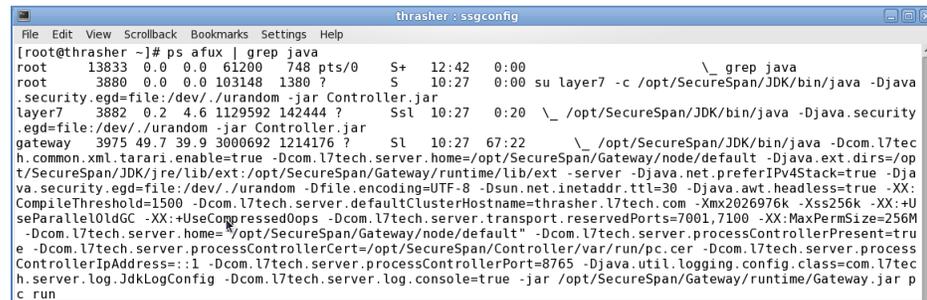
The *ps* command also integrates with the Gateway's SNMP functionality to check the Java processes. When using SNMP with *ps* to check Java, you can configure it to send alerts in the event the Java processes disappear. This would indicate a fatal problem with the Gateway that must be investigated immediately.

To check for the Java processes present on the Gateway, run the following command from the privileged shell:

```
ps afux | grep java
```

In the sample output in Figure 12, there are three Java processes running. The *-jar Controller.jar* Java process is the Gateway's process controller. This process is crucial

to the execution of the Layer 7 Gateway. It should always be listed as one of the running Java processes.



```

thrasher : ssgconfig
File Edit View Scrollback Bookmarks Settings Help
[root@thrasher ~]# ps -afux | grep java
root      13833  0.0  0.0 61200  748 pts/0    S+   12:42   0:00          \_ grep java
root      3880   0.0  0.0 103148 1380 ?        S    10:27   0:00 su layer7 -c /opt/SecureSpan/JDK/bin/java -Djava
.security.egd=file:/dev/./urandom -jar Controller.jar
layer7    3882   0.2  4.6 1129592 142444 ?        Ssl  10:27   0:20 \_ /opt/SecureSpan/JDK/bin/java -Djava.security
.egd=file:/dev/./urandom -jar Controller.jar
gateway   3975  49.7  39.9 3000692 1214176 ?        Sl   10:27   67:22 \_ /opt/SecureSpan/JDK/bin/java -Dcom.l7tec
h.common.xml.tarari.enable=true -Dcom.l7tech.server.home=/opt/SecureSpan/Gateway/node/default -Djava.ext.dirs=/op
t/SecureSpan/JDK/jre/lib/ext:/opt/SecureSpan/Gateway/runtime/lib/ext -server -Djava.net.preferIPv4Stack=true -Dja
va.security.egd=file:/dev/./urandom -Dfile.encoding=UTF-8 -Dsun.net.inetaddr.ttl=30 -Djava.awt.headless=true -XX:
CompileThreshold=1500 -Dcom.l7tech.server.defaultClusterHostname=thrasher.l7tech.com -Xmx2026976k -Xss256k -XX:+U
seParallelGC -XX:+UseCompressedOops -Dcom.l7tech.server.transport.reservedPorts=7001,7100 -XX:MaxPermSize=256M
-Dcom.l7tech.server.home=/opt/SecureSpan/Gateway/node/default" -Dcom.l7tech.server.processControllerPresent=tru
e -Dcom.l7tech.server.processControllerCert=/opt/SecureSpan/Controller/var/run/pc.cer -Dcom.l7tech.server.process
ControllerIpAddress=:1 -Dcom.l7tech.server.processControllerPort=8765 -Djava.util.logging.config.class=com.l7tec
h.server.log.JdkLogConfig -Dcom.l7tech.server.log.console=true -jar /opt/SecureSpan/Gateway/runtime/Gateway.jar p
c run
  
```

Figure 12: Running the "ps -afux | grep java" command

Command Line Policy Migration Toolkit

In addition to using Linux command line programs to monitor and administer your Gateway, there is also a Command Line Migration Toolkit available for the Layer 7 Gateway.

Note: contact Layer 7 Technologies to obtain the Command Line Policy Migration Toolkit.

Using the Command Line Migration Toolkit, you can gather information about policies and services that are published on a particular Gateway, or migrate services and policies between Gateways. For example, you can migrate services between development to pre-production instances of the Gateway, or from pre-production to a production Gateway.

Migrating services can be challenging when policies published on the Gateway rely on specific configuration items such as the following.

- LDAP repositories
- Database connections
- Federated identity providers
- JMS connections
- Policy fragments
- XML schemas
- Private keys
- Trusted certificates

Using the Command Line Migration Toolkit, you can easily migrate these items between environments without having to manually create or import each configuration item.

Understanding Policies vs. Services

When migrating services and policies between Gateways, it is important to note the difference between Gateway *services* and Gateway *service policies*.

- A Gateway *service* is an entity on the Gateway with specific configuration information. You can view this configuration information by right-clicking the service in the Policy Manager (lower left corner of the interface) and then selecting **Service Properties** (Figure 13).
- The service policy is separate from the service on the Gateway. It contains information about what policy rules will be enforced when the endpoint it being protected accepts an incoming request.

When using the Command Line Migration Toolkit to perform requests that gather information about (or those that create or modify) existing services, you are working with the data and information related to a published service. When using the toolkit to perform policy-related tasks, you are working with the policy XML definition itself.

The service information is the foundation of the Gateway; the policy information pertains to what is executed by the Gateway when a request is processed.

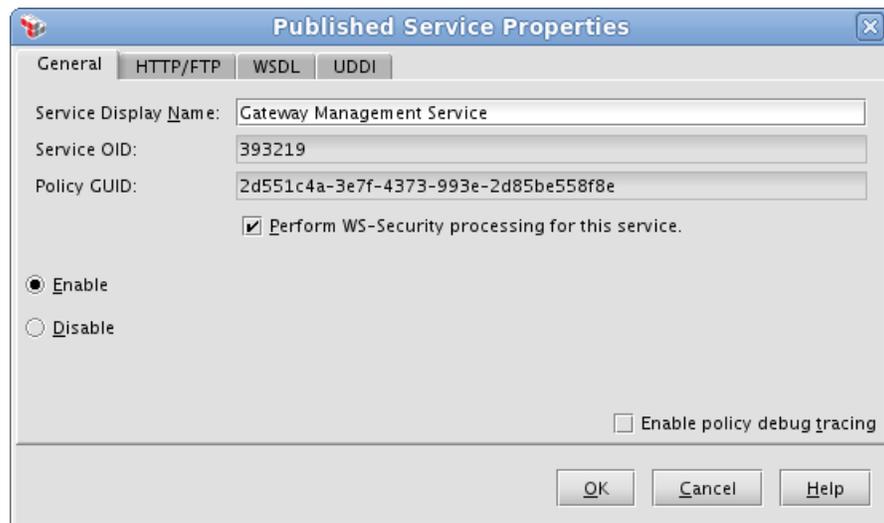


Figure 13: Published Service Properties

The Published Service Properties include the type of service, the HTTP methods that can be used to access the service, the service endpoint, and the policy that is associated with that service. If the service is a SOAP service, the WSDL for that published service is available as well. Note that the Published Service Properties only provides information about the Gateway service—it has no information related to the contents or structure of the policy associated with the service, aside from the Policy GUID being displayed.

For more information about the Published Service Properties, see “Service Properties” in the *Layer 7 Policy Manager User Manual*.

Using the Command Line Policy Migration Toolkit

Unzip the toolkit .zip file to a folder on your local drive. The toolkit is located in the folder named *GatewayCommandlineMigration-x.x*, where “x.x” is the version of the Gateway for which the toolkit is intended.

Ensure that the internal Gateway Management service is published before using the command line toolkit.

Tip: For more information about the policy migrator tools, including a more detailed description of the parameters, refer to *Using the Command Line Policy Migration Tools v1.1*. You may obtain this document by contacting Layer 7 Technical Support.

- *To publish the Gateway Management service:*

Start the Layer 7 Policy Manager and connect to your Gateway.

Select **Tasks > Publish Internal Service** (browser client: **Manage > Publish Internal Service**). The Publish Internal Service Wizard appears.

Select **Gateway Management Service** from the drop-down list and leave the routing URI at the default

Select **[Finish]** to publish the service.

For more information, refer to these topics in the *Layer 7 Policy Manager User Manual*:

Publish Internal Service Wizard
Working with Internal Services

Listing Services

You can use the *listServices* tool to list the services on a Gateway.

- *To list the available services on a Gateway:*

- Run the following command from the toolkit folder:

```
policyMigrator listServices <parameters>
```

Provide the following parameters (all in one line with spaces):

```
-h ${sourceGatewayIP/Hostname}  
-u ${sourceAdminUsername}  
-x ${sourceAdminPassword}  
--trustServerCertificate=yes  
--trustServerHostname=yes
```

The *listServices* command will display the services published on the Gateway, including the service ID, name, resolution URL, and whether the service is enabled:

```

Service ID: 393219
Name: Gateway Management Service
URL: /wsman
Enabled: true

Service ID: 393218
Name: testSvc2
URL: /testSvc2
Enabled: true

Service ID: 393217
Name: testSvc1
URL: /testSvc1
Enabled: true

Service ID: 393216
Name: testSvc
URL: /testSvc
Enabled: true

```

Figure 14: Displaying the published services on a Gateway

Listing References

For each service published on the Gateway, you can list the references belonging to the service using the *listReferences* tool. These references can include an LDAP connection or JDBC connection, for example.

➤ To list the references for a service:

- Run the following command from the toolkit folder:

```
policyMgrator listReferences <parameters>
```

Provide the following parameters all in a single line with spaces.

```

-h ${sourceGatewayIP/Hostname}
-u ${sourceAdminUsername}
-x ${sourceAdminPassword}
--trustServerCertificate=yes
--trustServerHostname=yes
-n ${serviceId}
-o PolicyExportFile.xml
-r ReferenceExportFile.xml

```

The *listReferences* tool will export the policy related to the service ID specified in the “-n” parameter. The policy will be exported into the “-o” file name and the references will be exported into the “-r” file name.

For example, if you have a service that contains a reference to an LDAP configuration, the LDAP configuration information will output into the references file (shown in the example below). The following is a sample references file for a Gateway that contains an LDAP configuration:

```

#cat testSvc_references.xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<references>
<ldapProvider name="surfscoter" oid="786432">
<baseDn>dc=l7tech,dc=com</baseDn>
<url>ldap://10.7.48.231:389</url>
</ldapProvider>
</references>

```

The reference file contains the LDAP configuration information and can be used to generate a map file. The map file is then used to map the LDAP on the “dev” Gateway to the LDAP on the “production” Gateway when migrating a service from the development environment to production.

Note: This functionality is normally provided on the GUI when using the Policy Manager. When importing a policy file that contains a reference to an LDAP connection, the *Resolve External Dependencies Wizard* will ask you to select the LDAP on the new Gateway to use for the policy.

Using the Command Line Migration Toolkit, you can create a map file to map the LDAP used in a development environment’s policy to a corresponding LDAP that exists in the production environment. You can import the policy to a new service in the production environment and have the LDAP reference resolve correctly.

Creating a Map File

A map file is used to map resources from one Gateway to resources on another Gateway. For example, your development Gateway has an LDAP Identity Provider configured and referenced in policy. You wish to migrate that policy to your production environment. You will need to provide a mapping file when using the command line policy migration tool, so that the policy will refer to the correct production LDAP configured on the production Gateway after you migrate the policy. This same mapping is required for other resources configured on your Gateways, including JMS and JDBC connections.

The following is an overview of creating the map file:

- Generate the reference file (“-r” parameter) using the *listReferences* command.

Pass the referenced .xml file as a parameter to the policyMigrator script *createMap* option.

When you migrate the policy to the production Gateway, use the map file created in step 2. This will ensure that your policy references the correct resources available on the production Gateway.

➤ *To create the map file, execute the following steps.*

- Run the following command from the toolkit folder:

```
policyMigrator createMap <parameters>
```

Provide the following parameters (all in one line with spaces):

```
-h ${destinationGatewayIP/Hostname}  
-u ${sourceAdminUsername}  
-x ${sourceAdminPassword}  
--trustServerCertificate=yes  
--trustServerHostname=yes  
-r testSvc_references.xml  
-o map.xml
```

The sample output file `map.xml` (created by the “-o” parameter) for a referenced `.xml` file (specified by the “-r” parameter) containing an LDAP reference might look like the following:

```
#cat map.xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<mappings>
<!-- Example mappings
  <mapping sourceOid="12345" targetOid="67890" type="ldapProvider"/>
  <mapping sourceOid="12345" targetOid="67890" type="ftpProvider"/>
  <mapping sourceOid="12345" targetOid="67890" type="jmsConnection"/>
  <mapping sourceOid="12345" targetOid="67890" type="privateKey"/>
  <mapping sourceOid="12345" targetOid="67890"
type="trustedCertificate"/>
  <mapping sourceOid="12345" synId="12345678-9abc-def0-1234-
56789abcdef0" targetOid="67890" type="jdbcConnection"/>
  <mapping sourceOid="12345678-9abc-def0-1234-56789abcdef0"
targetOid="0fedcba9-8765-4321-0fed-cba987654321"
type="policyInclude"/>
  <mapping sourceOid="12345" targetOid="67890" type="schema"/>
-->
<!-- Missing Mappings
  <mapping sourceOid="786432" targetOid="??" type="ldapProvider"/>
-->
</mappings>
```

Pay particular attention to this section in the file, if it is present:

```
<!-- Missing Mappings
  <mapping sourceOid="786432" targetOid="??" type="ldapProvider"/>
-->
```

In this example, the mapping `sourceOid` of the LDAP on the development Gateway is 786432, but the mapping tool was unable to map the target Oid of the production LDAP Oid (as indicated by “??”). In this case, you must manually enter the Oid of the production LDAP you want to reference when migrating the policy to the production Gateway.

➤ *To locate the Oid of the LDAP configuration on the production Gateway:*

- Start the Policy Manager and connect to the production Gateway.

Open an existing policy that references the LDAP configuration you want to map to.

Right-click on the *Authenticate Against Identity Provider* assertion in the policy window and then select **Copy**. This copies the assertion’s XML code into the clipboard.

Paste the contents of the clipboard into a text editor.

View the XML code and locate the following section:

```
<?xml version="1.0" encoding="UTF-8"?>
<wsp:Policy xmlns:L7p="http://www.layer7tech.com/ws/policy"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2002/12/policy">
  <wsp:All wsp:Usage="Required">
    <L7p:Authentication>
      <L7p:IdentityProviderOid longValue="786432"/>
    </L7p:Authentication>
  </wsp:All>
</wsp:Policy>
```

Make a note of the Oid of the LDAP configuration as shown highlighted above, next to “longValue”.

Open the *map.xml* file from above in a text editor.

Replace the “??” with the Oid noted in step 6.

Delete the comment “Missing Mappings”:

```
<!-- Missing Mappings
-->
```

Save and exit *map.xml*.

The *map.xml* mapping file can then be used to migrate your policy to the production Gateway using the *testMap* and *importPolicy* or *importAsNewService* toolkit options.

Testing the Map File

After you have generated the map file and manually edited it as required, you can use this file to check the mappings against your target Gateway. To test the map file against the target Gateway, the *testMap* tool is used.

For information on generating and editing the map file, see “*Creating a Map File*” on page 22.

➤ *To test the map file:*

- Run the following command from the toolkit folder:

```
policyMigrator testMap <parameters>
```

Provide the following parameters (all in one line with spaces):

```
-h ${targetGatewayIP/Hostname}  
-u ${adminUsername}  
-x ${adminPassword}  
--trustServerCertificate=yes  
--trustServerHostname=yes  
-i ReferencesToCheck.xml  
-m map.xml
```

This command checks the map file for accuracy and validity.

Note: The *testMap* tool generates an output message only if the map file is not successfully tested.

Once you have generated your referenced resources from your policy, created and modified your map file, tested the map file successfully, you are now ready to migrate your policy to the target server.

Reusing Map Files

Once you establish working map files that help resources from one environment to another, you can reuse these map files.

In some instances, the resources that your policies and Gateways depend on do not change often, which leaves the map files unchanged as well. For example, when development and test environment LDAP identity providers are established in their internal test environments, they may never change at all. Therefore in this instance, you only need to create map files that map development resources to test resources once, and reuse them many times while migrating services and policies from Gateways in one environment to the other. These map files can then be stored in a source code repository.

Creating a New Service and Importing the Policy into the Target Server

You must create a target Gateway service before you can use the Command Line Migration Toolkit to import a policy to the Gateway. To do this, export the service from the original Gateway first, and then use the exported service data to create a new service on the target Gateway.

Tip: Remember that the Gateway service data is separate from the policy that is associated with the exported data service. An export data service *only* contains *service* information, not the *policy* information.

- To export a service from a Gateway:
 - Run the following command from the toolkit folder:

```
policyMigrator exportService <parameters>
```

Provide the following parameters (all in one line with spaces):

```
-h ${sourceGatewayIP/Hostname}  

-u ${adminUsername}  

-x ${adminPassword}  

--trustServerCertificate=yes  

--trustServerHostname=yes  

-s ${serviceOidToExport}  

-o FiletoSaveServiceMetadata.xml
```

This parameter will export the service from the source Gateway (identified by the service Oid in the “-s” parameter) and put the service definition in the file specified by the “-o” parameter.

After exporting the service definition from the source Gateway, you can create the new service on the target Gateway by running the *importAsNewService* tool.

Note: Ensure that the internal Gateway Management Service is enabled on the target Gateway. Otherwise, the import process will fail.

➤ To create a new service on the target Gateway:

- Run the following command from the toolkit folder:

```
policyMigrator importAsNewService <parameters>
```

Provide the following parameters (all in one line with spaces):

```
-h ${targetGatewayIP}
-u ${adminUsername}
-x ${adminPassword}
--trustServerCertificate=yes
--trustServerHostname=yes
-e svc_export.xml
```

The service that is created on the target Gateway will have the same service name and resolution URI as the exported service. The created service is disabled by default after running the *importAsNewService* tool.

Once you have created a copy of the service on the target Gateway, you can use any reference and policy XML files that may have created previously to import the policy for that service. The *map.xml* (see “Testing the Map File” on page 24) and *svc_reference.xml* files are used to import the service into the newly created service on the target Gateway.

Before importing the policy to the new service, determine the service OID on the target Gateway first.

➤ To determine the service OID on the target Gateway:

- Run the following command from the toolkit folder:

```
policyMigrator listServices <parameters>
```

Provide the following parameters (all one line; space separated):

```
-h ${targetGatewayIP}
-u ${adminUsername}
-x ${adminPassword}
--trustServerCertificate=yes
--trustServerHostname=yes
```

You will see an output similar to the following:

```
Sample output:
Service ID: 393217
Name: Gateway Management Service
URL: /wsman
Enabled: true

Service ID: 393216
Name: test
URL: /test
Enabled: true

Service ID: 393218
Name: testSvc
URL: /testSvc
Enabled: false
```

Figure 15: Sample output

Make a note of the Service ID for the imported service and continue to import the policy to that service. In the example above, it is **testSvc**.

➤ *To import policy to the service:*

- Run the following command from the toolkit folder:

```
policyMigrator importPolicy <parameters>
```

Provide the following parameters (all in one line with spaces):

```
-h ${targetGatewayIP}  
-u ${adminUsername}  
-x ${adminPassword}  
--trustServerCertificate=yes  
--trustServerHostname=yes  
-i testSvc_export.xml  
-m map.xml  
-n ${targetServiceOid}  
-o testSvc_updated.xml
```

The *svc_export.xml* file (“-i” parameter) is the policy exported by the *listReferences* tool, and the *map.xml* file (“-m” parameter) is the modified and tested map file. The *targetServiceOid* (“-n” parameter) is the target service OID for the service just created on the target Gateway. The file name specify by the “-o” parameter will contain the updated policy for the target Gateway once it is imported. The updated policy refers to the references that are resolved in the mapping XML file during import.

When the *importPolicy* tool is run successfully, you should see the following message:

```
Policy was successfully imported.
```

You can now enable the service on the target Gateway and begin to test the migrated service.

Enabling and Disabling a Service

At this point, you should have accomplished the following:

- Exported the policy and referenced sources from the original Gateway
- Created the map file to map the resources to their equivalents on the target Gateway
- Exported the original Service definition
- Created the new service on the target Gateway
- Imported the policy for the new service on the target Gateway

The final steps are to enable the service, and test the migrated service and policy.

Before you can enable the service, you need the service OID. To do this, use the *listServices* tool (see “*Listing Services*” on page 20 for details). Once you have obtained the service OID, you can enable the service.

➤ To enable the service:

- Run the following command from the toolkit folder:

```
policyMigrator enableService <parameters>
```

Provide the following parameters (all one line; space separated):

```
-h ${gatewayIP/Hostname}  
-u ${adminUsername}  
-x ${adminPassword}  
--trustServerCertificate=yes  
--trustServerHostname=yes  
-s ${serviceId}  
--enable=yes
```

Tip: You can disable the service by setting `--enable=no` in the parameters.

Modifying Service Properties

In addition to migrating policies between Gateway and creating services, you can use the command line policy migration toolkit to modify service definition information. For example, you can modify the service name and the resolution URI.

The following is an overview of the procedure to change the service definition using the command line policy migration toolkit.

- Export the existing service definition. For more information, see “To export a service from a Gateway” on page 25.
- Manually edit the definition file created by the export service tool (specified by the “-o” parameter) with your changes.
- Update the published services definition. This is described below.

➤ To update the published service definition:

- Run the following command from the toolkit folder:

```
policyMigrator updateService <parameters>
```

Provide the following parameters (all one line; space separated):

```
-h ${gatewayIP}  
-u ${adminUsername}  
-x ${adminPassword}  
--trustServerCertificate=yes  
--trustServerHostname=yes  
-s updatedServiceDefinition_export.xml  
-n ${serviceId}
```

The service definition is now modified on the published service with the corresponding service ID.

Conclusion

When you are monitoring and administering the Layer 7 Gateway, it is often useful to using the command line to perform commands and monitoring actions. With command line access to the Gateway, you can gather information about services that are published, view the processes that are currently running and available, view the load on the CPU, and access other helpful information related to Gateway health.

You can also use the command line on client computers to migrate existing services between Gateways. This is helpful when moving services and policies from internal development environments to your test and production environments. The toolkit also lets you see what services have been published and the status (whether enabled or not), as well as exporting and adjusting related service and policy definitions.

Chapter Five: General SNMP Monitoring

This chapter describes general SNMP (Simple Network Management Protocol) monitoring with the Layer 7 Gateway.

Introduction

As a useful tool for monitoring the Layer 7 Gateway, SNMP can provide information about a server or service to generate reports. It can send out automatic alerts to appropriate contacts, and it provides easy-to-understand “green light, red light” status of system availability.

The Layer 7 Gateway provides data via the SNMP that can be used in SNMP tool suites. Examples of data are whether the Gateway is available and running well, and if the services are correctly processing requests at a rate consistent with service level agreements.

About SNMP MIBs

The Layer 7 Gateway ships with several MIBs.

- Standard system MIB for Red Hat Enterprise Linux 5
- UCD Extended MIB (net-snmp)
- Layer 7 MIB

For more information on configuring the SNMP agent on the Layer 7 Gateway, contact Layer 7 Technical Support to obtain the *Configuring the Net-SNMP Agent* document.

Standard System MIB

Using the Standard System MIB, you can monitor information about the Gateway including the System Contact, the System Location, and the Uptime.

For more information, see the *Configuring the Net-SNMP Agent* document.

UCD Extended MIB

When you are using SNMP to monitor the Layer 7 Gateway health and availability, the UCD Extended MIB can be used to check running processes, disk space, and load average.

When you are checking for running processes, verify the following.

- **Java is running:** The Layer 7 Gateway processes are Java based. If Java is running, the Gateway should be running. If the Java process is not found when querying the gateway with SNMP, this indicates that the Gateway is not processing messages as expected.

Minimum 10GB of free hard disk space: If there is less than 10 gigabytes of available space, an alert should be flagged in the SNMP tooling to alert an administrator to evaluate the contents of the disk and attempt to free up more hard disk space.

Load average for the 1 minute, 5 minute, and 15 minute averages are not exceeded: If any of the load averages are exceeded, evaluate the Gateway environment and re-examine the scaling of the Gateway support.

Layer 7 MIB

The Layer 7 MIB provides SNMP tooling with access to service metric data for the Layer 7 Gateway. The Layer 7 Tactical SNMP implementation should be used to gather information about the Gateway's services via SNMP. By using the tactical SNMP implementation, you can gather information such as the number of requests processed for a particular service, failed routes for a service in the past hour, the average back-end response time for a particular service, or the number of policy violations a service has had in the last 24 hours.

For a complete list of information available in the Layer 7 MIB, see Chapter 6.

SNMP Events to Monitor

Each organization will have different requirements on what events to monitor for the Layer 7 Gateway. It is important to know your internal service level agreements and set up monitoring for Layer 7 Gateway status to meet those requirements.

In general, the monitoring requirements for a Gateway can be divided into two categories: **appliance monitoring** (hardware events) and **service monitoring**. There are two minimum requirements for the Gateway to be running properly. These are the running **Java processes** and **mysqld daemon**. If the Java processes are running, then the Gateway should process messages. If the *mysqld* process is running, then the database should be available to the Gateway. Monitoring these events will ensure that the basics of Gateway available have been met.

You can also monitor hardware events, which includes monitoring of fan failures, CPU temperatures, and system load to ensure that the Gateways are not being overwhelmed at the hardware level. If the load average on the Gateway is very large, the operations staff can begin expanding the infrastructure to allow the load to be dispersed across more instances of the Layer 7 Gateway. For more information, refer to the *Configuring the Net-SNMP Agent* document available from Layer 7.

Service monitoring can be implemented with SNMP. Monitoring of services can be configured to also help maintain service level availability (SLA) set out in your service level agreement policies. For information on monitoring the Gateway's services using SNMP see Chapter 6.

In general, it is recommended that you monitor particular services to ensure that they are not triggering a number of policy violations that exceeds the limits available in your SLA. Each service can be monitored and have alerts configured if the number of policy violations exceeds a limit.

Additionally, backend response times can help pinpoint problems with service infrastructure. If services are using backend components to perform authentication and/or authorization like Tivoli Access Manager or Oracle Access Manager, large backend response times could help identify bottlenecks that may cause these backend authentication systems to perform poorly. Large backend response times generally account for the majority of the total request processing time for the Layer 7 Gateway. If services with large backend response times are identified by monitoring the services with SNMP, you may implement measures such as caching to the Gateway policies to help reduce backend response time processing for the Gateway. This in turn could help reduce overall processing time and improve performance of the systems being guarded by the Gateway.

By comparing the available service monitoring metrics available, you can configure your SNMP tooling to help keep the Layer 7 Gateway performing within the thresholds outlined in your SLA.

Conclusion

SNMP is an excellent standardized method of monitoring the Layer 7 Gateway. The SNMP functionality is not enabled on the Gateway by default, but it is straightforward to set up. Using SNMP to monitor the Gateway, you have the flexibility of using any SNMP tooling within your organization and can monitor standard system properties using the UCD MIB.

Chapter Six: Monitoring Using Tactical Agent

This chapter describes how to monitor the Layer 7 Gateway services using the Tactical implementation of the SNMP Agent.

Introduction

The SNMP functionality built into the Layer 7 Gateway provides excellent monitoring using the *net-snmp* tooling to monitor the hardware. However, it is also possible to monitor the Gateway using your own SNMP infrastructure using the implementation developed by the Tactical Department at Layer 7 Technologies.

Technical Note: There is currently a known issue when using the SNMP tooling to monitor the Layer 7 Gateway services. When the Gateway restarts, the SNMP OIDs that identify a particular service changes, forcing the need to update the SNMP tooling configuration. contact the Tactical Department at Layer 7 for a workaround solution to this issue.

The Basic Solution

The Tactical implementation changes the way the SNMP OIDs are identified. The SNMP OID is now tightly coupled to Layer 7 Gateway's published service ID. A service on the Gateway has an ID such as "655360". For that service, the SNMP monitoring OID will now be: "1.3.6.1.4.1.17304.7.1.1.655360".

Using the Tactical SNMP Agent, the SNMP ID for a particular service statistic includes the Gateway service ID. With the Tactical implementation in effect, the SNMP OID for each service will not change when the Gateway is restarted.

The Extended Solution

In addition to resolving the issue of changing SNMP OIDs, the Layer 7 Tactical implementation of the SNMP agent exposes more metric data per service.

The original Layer 7 Gateway MIB file exposed a handful of statistics for the Gateway services.

- Service OID
- Service Name
- Number of requests the service has received
- Number of requests that have been authorized for that service
- Number of requests that have been completed for that service

The new Tactical implementation of the SNMP agent exposes more statistics about each service, including metric statistics that are available only through the Policy Manager Dashboard.

The following table lists the new metric data available via SNMP.

Table 1: New metric data available per Layer 7 Gateway published services

| Metric | Type |
|--|----------------|
| service OID | INTEGER, |
| serviceName | DisplayString, |
| requests | Counter32, |
| authRequests | Counter32, |
| completedRequests | Counter32, |
| failedRoutesLast24Hrs | INTEGER, |
| failedRoutesLastHour | INTEGER, |
| failedRoutesFineBin | INTEGER, |
| averageBackEndResponseTime24hours | INTEGER, |
| averageBackEndResponseTimeLastHour | INTEGER, |
| averageBackEndResponseTimeFine | INTEGER, |
| averageFrontEndResponseTimeLast24hours | INTEGER, |
| averageFrontEndResponseTimeLastHour | INTEGER, |
| averageFrontEndResponseTimeFine | INTEGER, |
| policyViolationsLast24Hours | INTEGER, |
| policyViolationsLastHour | INTEGER, |
| policyViolationsFine | INTEGER |

By using the new metric information available to SNMP monitoring tools, members of operations team involved in monitoring the Layer 7 Gateway can set up alerts based on service processing. For example, a particular service is being monitored via SNMP and is expected to be processing messages frequently. You can establish a threshold on the policy violations and failed routes statistics so that administrators can be notified if a service is failing to route incoming requests or if the policy is incurring violations when it should not be.

Also, data such as the average front and back end response times can be monitored and stored in order to generate reports and charts to analyze service trends.

Deploying the Tactical SNMP Agent

The Tactical SNMP agent solution involves the following:

- The Layer 7 Gateway SNMP daemon
- The Tactical SNMP Agent Assertion
- A modified shell script to handle data for the SNMP daemon
- A Layer 7 Gateway REST service to act as an SNMP agent

The Tactical SNMP implementation files package includes these files:

```
SnmpAgentAssertion.aar  
passTacticalServiceUsage.sh  
snmpService.xml  
l7_tactical_mib.mib
```

Prerequisites:

- a configured and operational SNMP daemon
- configured test clients that can retrieve data from the Gateway using SNMP commands

Deploying the Agent

➤ *To deploy the Tactical SNMP Agent:*

- Open a privileged shell on the Gateway. For more information, see option **3** in Figure 9 on page 16.

Stop the Gateway service with this command:

```
service ssg stop
```

Copy the Tactical SNMP Agent assertion file (*SnmpAgentAssertion.aar*) to this directory:

```
/opt/SecureSpan/Gateway/runtime/modules/assertions
```

Modify the ownership of the assertion file by executing this command (all on one line):

```
# chown layer7:layer7  
/opt/SecureSpan/Gateway/runtime/modules/assertions/SnmpAgentAssertion.aar
```

Modify the permissions for the assertion .aar file by running these commands (each command is a single line):

```
# chmod a+r  
/opt/SecureSpan/Gateway/runtime/modules/assertions/SnmpAgentAssertion.aar
```

```
# chmod a-wx  
/opt/SecureSpan/Gateway/runtime/modules/assertions/SnmpAgentAssertion.aar
```

Open the following file in a text edit (must be logged in as the root user):

```
/etc/snmp/snmpd.conf
```

Locate the line containing the “pass” command:

```
pass .1.3.6.1.4.1.17304.7 /bin/sh  
/opt/SecureSpan/Appliance/bin/passServiceUsage.sh
```

Comment out that command by typing a “#” character at the front of the line.

Copy that line and paste it underneath the original line.

Remove the “#” characters, and change the line to this:

```
pass .1.3.6.1.4.1.17304.7 /bin/sh  
/opt/SecureSpan/Appliance/bin/passTacticalServiceUsage.sh
```

Save the file and exit the editor.

Copy the *passTacticalServiceUsage.sh* file to this directory:

```
/opt/SecureSpan/Appliance/bin
```

Change the *passTacticalServiceUsage.sh* file to be executable by all users with this command:

```
chmod a+x /opt/SecureSpan/Appliance/bin/passTacticalServiceUsage.sh
```

Restart the SNMP daemon with this command:

```
service snmpd restart
```

Restart the Gateway with this command:

```
service ssg restart
```

Adding the Service to the Gateway

The Tactical SNMP agent solution uses a Layer 7 Gateway service to act as the SNMP agent for the SNMP daemon. The next step is to log into the Policy Manager and publish a new service to act as the SNMP Agent.

➤ *To add the service to the Gateway:*

- Open the Layer 7 Policy Manager and connect to the Gateway.

On the Home page, click **Publish REST, Web API, or Other Service**. The Publish REST, Web API, or Other Service Wizard is displayed.

Complete the wizard as follows:

- *Service Name:* **snmpService**

- *Target URL*: leave this empty
- Gateway URL: **snmp/management/***

Click [**Finish**].

Click **Import Policy** from the policy toolbar (above the policy editor).

Import the *snmpService.xml* policy file included in the Tactical SNMP package.

Click **Save and Activate** in the policy toolbar.

You have now deployed the assertion for the SNMP agent, modified the SNMP daemon settings in *snmpd.conf*, copied the *passTacticalServiceUsage.sh* file into the correct directory, and deployed the new SNMP Gateway policy that will act as the SNMP agent.

You can now test the tactical SNMP agent by issuing a call to the SNMP daemon using a command such as:

```
# snmpwalk -m /home/myhome/17_tactical_mib.mib -v 2c rat.17tech.com -c  
17 1.3.6.1.4.1.17304.7.1
```

Where “*/home/myhome/17_tactical_mib.mib*” is the full path to the Layer 7 Tactical SNMP MIB file.

Conclusion

SNMP is a powerful service that can help you monitor the health of your Layer 7 Gateway. Using the Layer 7 Tactical SNMP agent implementation ensures that the SNMP OIDs for services remain consistent and more metric data are available. By using SNMP, the retrieved data can be used to trend statistics and generate reports that can help forecast future needs of the team responsible for monitoring the health of the Gateway and its services.

Chapter Seven: Monitoring the Gateway Using WSDM

Introduction

This chapter describes how to use the WSDM standardized Subscription Manager tool to send service statistics from the Layer 7 Gateway to a reporting tool in order to generate graphs and charts for business analysis.

What is “WSDM”?

The OASIS Web Services Distributed Management (WSDM, pronounced *wisdom*) is a Web service standard for managing and monitoring the status of other services. The standard defines two sets of specifications: *Management Using Web Services* (MUWS) and *Management of Web Services* (MOWS). To learn more about these specifications, refer to the OASIS Website:

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsdm

Setting Up the Layer 7 Gateway

The Layer 7 Gateway has existing functionality to monitor services and push out service metrics at a specified interval to a third-party reporting tool. One such tool is Splunk, which allows you to use HTTP transport to push this data.

Step 1: Enable the Gateway WSDM Subscription Tool

First, set up the Layer 7 Gateway WSDM Subscription Manager and WSDM Metrics Notifier in the Policy Manager.

➤ *To enable the Gateway WSDM subscription tool:*

- Start the Layer 7 Policy Manager and connect to the Gateway.

Set up the *Gateway Management Service* by doing the following:

- Select **Tasks > Publish Internal Service** (browser client: **Manage > Publish Internal Service**). The Publish Internal Service Wizard appears.
- Choose **Gateway Management Service** from the dropdown list. Leave the Routing URI at the default.

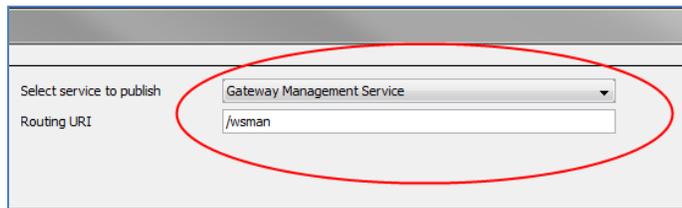


Figure 15: Publishing the Gateway Management Service

- Click [**Finish**]. The Gateway Management Service (/wsman) appears in the Services and Policies list at lower left of the Policy Manager.

Set up the WSDM Subscription Manager by doing the following:

- Select **Tasks > Publish Internal Service** (browser client: **Manage > Publish Internal Service**). The Publish Internal Service Wizard appears.
- Choose **WSDM Subscription Service** from the dropdown list. Leave the Routing URI at the default.

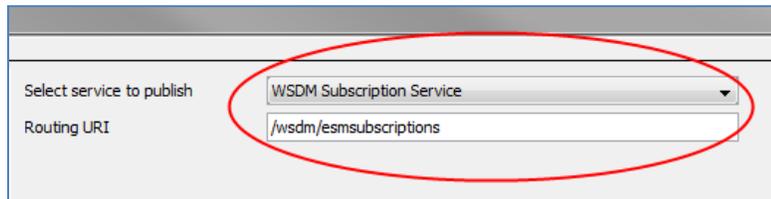


Figure 16: Publishing the WSDM Subscription Service

- Click [**Finish**]. The WSDM Subscription Service (/wsdm/esmsubscriptions) appears in the Services and Policies list at lower left of the Policy Manager.

Set up the WSDM Notification Service by doing the following:

- Select **Tasks > Create Policy** (browser client: **Manage > Create Policy**). The Policy Properties dialog appears.
- Enter a name for this policy (for example “notifyPolicy”).
- From the **Policy Type** drop-down list, choose **Internal Use Policy**.
- From the Policy Tag drop-down list, choose **wsdm-notification** and then click [**OK**].

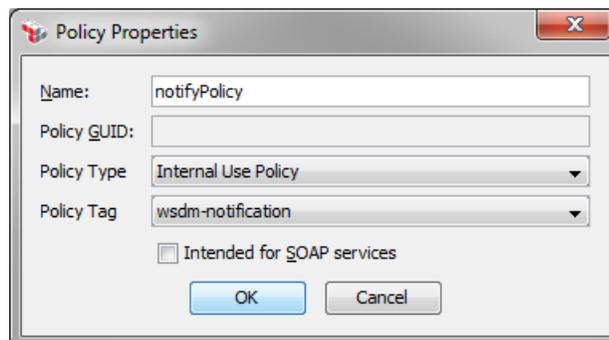


Figure 17: Creating a policy for WSDM Notification

The new policy will appear in the Services and Policies list, with a yellow icon () indicating that it is an internal service.

Step 2: Link Notification Service to Subscription Manager

Next, you will link the WSDM Notification Service to the WSDM Subscription Manager.

➤ *To link the notification service to the subscription manager:*

- In the Policy Manager, double-click the **WSDM Subscription Service** in the Services and Policies list. This loads the policy for the WSDM Subscription Service.

Add the *Subscribe to WSDM Resource* assertion to the Policy Development window. It can be added at the bottom of the policy. This assertion is located under the **Internal Assertions** category in the assertions palette. The WSDM Subscription Properties is displayed.

Select your new internal policy (e.g., “notifyPolicy”) from the properties and then click [OK].

The Layer 7 Gateway WSDM Subscription Manager and the WSDM Metrics Notification Manager are now set up. You can now begin to subscribe services.

Step 3: Configure Notification Settings

There are two cluster properties for controlling WSDM notifications:

Table 2: WSDM notification settings

| Cluster property | Value |
|-----------------------------------|--|
| wsdm.notification.enabled | Enable notifications when subscribing to a WSDM resource. Value is a Boolean. Default: true |
| wsdm.notification.interval | The interval between WSDM subscription notifications attempts. Default: 60000 (milliseconds) |

These properties are global values and apply to all services subscribed.

➤ *To modify the WSDM notification settings:*

- In the Policy Manager, select **Tasks > Manage Cluster-Wide Properties** (browser client: **Manage > Manage Cluster-Wide Properties**). The Manage Cluster-Wide Properties dialog appears.

Click [Add] and then locate the WSDM cluster property to change. Tip: You should normally keep notifications enabled; only change the interval.

Type a new **Value** and then click [OK].

Step 4: Subscribe Services

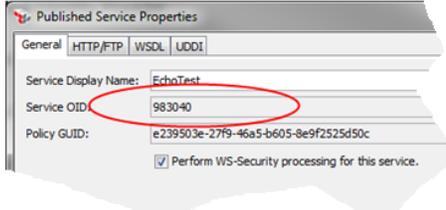
You can now subscribe services to the WSDM Subscription Manager to monitor and gather data.

Tip: To automate this process, contact Layer 7 Technical Support to obtain the *Layer 7 Gateway WSDM Auto-Subscription Command Line Tool*. This tool will monitor services and automatically subscribe/unsubscribe services as they are created or deleted from the Gateway.

To manually work with the Layer 7 Gateway WSDM Subscription Manager, see Table 3 below.

Note: The URL to target for subscribing, unsubscribing, and renewing a service is in the format:
 http://<gateway>:8080/wsdm/esmsubscriptions?serviceid=<serviceoid>

Table 3: Explanation of elements for working with WSDM Subscription Manager

| Element | Description |
|----------------|--|
| <gateway> | Address of the Layer 7 Gateway. |
| <serviceoid> | <p>The OID of the service being subscribed to the Gateway WSDM Subscription Manager for monitoring and gathering statistics.</p> <p>To find the OID, right-click the service in the Services and Policies list in the Policy Manager, and then select Properties. The service OID is displayed in the [General] tab of the Published Service Properties.</p>  |
| <subscription> | <p>The subscription ID of the service being unsubscribed. This can be found in the Gateway's <code>wsdm_subscription</code> table within the Gateway database. It can be found using the service OID (e.g. of a subscription id <code>4f4d4a56-c2ef-4450-a461-963279b74198</code>). SQL Query:</p> <pre>select uuid from wsdm_subscription where published_service_oid=<serviceoid>;</pre> |

| Element | Description |
|--|---|
| <reportingendpoint> <service> | The URL of the reporting tool that you are sending the data to (for example, <i>mySplunkserver.com</i>) and the service of the tool that is exposed for accepting the data. <Need to elaborate on "service"> |
| <terminationtime> | The time of expiry for your subscription. Must be in the format yyyy-MM-dd'T'HH:mm:ss:ss:SS (for example, "2015-10-10T12:00:00-05:00") |

Subscribing a Service to the Gateway WSDM Subscription Manager

The message body should be in the format shown in Figure 18.

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:b="http://docs.oasis-open.org/wsn/b-2"
  xmlns:add="http://www.w3.org/2005/08/addressing">
  <soapenv:Header/>
  <soapenv:Body>
    <b:Subscribe>
      <b:ConsumerReference>
        <add:Address>http://<reportingendpoint>:8080/<service></add:Address>
        <!--Optional:-->
        <add:ReferenceParameters>
          <!--You may enter ANY elements at this point-->
        </add:ReferenceParameters>
        <!--Optional:-->
        <add:Metadata>
          <!--You may enter ANY elements at this point-->
        </add:Metadata>
        <!--You may enter ANY elements at this point-->
      </b:ConsumerReference>
      <!--Optional:-->
      <b:Filter>
        <b:TopicExpressionDialect="http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple">
          mowse:MetricsCapability
        </b:TopicExpression>
      </b:Filter>
      <!--Optional:-->
      <b:InitialTerminationTime><terminationtime></b:InitialTerminationTime>
      <!--Optional:-->
      <b:SubscriptionPolicy>
        <!--You may enter ANY elements at this point-->
      </b:SubscriptionPolicy>
      <!--You may enter ANY elements at this point-->
    </b:Subscribe>
  </soapenv:Body>
</soapenv:Envelope>

```

Figure 18: Subscribing a service to the WSDM Subscription Manager

Unsubscribing a Service from the Gateway WSDM Subscription Manager

The message body should be in the format shown in Figure 19.

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:b="http://docs.oasis-open.org/wsn/b-2">
  <soapenv:Header/>
  <soapenv:Body>
    <b:Unsubscribe>
      <b:ConsumerReference>
        <b:SubscriptionId><b>subscription</b></b:SubscriptionId>
      </b:ConsumerReference>
    </b:Unsubscribe>
  </soapenv:Body>
</soapenv:Envelope>
```

Figure 19: Unsubscribing a service from the WSDM Subscription Manager

Renewing a Service to the Gateway WSDM Subscription Manager

The message body should be in the format shown in Figure 20.

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:b="http://docs.oasis-open.org/wsn/b-2">
  <soapenv:Header/>
  <soapenv:Body>
    <b:Renew>
      <b:TerminationTime><b>terminationtime</b></b:TerminationTime>
    </b:Renew>
  </soapenv:Body>
</soapenv:Envelope>
```

Figure 20: Renewing a service to the WSDM Subscription Manager

Conclusion

In this chapter, you have learned how to use existing Gateway capabilities (WSDM Subscription Manager) to retrieve service statistics from a Gateway and periodically push them to a third party reporting tool, using a variety of transport protocols (for example, FTP or HTTP). The XML data generated by the Layer 7 Gateway WSDM notification service can be transformed to any format using assertions in the Policy Manager before sending the data to external systems.

Chapter Eight: Enterprise Logging and Auditing

This chapter describes how to perform enterprise logging and auditing with the Layer 7 Gateway.

Introduction

Many organizations using the Layer 7 Gateway have specific requirements for logging and auditing to satisfy established service level agreements. The Gateway is capable of providing information at runtime about the transactions performed.

This chapter will outline the best practice for logging transactional data from the Layer 7 Gateway. There are several factors to keep in mind when logging data from the Gateway, such as performance considerations and file size issues. The Gateway may perform well while gathering data about transactions, and yet the disk space may be consumed quickly, depending on the amount of information being logged.

The best practice for enterprise logging from the Layer 7 Gateway is to log the data to a disk source off of the Gateway itself, using a queuing mechanism to process the transactional data, then manipulating the data to suit the needs of your enterprise reporting and data trending tools. This helps minimize the enterprise logging overhead on the Gateway and minimize the disk usage of the Gateway appliance's hard drives.

Components of an Enterprise Logging/Auditing System

Enterprise logging with the Layer 7 Gateway includes several components, including:

- the Gateway cluster
- a queue management system
- a database
- a data processing component
- an enterprise reporting toolset

At a high level, the components interact in the following manner.

The Gateway is configured to log messages in policy using the audit sink policy. The audit sink policy is essentially a policy fragment that is run at the end of every policy that is published on the Gateway is executed. The audit sink policy is responsible for gathering the specific data you are interested in and assembling your message to log.

Once the information to log is gathered, a *Route via JMS* or *Route via MQ Native* assertion is put into the sink policy to route your logging message to a queue. This method is efficient because no information is persisted on the Gateway for the log message and no extra processing is required by the Gateway to handle the message.

The queuing system is responsible for gathering the logging messages being pushed to the queue from the Gateway. The queuing system can move the messages from the queue to a database, feed the information to a processing application, or write the messages to a file. Using a reliable queue messaging infrastructure ensures that no important information to be logged will be lost.

As a general best practice, the transactional information that is gathered by the Gateway and written to a queue is put into a transactional table on a database. Transactions that appear in the transactions table of the database can then be read and manipulated by a stored procedure and put into an OLAP star schema. The OLAP cube can then be referenced by either data processing applications to format information into specific inputs needed by applications, or more preferably, enterprise logging and trending tools can be hooked up to the OLAP data directly to generate reports and trending graphical data to be used and viewed by analysts.

Having the Layer 7 Gateway transactional data moved to a database away from the Gateway itself allows for best practice of cleanup and scalability for storage that is required to house all of the data in a data warehousing scenario.

Gathering Audit Data from the Gateway

The audit sink policy is where the transactional data should be gathered for enterprise reporting of the Gateway's transaction information.

Tip: For more information about the audit sink policy, refer to these topics within the *Layer 7 Policy Manager User Manual: Managing the Audit Sink, Working with the Audit Sink Policy*.

- *To enable the audit sink policy:*
 - In the Policy Manager, select **Tasks > Manage Log/Audit Sinks** (browser client: **Manage > Manage Log/Audit Sink**). The Manage Log Sinks dialog is displayed.
 - Click [**Manage Audit Sink**], located in the lower right corner. The Audit Sink Properties dialog is displayed.



Figure 21: Audit Sink Properties

- Configure the two options as required:
 - **Save audit records to Gateway database:** This outputs audit records to the Gateway's database. Depending on the needs of your enterprise, you may disable this option. Many people prefer the audit logs to appear only in the syslog log file, as the tooling for managing the syslog logs helps to manage the file size of the syslog and rollovers and backups as well.
 - **Output audit records via audit sink policy:** This option enables the audit sink policy and is the best practices for enterprise logging on the Gateway. The first time this option is selected, a new “[Internal Audit Sink Policy]” is added to the Services and Policies list in the Policy Manager. This new policy will be executed at the end of each request that is made to the Gateway, after the policies created to handle the Gateway's requests have finished.

For example, you have a Gateway service that protects a backend Web service. The service has a Gateway security policy that is executed when the protecting service is invoked. The service policy is run to completion when a service request is routed to it. After the service policy has finished executing, the audit sink policy is then invoked and the assertions within the audit sink policy are executed. The audit sink policy is meant to be a single point of contact to gather information for audits, and route the audit information somewhere to be stored.

Using the Audit Sink Policy

To use the audit sink policy, you create context variables containing the information you are interested in gathering about a single request. You can then form this context information into a single entity, and write the information off of the Layer 7 Gateway. You may write the information off of the Gateway using any of the methods available in the Layer 7 Gateway policy. For example, you may write the information to a database using a *Perform JDBC Query* assertion or you might route the information to a logging web service via HTTP somewhere. For best practices, you should write the information to a queuing system.

You are able to write information into a queue from the Layer 7 Gateway policy using either the *Route via JMS* or *Route via MQ Native* assertions. These best practices allow the audit information to be gathered and sent outside of normal policy execution. This prevents the overhead of audit data collection from affecting the policy execution time, which is important if you are attempting to minimize policy request latency.

Manipulating/Storing the Audit Data

Using the audit sink policy to gather audit information from the Layer 7 Gateway allows you to manipulate and store the data as required by your enterprise.

Depending on your infrastructure, you might have enterprise audit and logging tooling that accesses and uses data in a specific format. Following the audit log best practice, once you gather the important Gateway request processing information and add it the queue, it can be gathered off the queue and manipulated as necessary.

The data can be taken from the queue and put into a transaction table in a database. This database table can then be processed using stored procedures to massage and spread the data into an OLAP cube for example.

Once the Gateway transactions are processed from a transaction table into the OLAP cube, reporting tooling can use the OLAP structures to display, for example: charts, generate reports, and view trending information about how many requests have been processed by the Gateway, the users who were sending those requests, and how many times the requests were processed or denied for which users.

Sample Audit Sink Policy

Once the audit sink policy has been enabled, you can edit it the same way you would edit any other policy related to security processing in the Layer 7 Policy Manager. You have full access to all of the same policy assertions you have when composing other Gateway policies.

Figure 22 shows a sample audit sink policy.

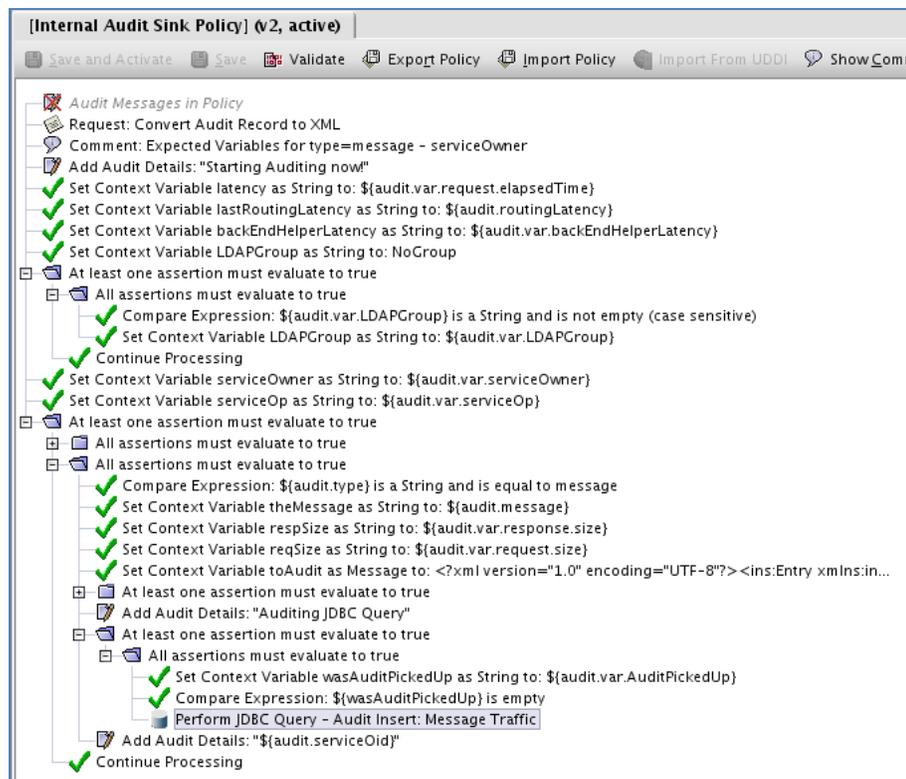


Figure 22: Sample audit sink policy

The audit sink policy has access to any context variables that were set during the execution of your Gateway's security service policy. When composing your policies, you can include *Set Context Variable* assertions to set the data that you are interested in auditing.

Once your security policy has finished processing the incoming request, the audit policy is run and has access to the context variables the security policy set. In this example, the information such as the routing latency and elapsed times were set in context variables during security policy execution. They are then referenced and written to the database in the *Perform JDBC Query* assertion along with other information including the user ID that was used to access the service.

Writing information to a database is ideal when setting up an audit sink policy in a development environment. You can use a development database to capture the testing information and perform queries. Once the audit sink policy is migrated to a production environment, the information can be assembled into an XML message or custom message type, and the *Perform JDBC Query* assertion can be replaced with a *Route via JMS* or *Route via MQ Native* assertion to drop the information to a queue for processing.

Dropping messages onto a queue in production is recommended to help minimize outbound overhead from the Gateway by using a “fire and forget” method of passing on the audit information.

Conclusion

Enterprise logging and auditing of Layer 7 Gateway events can be integrated easily with your enterprise audit solution. You can use the audit sink policy to gather the necessary audit information using standard policy assertions. The audit sink policy can be modified and versioned just like other service policies on your Gateway.

Using the audit sink policy, you only need to configure it once. After that, it will automatically route audit messages to a queuing system, removing any potential auditing overhead that could incur during the Gateway's message processing.

The Gateway auditing system is flexible. Even if messages are not sent to a queue, the assertions available in the policy editor can be used to send audit messages directly to a database, or over the network via HTTP, to an auditing application or subsystem of your choice.

You have full control of what data you collect and store, which may be based on your enterprise audit rules and internal service level agreements.

Chapter Nine: Monitoring Using Splunk

This chapter describes how to monitor the Layer 7 Gateway using the Splunk enterprise software package.

Introduction

Once a Layer 7 Gateway is deployed within an organization, different groups may demand machine and service statistics. This chapter will help you configure the Gateway appliance in retrieving the data needed to build these statistics.

Previous chapters described how to use the Gateway functionalities included in the deployment to monitor the Gateway.

This chapter describes how to use Splunk to analyze and display the data gathered from your Gateway.

What is Splunk?

Splunk is enterprise software used to monitor, report, and analyze the machine data produced by the applications, systems and infrastructure that run a business. Splunk lets users search, monitor and analyze machine-generated data via web-style interface. Splunk captures indexes and correlates real-time data in a searchable repository from which it can generate graphs, reports, alerts and dashboards.

There are two components to using Splunk to retrieve machine statistics from the Gateway appliance:

- Splunk server, which is used to monitor and view graphs through a Web interface.
- Splunk forwarder, which is installed on the Gateway machine. This allows Splunk to index the Gateway logs (ssg*.log).

Setting Up the Splunk Server

This section describes how to download, install, and configure Splunk.

Step 1: Download Splunk

You can download Splunk at no charge from the Splunk website:

<http://www.splunk.com/download?r=header>

Splunk is available for many different operating systems. This chapter is based upon a Linux 64bit OS.

Note: You will need to create an account in order to download. Be sure to remember your username and password, as you will need it later for downloads.

Step 2: Install Splunk

➤ *To install the Splunk server:*

- Copy the downloaded .rpm file to the server where Splunk will be deployed.

Install the application using this command:

```
rpm -ivh splunk-X.X.X-XXXXX-linux-2.6-x86_64.rpm
```

Start the Splunk daemon for the first time with this command:

```
/opt/splunk/bin/splunk start --accept-license
```

Once the Splunk daemon has started, the Web interface is available at:

<http://<yourservername>:8000/en-US/app/launcher/home>

Tip: If Splunk daemon is not running in the future, execute
`/opt/splunk/bin/splunk start`

Configuring the Server Firewall

You may need to configure your server's firewall to allow access to these ports:

- **Port 8000:** Required for running the Web interface to Splunk.
- **Port 9000:** Required for running the *nix application within the Web console (see page 55).

➤ *To configure the server firewall:*

- Open a privileged shell from the Gateway main menu (see Figure 9 on page 16).

Load the following file in a text editor:

```
/etc/sysconfig/iptables
```

Add the following lines under "custom rules":

```
[0:0] -A INPUT -i eth0 -p tcp -m tcp --dport 8000 -j ACCEPT
```

```
[0:0] -A INPUT -i eth0 -p tcp -m tcp --dport 9000 -j ACCEPT
```

Save and exit the file.

Step 3: Set Up the Splunk Web Console

➤ To set up the Splunk Web console:

- Start the Web console by navigating to this URL:

http://<SplunkServerName>.com:8000

Enter the following default credentials:

Username: admin

Password: changeme

Enter a new password when prompted. Be sure to remember this password.

The Splunk manager console is displayed:

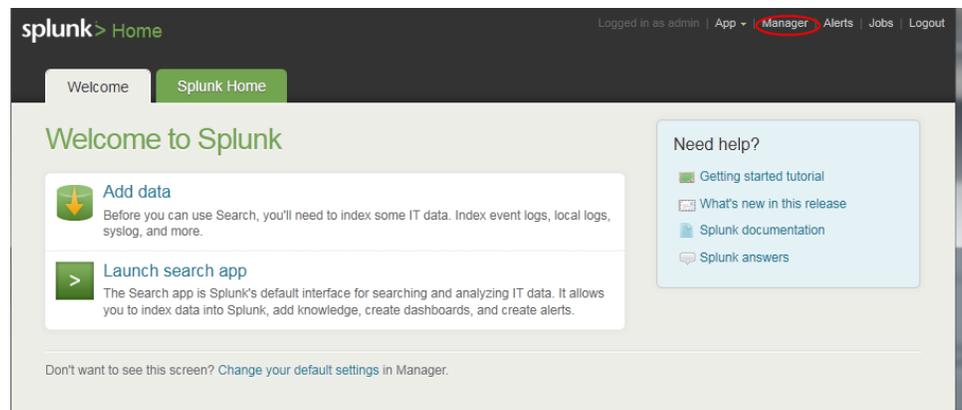


Figure 23: The Splunk manager console

Install the .nix application¹ by doing the following: < need access to system to fix up these instructions>

- Select App > Find More apps > *nix application.
- Click Install Free > Input login details > Enable now > Set up now. Save the default polling intervals. You can amend these at a later stage.
- Return to the Manager home page and select “Forwarding and Receiving” under the Data Section.
- Click on “Configure receiving” > New -> Enter port (e.g. 9000) > Save

Setting Up the Splunk Forwarder

This section describes how to install the Splunk Forwarder (also known as the “Splunk Agent”) on the Layer 7 Gateway server.

¹ For a description of the *.nix application, see: <http://splunk-base.splunk.com/apps/22314/splunk-for-unix-and-linux>

Step 1: Download the Splunk Forwarder

You can download Splunk Forwarder from the Splunk website:

<http://www.splunk.com/download/universalforwarder>

Select your OS from the list. This section will assume a Linux 64bit OS.

Tip: Sign in using the account that you created earlier when downloading Splunk (see page 53).

Step 2: Install the Splunk Forwarder

➤ *To install the Splunk forwarder:*

- Copy the downloaded .rpm file to the server on which it will be deployed.

Install the application using this command:

```
rpm -ivh splunkforwarder-X.X.X-XXXXXX-linux-2.6-x86_64.rpm
```

Once you have installed the rpm, run the Splunk forwarder. Then, in the console, navigate to the installation location of Splunk on your gateway server. Run the following command:

```
bin/splunk start --accept-license
```

This starts the Splunk Forwarder daemon. Next, you will set up the server where the full Splunk application is installed.

From the same location in step 3, run the command substitution hostname and port with the Splunk server settings (for example, splunkserver:9000).

```
./bin/splunk add forward-server <hostname>:<port>
```

Enter your username and password to log into Splunk.

Tip: The default username is **admin** and the password is the new password.

Step 3: Index the Gateway Log Files

➤ *To configure the Universal Forwarder to watch the Gateway's log files:*

- From your Splunk forwarder installation folder, run the following command (entire command is on one line):

```
/opt/splunkforwarder/bin/splunk add monitor  
/opt/SecureSpan/Gateway/node/default/var/logs -active-only True -  
follow-only True
```

The following message should be displayed. This indicates that the Gateway's log files will be indexed by Splunk and can be searched.

```
Added monitor of '/opt/SecureSpan/Gateway/node/default/var/logs'.
```

Step 4: Configure *nix on the Gateway

The *nix application works with Splunk to provide searches, reports, and alerts to help you manage and troubleshoot *nix operating systems.

For more information about the *.nix application, see <http://splunk-base.splunk.com/apps/22314/splunk-for-unix-and-linux>.

This step describes how to download and install the *nix application.

Downloading *nix

You can download the *nix application from the Splunk website:

<http://splunk-base.splunk.com/apps/22314/splunk-for-unix-and-linux>

Copy the downloaded .tar file to the Gateway that you want to monitor.

Installing *nix

- Extract the *nix application to the “/etc/apps/” folder of the location where the Splunk forwarder is installed. For example, if the Splunk Forwarder is located in “/opt/splunkforwarder”, you will extract *nix

```
tar xvfz unix.tar.gz /opt/splunkforwarder/etc/apps/
```

into “/opt/splunkforwarder/etc/apps/”.

Enter the following command to change recursive ownership of the newly created UNIX directory (and all directories and files included)

```
/opt/splunkforwarder/etc/apps/unix
```

```
chown -R splunk:splunk unix
```

Enter the following command to copy the *inputs.conf* file from default to local (entire command is one line):

```
cp /opt/splunkforwarder/etc/apps/unix/default/inputs.conf  
/opt/splunkforwarder/etc/apps/unix/local/inputs.conf
```

Edit the config file and configure or enable the desired inputs; for example, to enable certain inputs, change the disabled value from **1** to **0**.

```
vi /opt/splunkforwarder/etc/apps/unix/local/inputs.conf
```

For more information, refer to the following page on the Splunk website:

<http://docs.splunk.com/Documentation/Splunk/4.2.3/admin/Inputsconf>

Enable the *nix application with the following command.

```
/opt/splunkforwarder/bin/splunk enable app unix
```

Tip: If the message “In handler 'localapps': Application does not exist: unix” (or similar) appears, restart the Splunk Forwarder and then try the command again. See “Restart Splunk Forwarder ” on page 58.

Step 5: Start Using Splunk

➤ To start using Splunk:

- Start your web browser and navigate to the following location to view your new indexed files in Splunk:

`http://<nameofyoursplunkserver>.com:8000/en-US/app/unix/`

An entry in the hosts section should appear at the bottom right of the browser page. If not, refer to the “Troubleshooting” below for tips.

Click on your host to begin viewing the statistics forwarded by the Splunk Forwarder.

Troubleshooting

Check if Splunk Forwarder is working

To verify whether the Splunk Forwarder is working, enter the following command to view the progress of the Splunk logs on the Gateway machine:

```
tail -f /opt/splunkforwarder/var/log/splunk/splunkd.log
```

There should be no connection failures, just *watchedfile* entries in the log.

If there are connection failures, verify that the correct port is set up on the Splunk server and that the port is opened in the firewall. For more information, see “Configuring the Server Firewall” on page 54.

Restart Splunk Forwarder

If you see a message similar to the following when attempting to start the .nix application, it may indicate that the Splunk Forwarder has crashed:

```
“In handler 'local apps': Application does not exist: unix”
```

In this case, restart the Splunk Forwarder with this command:

```
/opt/splunkforwarder/bin/splunk restart
```

Retrieving Data for Gateway Service Statistics

The Layer 7 Gateway has built-in functionality to monitor services and push the service metrics to a third party reporting tool at specified intervals.

This section assumes that you have completed the Splunk configuration described earlier in this chapter.

Prerequisites:

Ensure that there are no services currently subscribed to the WSDM Subscription service for the Gateway being monitored. The automated WSDM subscription script

that you will be running assumes there are no pre-existing subscriptions. <emailed Geoff for an easier way to remove subscription>

Installing the WSDM Auto-Subscription Tool

Installing the WSDM auto-subscription tool is a multi-step process.

Step 1: Install the Gateway Command Line Migration Tool

Contact Layer 7 Technologies to obtain the Command Line Migration Tool. Note that you will not use the tool directly in the WSDM auto-subscription procedure, but it is required by the scripting process.

This step describes how to install the Linux version of the tool.

- *To install the Command Line Migration Tool:*
 - Extract the contents of the *GatewayCommandlineMigration-x.x.zip* file to a folder on your local drive, where “x.x” is the version of the Gateway for which the toolkit is intended.

Change to the directory that was extracted from the .zip file (for example, *GatewayCommandLineMigration-x.x*).

Ensure that JAVA_HOME is set:

```
echo $JAVA_HOME
export $JAVA_HOME=/opt/SecureSpan/JDK/bin
```

Note: If the environment variable JAVA_HOME is not set, then the “--jdk” argument can be used to specify the location of the Java SDK/JRE; for example: `./script.sh --jdk /usr/java/j2sdk1.6.0_23`.

Step 2: Set up the WSDM Subscription Manager and WSDM Metrics Notifier

Set up the Gateway WSDM Subscription Manager and WSDM Metrics Notifier in the Layer 7 Policy Manager.

For detailed instructions on how to do this, see “Setting Up the Layer 7 Gateway” on page 41.

Step 3: Copy the WSDM auto-subscription file

- *To copy the WSDM auto-subscription file:*
 - Copy the file *wsdmSubscriptionTool.zip* to the location of the Gateway Command Line Migration Tool folder. For example:

```
cp wsdmSubscriptionTool.zip /opt/
GatewayCommandlineMigration/GatewayCommandlineMigration-x.x
```

Extract the .zip file into that location. This creates a new *scripts* folder.

Step 4: Configure the WSDM Auto-Subscription Tool

Modify the configuration files in the *scripts* folder as follows:

config.pl

Customize this script for your own Gateway as per Table 4 below.

Table 4: Customizing the config.pl script

| Attribute | Description | Substitution | Must Change? |
|----------------------------|--|--|--------------|
| posturl | The URL used to subscribe each service from within the script | Replace "<hostname>" with the hostname of your Gateway. | YES |
| host | The hostname of your Gateway | Replace "<hostname>" with the hostname of your Gateway. | YES |
| hostUsername | The host username of your policy manager | Replace <hostManagerUsername> with the host username of your manager. | YES |
| hostPassword | This is the host password of your policy manager. | Replace <hostManagerPassword> with the host password of your manager. | YES |
| consumerHostService | The endpoint for the statistics (i.e., the 3 rd party reporting tool to send data to). The method is assumed to be HTTP(s). | Enter value as an absolute URL: http(s)://<reportingserver>:<reporting serverport>/service | YES |
| hostTrustServerCertificate | Should the server's certificate be trusted even if it is not signed by a certificate authority? | Enter yes or no . Default: no | NO |
| hostTrustServerHostname | Should the server be trusted even if its certificate has a different hostname than specified with -host? | Enter yes or no . Default: no | NO |
| terminationTime | Termination time for the subscriptions to the host | Value is in format: <i>yyy-MM-dd'T'HH:mm:ss-HH:mm</i> Default: 2015-10-10T12:00:00-05:00 | NO |
| logdir | Directory for outputting logs from the script | Enter a location or leave blank to use the same directory as the script location. | NO |

| Attribute | Description | Substitution | Must Change? |
|---|---|---|---------------|
| | | Default: blank | |
| oidServicesFileNew & oidServicesFileCurr | The file system used for comparison of newly registered/deleted services in the Gateway | File names are for script use only. Default: oids.new and oids.current | NO |
| soapMessageToUnsubscribe1 & soapMessageToUnsubscribe2 | The unsubscription SOAP body message sent as the payload to the subscription service on the Gateway | N/A | DO NOT CHANGE |
| soapMessageToSubscribe1 & soapMessageToSubscribe2 & soapMessageToSubscribe3 | The subscription SOAP body message sent as the payload to the subscription service on the Gateway | N/A | DO NOT CHANGE |

retrieveListServices.cfg

Customize this script for your own Gateway as per Table 5 below.

Table 5: Customizing the retrieveListServices.cfg script

| Attribute | Description | Substitution | Must Change? |
|-----------------------------|---|--|--------------|
| commandLineMigrationJarPath | The location of the Gateway Command Line Migration Tool .jar file | Default assumes that you have extracted the scripts to the location of the Gateway Command Line Migration Client Default: J../GatewayCommandlineMigration.jar | YES |
| host | The hostname of your Gateway | Replace "<hostname>" with the hostname of your Gateway. | YES |
| hostUsername | The host username of your Gateway manager | Replace "<hostManagerUsername>" with the host username of your manager. | YES |
| hostPassword | The host password of your Gateway manager | Replace "<hostManagerPassword>" with the host password of your manager. | YES |
| hostTrustServerCertificate | Should the server's certificate be trusted even if it is not signed by a certificate authority? | Enter yes or no . Default: no | NO |

| Attribute | Description | Substitution | Must Change? |
|---|--|--|--------------|
| hostTrustServerHostname | Should the server be trusted even if its certificate has a different hostname than specified with <code>-host</code> ? | Enter yes or no . Default: no | NO |
| oidServicesFileNew & oidServicesFileTemp | The file system used for comparison of newly registered/deleted services in the Gateway | filenames are for script use only. Default: oids.new and oids.tmp | NO |

Step 5: Run script manually or create cron job

You can now use the scripts for auto subscription/unsubscription by creating a *cron* job or a scheduled task in Windows.

Splunk Scripting

Splunk also has a scripting mechanism where you can drop a shell script in the Splunk bin location and call it periodically from Splunk. You can call it from either *inputs.conf* or via the web interface.

For information, refer to the following pages.

<http://docs.splunk.com/Documentation/Splunk/4.2.3/Data/Setupcustominputs>

<http://docs.splunk.com/Documentation/Splunk/4.2.3/Data/JMXwriteyourown>

<http://docs.splunk.com/Documentation/Splunk/4.2.3/Developer/ScriptedInputsIntro>

Step 6: Monitor the log file

Running the script creates a log file that is constantly updated. You can monitor the scripts progress by using the Linux *tail* command:

```
tail -f 1-3-logfile.log
```

Understanding the WSDM Auto-Subscription Tool

To help you understand the effects of the WSDM Auto-Subscription Tool, refer to the following shell script sequence diagram. This diagram shows the sequence of events triggered by running the shell script *retrieveListServices.sh*.

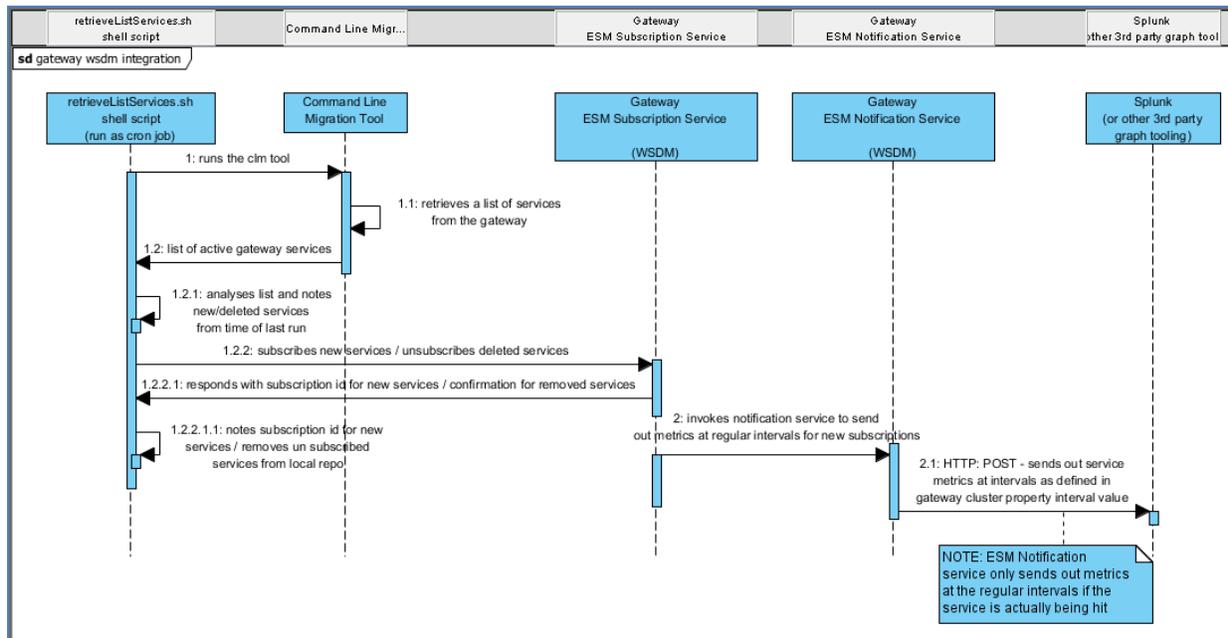


Figure 24: Shell script sequence diagram for retrieveListServices.sh

Third Party Licensing

The Gateway WSDM Subscription Manager allows data to be pushed to a third-party tool at preset intervals. This chapter has described how to configure Splunk as your third-party tool. Splunk provides a fully featured free trial for 60 days. Refer to the following page on the Splunk website for more details:

<http://www.splunk.com/view/free-vs-enterprise/SP-CAAEE8W>

Tip: The Gateway provides a cluster property to control the data volume that is pushed to third party tools. This could result in an overall lower volume of statistics compared to the “per request” method of service statistics, allowing you to pursue more economical solutions.

Conclusion

In this chapter, you learned how to retrieve machine statistics and service statistics from a Layer 7 Gateway Appliance or Virtual Appliance. You also learned how to use this gathered data to display trending on a 3rd party reporting tool such as Splunk.

You learned how you can use the Splunk Forwarder to index log files on the Layer 7 Gateway and used this log file for clever graphing on the Splunk Server. The Splunk Server also has the capability to monitor the appliance/VM CPU usage, memory, and other machine statistics. For service statistics, you learned how to use internal Gateway capabilities (Gateway WSDM Subscription Manager) to push data periodically to Splunk using HTTP as the transport protocol and XML as the

communication protocol. Either of these protocols can be replaced to suit the requirements of the reporting tool you wish to push data to. For example, you can use FTP instead of HTTP; the XML data can be transformed to any format using the assertions in the Layer 7 Policy Manager before sending the data to an external system.

Chapter Ten: Nagios Integration

This chapter describes how to make the Layer 7 Gateway work with Nagios.

Introduction

When the Layer 7 Gateway is deployed within an organization, a number of groups may be interested in machine and service statistics. The previous chapters have described how you can use core tools such as SNMP and WSDM, and external third party tools, such as Splunk, which are built into the Gateway. This chapter describes how to gather statistics from a Layer 7 Gateway using the Nagios Core system and monitoring application.

Setting Up Nagios

This section describes how to download and install Nagios.

Prerequisites:

Nagios relies on the following tools: *httpd*, *gcc*, *glib*, *glibc-common*, *gd* and *gd-devel*. These tools must be installed on the target machine to successfully install Nagios. Install these tools using a utility such as *yum* (for example, "*yum install gcc*").

Downloading Nagios

You can download Nagios from their website:

<http://www.nagios.org/download>

Be sure to select the latest stable release. Note that Nagios Core is only available for Linux and the installation package is a .tar-gz file. This chapter will assume that the package is being installed on a Linux Red Hat 5 machine.

Nagios plugins can also be downloaded from this location:

<http://www.nagios.org/download/plugins>

Creating an Account

- *To create a Nagios account*
 - Create a Nagios user account. To do this, enter the following commands to create the user, and then change its password:

```
/usr/sbin/useradd -m nagios
```

```
passwd nagios
```

Enter these commands to create a nagcmd group with the nagios user:

```
/usr/sbin/groupadd nagcmd
```

```
/usr/sbin/usermod -a -G nagcmd nagios
```

```
/usr/sbin/usermod -a -G nagcmd apache
```

Both the nagios and apache users are added to this group to allow external commands to be submitted from the web interface.

Installing Nagios

➤ *To install Nagios:*

- Copy the two archive files to the target machine.

Start the process by decompressing the core nagios archive file. For example, if the file is named *nagios-3.2.3.tar.gz*, use this command:

```
tar xzf nagios-3.2.3.tar.gz
```

Run the configure script to prepare the installation:

```
cd nagios-3.2.3
```

```
./configure --with-command-group=nagcmd
```

If you intend to use the Perl plugins (recommended), use this command to run the configure script instead:

```
./configure --with-command-group=nagcmd --enable-perl-modules
```

Install the package using the *make* utility:

```
make all; make install; make install-init; make install-config; make  
install-commandmode; make install-webconf
```

Configure the user and email address from which Nagios will send notifications. This step is optional. You will first open the *contacts.cfg* file in a text editor. The command below uses the *vi* editor:

```
vi /usr/local/nagios/etc/objects/contacts.cfg
```

In *.cfg* file, create a new contact and add the email information that you desire. The contact is defined in the "Contacts" section of the file using the following format.

```
define contact{  
    contact_name <user> ;Short name of user  
    use generic-contact ;Inherit default values from generic-  
                           contact template (defined above)  
    alias Nagios Admin ;Full name of user
```

```

email <email address> ;Change this to your e-mail address
}

```

Save and exit the file.

Now create the same user as `<user>`, which was specified in the `define contact {` code excerpt in Step 5.

Execute the command

```
htpasswd -c /usr/local/nagios/etc/htpasswd.users <user>
```

by replacing `<user>` with the specified `contact_name`.

Save the changes and exit the editor.

Restart the HTTP server with this command:

```
service httpd restart
```

At this point, the Nagios package is ready to be used. You can access the application by navigating to this the URL in a Web browser:

```
http://<yourserver>/nagios/
```

Installing the Plugins

Nagios accepts plugins, allow you to monitor data from a variety of different sources. A set of common plugins can be obtained from the Nagios website. This chapter will deal with the plugin for SNMP capability.

Note: For the SNMP plugin to be installed correctly, `net-snmp` and `net-snmp-utils` need to be installed on the system. You should use a tool such as `yum` to install these tools before installing the Nagios plugins.

➤ To install a Nagios plugin:

- Download the plugin and copy it to the target server.

Decompress the `.tar.gz` archive file. For example, if the name is `nagios-plugins-1.4.15.tar.gz`, use the following command:

```
tar xzf nagios-plugins-1.4.15.tar.gz
```

Change to the decompressed folder. For example:

```
cd nagios-plugins-1.4.15
```

Configure and install the plugins with these commands:

```

# ./configure --with-nagios-user=nagios --with-nagios-group=Nagios
#make
#make install

```

Starting Nagios

The steps below describe how to add Nagios to the list of system services, to be started automatically when the system starts up.

➤ To start Nagios:

- Run the following commands to add Nagios to the list of system services:

```
#chkconfig --add nagios
```

```
#chkconfig nagios on
```

Each time the Nagios configuration files are modified including during installation, they should be tested. You can do this with the following command:

```
#!/usr/local/nagios/bin/nagios -v #/usr/local/nagios/etc/nagios.cfg
```

If the test returns no warnings or errors, you can start Nagios with this command:

```
#service nagios start
```

You can access the Nagios web application using the following URL:

```
http://<machinename>/nagios.
```

From this URL, you can monitor machines, create notifications, schedule various events, and administer the Nagios server.

Tip: To log into this machine, you will need the *nagiosadmin* user created with the *htpasswd* command. See Step 7 under “Installing Nagios” on page 66.

Configuring Nagios

The Nagios configuration settings are stored in the *nagios.cfg* file, which is found in this directory:

```
<NagiosInstallationDirectory>/etc/nagios.cfg
```

If the default installation directory is used, it will be found here:

```
/usr/local/nagios/etc/nagios.cfg
```

Main Configuration

The *nagios.cfg* file stores the entire configuration for Nagios, including the enablement of various options, location of log files and specific configuration files. In its default configuration, it will provide the location of the configuration files for the commands executed by Nagios, the hosts monitored by Nagios, and the specific services monitored by Nagios using the aforementioned commands on those hosts.

For more information about this configuration file, refer to the Nagios online documentation here: http://nagios.sourceforge.net/docs/3_0/configmain.html.

SNMP Configuration

The *switch.cfg* configuration file is vital in Nagios' SNMP capability to monitor the Layer 7 Gateway services. This file is used to monitor network devices using technology such as SNMP.

➤ To enable the *switch.cfg* file:

- Open the *nagios.cfg* file in a text editor.

Uncomment the following line by removing the leading pound symbol (#):

```
#cfg_file=/usr/local/nagios/etc/objects/switch.cfg
```

This enables a file called *switch.cfg* at the location specified. This file contains a sample configuration for a device named *linksys-srw224p*. This configuration will include the definition of the host, the definition of a hostgroup associated with the host, and services that will run commands against the host.

Note: Nagios will try to access *linksys-srw224p* when this configuration change is loaded. It is recommended that you delete or comment out this line as the host is nonexistent. This sample configuration is intended to be used as a template for creating checks on other devices.

Nagios will execute services that will generally contain reference to a command (as defined in *<NagiosInstallationDirectory>/etc/objects/commands.cfg*), a host (as defined in another file or in the switch file), and other configurations. A sample service definition looks like the following:

```
define service{
    use generic-service ;Inherit values from a template
    host_name gateway.l7tech.com
    service_description Warehouse Response Times During Last Hour
    check_command check_snmp! - C 17 - o
    1. 3. 6. 1. 4. 1. 17304. 7. 1. 13. 21233664 - w 500: 1000 - c : 2000
    normal_check_interval 1 ; Check the service every 5
    minutes under normal conditions
    retry_check_interval 1 ; Re-check the service every
    minute until its final/hard state is determined
}
```

If the Nagios plugin is installed, all the elements for SNMP are in place to utilize it as a service according to the above definition. The *commands.cfg* file should contain a definition for the *check_snmp* command. This file requires a service that calls the *check_snmp* command with an MIB OID, and a host that has been used by the “define host” syntax, which is in the *switch.cfg* file.

The *check_snmp* program and similar plugins can be found in this folder:

```
<NagiosInstallationDirectory>/libexec/
```

To learn more about the syntax of the various definitions, refer to the online Nagios documentation:

http://nagios.sourceforge.net/s/3_0/toc.html

Loading Configuration Changes into Nagios

Each time a configuration changes in the Nagios configuration files, retest them before loading them into Nagios:

```
#!/usr/local/nagios/bin/nagios -v #/usr/local/nagios/etc/nagios.cfg
```

If no major errors or warnings are reported by the utility, you can reload the configuration:

```
#!/etc/rc.d/init.d/nagios reload
```

Using Perl Scripts as Nagios Commands

A powerful feature of Nagios is its ability to use Perl scripts to monitor other devices via a built-in Perl interpreter. To enable this interpreter during configuration time, include the following option when calling the `./configure` script within the installation package:

```
--enable-embedded-perl
```

To add new Perl scripts as commands, copy the scripts into the `<NagiosInstallationDirectory>/libexec/` folder. Ensure that proper permissions are in place to execute, and that an entry is created for the perl script in the `commands.cfg` file. Once this is complete, the command can be referenced in the `switch.cfg` configuration file.

For more information about using the Perl plugin for Nagios, refer to this page:

<http://nagiosplugins.org/faq/development/nagios-plugin-perl>

Conclusion

By utilizing Nagios, you can retrieve statistical information from a Layer 7 Gateway Appliance or Virtual Appliance. Nagios enhances the operational monitoring infrastructure of a production computing environment. Integrating Nagios to the Layer 7 Gateway can enhance your monitoring infrastructure throughout the organization. Nagios can be easily set up and configured to pull important statistics from a Layer 7 Gateway that is helpful in analyzing Gateway usage.

Summary

The Layer 7 Gateway has many options available for appliance monitoring and integration with your enterprise operations and service-level agreements. Whether you in operations and are looking for SNMP connectivity to the Gateway, or a business analyst looking to create reports for Gateway load trends, the Layer 7 Gateway can provide the monitoring support needed for different teams. Command line tools can be used to view Gateway statuses, and its migration tool can migrate policies and service configurations between different Gateway environments. Administrators can use the Gateway's dashboard feature to view current load and status of accessed services in real time.

