HP Operations Analytics

Software Version: 2.20

Operations Analytics Configuration Guide

Document Release Date: May 2015
Software Release Date: December 2014

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© Copyright 2013 - 2015 Hewlett-Packard Development Company, L.P.

Trademark Notices

Microsoft and Windows are trademarks of the Microsoft Group of companies.

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: https://softwaresupport.hp.com/group/softwaresupport/search-result?keyword=.

This site requires an HP Passport account. If you do not have one, click the **Create an account** button on the HP Passport Sign in page.

Support

Visit the HP Software Support web site at: **https://softwaresupport.hp.com**

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software Support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to **https://softwaresupport.hp.com** and click **Register**.

To find more information about access levels, go to: **https://softwaresupport.hp.com/web/softwaresupport/access-levels**

HP Software Solutions & Integrations and Best Practices

Visit HP Software Solutions Now at **https://h20230.www2.hp.com/sc/solutions/index.jsp** to explore how the products in the HP Software catalog work together, exchange information, and solve business needs.

Visit the Cross Portfolio Best Practices Library at **https://hpln.hp.com/group/best-practices-hpsw** to access a wide variety of best practice documents and materials.

Contents

Chapter 1: About this Guide 8

For Information about Operations Analytics 8

Environment Variables used in this Document 9

System Requirements 9

Terminology Used in this Document 10

Chapter 2: Deployment Prerequisites 12

Data Sources used in Operations Analytics 12

Supported Deployments 14

How Tags are Used in Operations Analytics 14

Chapter 3: Configuring Tenants and Collections 17

Preparing for Configuring Collections 20

Creating Tenants 20

Important Tenant Information 21

Registering Each Collector Appliance 24

Configuring Collections using Predefined Templates 25

Important Tenant Information 26

Configuration Steps 26

Configuring an HP Operations Smart Plug-in for Oracle Collection 27

Configuring an HP Operations Agent Collection 30

Configuring an NNM ISPi Performance for Metrics Component Health Collection 35

Configuring an NNM ISPi Performance for Metrics Interface Health Collection 40

Configuring an HP Operations Manager (OMi) Events Collection 44

Setting the Correct Time Zone 44

Configuring the HP OMi Events Collection (Automated Method) 45

Configuring an HP Operations Manager (HPOM) Events Collection 49

Setting the Correct Time Zone 49

Configuring an HP BSM RTSM Configuration Item (CI) Collection 53

Setting the Correct BSM User Name Permissions 53

Configuration Steps 55

Configuring an HP Business Process Monitor Collection 60

Setting the Correct BSM User Name Permissions 62

Configuration Steps (Automated Method) 63
Configuring ArcSight Logger Out of the Box Smart Connector Collections 67
Configuring an NNMi Custom Poller Collection 70
Configuring HP Operations Manager (HPOM) (Creating a Database User Account on an HPOM Database Server) 78
Configuring Collections for Custom Data Sources 87
Configuring a Custom CSV Collection 88
Important Prerequisite Steps 89
Configuration Steps 92
Troubleshooting the Custom CSV Collection 94
Removing the Registration and Data for a Custom CSV Collection 94
Configuring a Custom SiteScope Collection 95
Configuring a Structured Log Collection 101
Configuring Logger to Forward CEF Messages to Operations Analytics 101
Configuring the Maximum Logger Sessions 105
Configuring a Structured Log Collection 106

Chapter 4: Creating, Applying, and Maintaining Tags 116
Adding Tags 116
Listing Tags 116
Deleting Tags 117

Chapter 5: Communicating Collection Names and Meta Data Information to your Users 118

Chapter 6: Configuring Secure Access to Operations Analytics 119
Configuring SSL for the Operations Analytics Server Appliance 119
Configuring SSL with a Certificate Authority (CA) Signed Certificate for the Operations Analytics Server Appliance 119
Configuring SSL with a Self-Signed Certificate for the Operations Analytics Server Appliance 123
Editing the SSL Configuration for the Operations Analytics Server Appliance 124
Disabling the SSL Configuration for the Operations Analytics Server Appliance 124
Managing the Operations Analytics Keystore and Truststore for the Operations Analytics Server Appliance 125
Configuring SSL for the Operations Analytics Collector Appliance 127
Configuring SSL with a Certificate Authority (CA) Signed Certificate for the Operations Analytics Collector Appliance 127
Configuring SSL with a Self-Signed Certificate for the Operations Analytics Collector Appliance 130
Editing the SSL Configuration for the Operations Analytics Collector Appliance 131

Disabling the SSL Configuration for the Operations Analytics Collector Appliance 132
Managing the OPSA Keystore and Truststore for the Operations Analytics Collector Appliance 133
Configuring the HTTP and HTTPS Port for the Operations Analytics Collector Appliance 135
Configuring the HTTP and HTTPS User Name and Password for the Operations Analytics Collector Appliance 136
Configuring and Enabling Single Sign-on to Access Operations Analytics 136
Disabling Single Sign-on to Access Operations Analytics 138
Configure Two-Way SSL Authentication for Accessing HP ArcSight Logger 139
Configuring User Authentication using Public Key Infrastructure (PKI) to Access Operations Analytics 145
Disabling User Authentication using Public (PKI) to Access Operations Analytics 147
Editing User Authentication using Public (PKI) to Access Operations Analytics 148
Configuring SSL for Communication between Vertica and Operations Analytics 148
Enabling SSL Communications between the Operations Analytics Server Appliance and Vertica 149
Disabling SSL Communications between the Operations Analytics Server Appliance and Vertica 156
Enabling SSL Communications between the Operations Analytics Collector Appliance and Vertica 159
Disabling SSL Communications between the Operations Analytics Collector Appliance and Vertica 159
Configuring SSL for the SMTP Server Used for OpsA Alerts 160

Chapter 7: Maintaining Operations Analytics Collections 161
Managing Collected Data File Usage with Existing Delete Policies 161
Troubleshooting Operations Analytics Collections 162
Checking a Collector's Status 162
Troubleshooting Configurations from the Operations Analytics Server Appliance 163
Troubleshooting the Absence of Collection Data 163

Chapter 8: Maintaining Operations Analytics 167
Adding Operations Analytics Server Appliances 167
Checking Operations Analytics System Health 167
Configuring Operations Analytics Health 168
Using Operations Analytics Health 169
Deleting a Tenant 169

Removing a Collection Registration for a Tenant 170
Configuring the Operations Analytics Log File Connector for HP ArcSight Logger 172
Installing the Operations Analytics Log File Connector for HP ArcSight Logger 173
Configuring the Operations Analytics Log File Connector for HP ArcSight Logger 175
Option 1) Change Logger Server 176
Option 2) List Log Folders 176
Option 3) Add Log Folder 177
Option 4): Edit Log Folder 177
Option 5): Delete Log Folder 178
Option 6): Test Log Folders 178
Option 7): Exit 179
Filtering HP ArcSight Logger Queries 179
Manually Starting and Stopping the Operations Analytics Log File Connector for HP ArcSight Logger 180
Configuring the Operations Analytics Log File Connector for HP ArcSight Logger to Run as a Service 180
Stopping the Operations Analytics Log File Connector for HP ArcSight Logger from Running as a Service 186
Troubleshooting the Operations Analytics Log File Connector for HP ArcSight Logger 189
Using Other ArcSight Connectors 190
Raw Log Message 190
Setting Hostname, Application, and Process Names 198
Uninstalling the Operations Analytics Log File Connector for HP ArcSight Logger 199
Exporting and Importing Operations Analytics Dashboards 199
Exporting and Importing Dashboards Among Operations Analytics Tenants 200
Log Files in Operations Analytics 201
Using and Maintaining Audit Log Files 201
Maintaining the Operations Analytics Database 203
Backing up and Restoring Data 203
Managing Vertica Data 204
Setting Collection Retention Periods 205
Managing Data in Logger 205
Modifying Unit Scaling on Collected Data 206
Monitoring Operations Analytics Processes 207
Restarting the Operations Analytics Server and Collector Appliance 208
Using Parametric Dashboards 208
Additional Information 215

Configuring an HP Operations Smart Plug-in for Oracle Collection (Detailed Methods) 215
Configuring an HP Operations Agent Collection (Detailed Method) 221
Configuring an NNM ISPi Performance for Metrics Component Health Collection (Detailed Method) 226
Configuring an NNM ISPi Performance for Metrics Interface Health Collection (Detailed Method) 230
Configuring the HP OMi Events Collection (Detailed Method) 235
Setting the Correct Time Zone 236
Configuring the HP OMi Events Collection (Detailed Method) 236
Configuring an HP Operations Manager (HPOM) Events Collection (Detailed Method) 240
Setting the Correct Time Zone 240
Configuration Steps 241
Configuring an HP BSM RTSM Configuration Item (CI) Collection (Detailed Method) 246
Setting the Correct BSM User Name Permissions 246
Configuration Steps 247
Configuring an HP Business Process Monitor Collection (Detailed Method) 253
Setting the Correct BSM User Name Permissions 255
Configuration Steps (Manual Method) 257
Configuring ArcSight Logger Out of the Box Smart Connector Collections (Detailed Method) 260
Configuring a Custom CSV Collection (Detailed Method) 267
Important Prerequisite Steps 268
Configuration Steps 271
Troubleshooting the Custom CSV Collection 279
Removing the Registration and Data for a Custom CSV Collection 280
Configuring a Custom SiteScope Collection (Detailed Method) 281
Detailed Configuration Steps 282
Generating and Configuring Templates (Custom SiteScope Collection) 283
Configuring SiteScope for Integrating Data with Operations Analytics (Manual Method) 289
Task 1: Creating a SiteScope Tag 289
Task 2: Using the New SiteScope Tag to Mark the Monitor or Monitor Groups 290
Task 3: Creating a New Data Integration Preference 292
Daylight Savings Time Codes 294

Send Documentation Feedback 306

Chapter 1: About this Guide

Read this guide to understand the concepts required to install, configure, and use Operations Analytics OpsA) most effectively, including helpful tips and how to set up collections after installation.

For Information about Operations Analytics

To obtain a complete set of information about Operations Analytics (OpsA), use this guide along with other OpsA documentation. The table below shows all OpsA documents to date.

Documentation for Operations Analytics

What do you want to do?	Where to find more information
I want to quickly install OpsA	*Operations Analytics Quick Start Guide*
I want to install OpsA	*Operations Analytics Installation Guide*
I want to configure and maintain OpsA	*Operations Analytics Configuration Guide*
I want to upgrade OpsA 2.10 to OpsA 2.20	*Operations Analytics Upgrade Guide*
I want to obtain help about the Operations Analytics console	*Operations Analytics Help*
I want to find the hardware and operating system requirements for OpsA	*Operations Analytics Support Matrix*
I want to read a list of the new features and review any last minute issues for OpsA	*Operations Analytics Release Notes*
I want to open a view from HP OMi to OpsA	*Integration with HP Operations Manager i: Operations Analytics 2.20 White Paper*
I want to view a list of software products integrated with OpsA	See the list of integrations for Operations Analytics and other HP products at http://support.openview.hp.com/sc/solutions/index.jsp
I want to obtain assistance developing AQL queries.	*AQL Developer Guide*
I want to obtain assistance configuring the OpsA - OneView integration.	*Operations Analtyics - OneView Integration Guide*

Environment Variables used in this Document

This document refers to the following environment variables and other useful directories when explaining installation and configuration instructions for the Operations Analytics (OpsA) Software, including the OpsA Server Appliance and the OpsA Collector Appliance. The environment variables are set automatically for the opsa user who can use all OpsA functionality and has access to data at the tenant level. See " Configuring Tenants and Collections " on page 17 for more information.

Table 1: Environment Variables

Variable Name	Path	Operations AnalyticsServer Appliance or Collector Appliance
OPSA_HOME	/opt/HP/opsa	Server and Collector Appliances
JAVA_HOME	/opt/HP/opsa/jdk	Server and Collector Appliances

Table 2: Other Useful Directories

Folder Name	Path	Operations AnalyticsServer Appliance or Collector Appliance
JBOSS Home Directory	/opt/HP/opsa/jboss	Server Appliance
JDK Folder	/opt/HP/opsa/jdk	Server and Collector Appliances
scripts Folder	/opt/HP/opsa/scripts	Server and Collector Appliances
conf Folder	/opt/HP/opsa/conf	Server and Collector Appliances
data Folder	/opt/HP/opsa/data	Server and Collector Appliances
log Folder	/opt/HP/opsa/log	Server and Collector Appliances
lib Folder	/opt/HP/opsa/lib	Server and Collector Appliances
bin Folder	/opt/HP/opsa/bin	Server and Collector Appliances
Vertica Database Installation Folder	/opt/vertica	Server and Collector Appliances have the Vertica client installed in this folder

System Requirements

See the *Operations Analytics Support Matrix* for the hardware and operating system requirements for Operations Analytics.

Any command examples shown in this document as being run by an opsa user can also be run by a root user.

`$OPSA_HOME` is set to `/opt/HP/opsa` in the OpsA Server Appliance

Terminology Used in this Document

Analytic Query Language (AQL): The more advanced offering of two query languages supported by Operations Analytics (OpsA). Use AQL when the Phrased Query Language (PQL) syntax is not specific enough to return the data you need. When using AQL, it is helpful if you have programming or scripting skills as well as some knowledge of databases. See *About Analytics Query Language (AQL) Functions* in the *Operations Analytics Help* for more information.

Collection: A collection defines the data to be collected and corresponds to a database table in which the OpsA Collector Appliance stores the data. Collections can be separated by tenant and collection information cannot be shared among tenants

Custom Collections The list of collections supported by the OpsA Server Appliance that do not have predefined templates.

Collector Appliance: This virtual appliance is the server used to manage the data collections.

Data Sources: OpsA collects metrics, topology, event, and log file data from a diverse set of possible data sources.

HP Service Health Analyzer (SHA): HP Service Health Analyzer analyzes abnormal service behavior and alerts IT managers of service degradation before an issue affects their business.

Link Tags: Special tags used to relate collection information. Create the same link tag for each collection you want to link together.

Meta Model: A way to describe the data to collect for analysis; it includes the construction and development of the frames, rules, constraints, models and theories applicable and useful for modeling a predefined class of problems.

Metrics: Structured data that is typically collected from HP's existing management products, other data files or from other 3rd party management software. A metric is a measurement of one attribute at specific point in time for a specified sub-entity or resource (such as CPU utilization). A metric is based on the most recent user-initiated search query.

Outlier or **Outliers**: Data that is outside of the normal range based on the data collected to date.

predefined Collection Templates: The list of predefined collection templates that reside on the OpsA Server Appliance for the collections OpsA supports by default.

Phrased Query Language (PQL) : The less advanced offering of two query languages supported by OpsA. Use PQL in the early stages of troubleshooting a problem. With this approach, type a word or phrase that begins to describe the type of problem you want to resolve and then select from the list of suggestions provided by OpsA. See *About the Phrased Query Language* in the *Operations Analytics Help* for more information.

Raw Logs: These are log messages as they appear from the log management application with which OpsA is integrated. These log files must be configured using the log file management software supported by OpsA. See the*Operations Analytics Support Matrix* for more information.

Server Appliance: This virtual appliance is the OpsA Server.

Structured Logs: Structured logs are fragments of log file data read by Operations Analytics (OpsA) from HP ArcSight Logger. This log information is stored (as collections) in OpsA. These collections exist so that users can perform analytics on the log file contents. For example, users might want to query for all outliers by host name and application for a particular time range.

Tenant: OpsA gathers metrics, topology, event, and log file data from a diverse set of possible data sources. Tenants enable you to separate information from these data sources into groups, such as collections, user accounts, and user groups. Collections can be separated by tenant and collection information cannot be shared among tenants.See "Creating Tenants " on page 20 for more information.

Virtual Appliance: A virtual appliance, also referred to as **appliance** in this document, is a self contained system that is made by combining a software application, such as OpsA software, with just enough operating system for it to run optimally on industry standard hardware or a virtual machine, such as VMWare.

Chapter 2: Deployment Prerequisites

Study the information in the following section before configuring Operations Analytics (OpsA) collections.

Data Sources used in Operations Analytics

In today's complex data center environments, the source of a problem is not always easy to detect using traditional management and troubleshooting tools that look only for predetermined solutions to known potential problems. For example, many management and troubleshooting tools are designed to provide analytics for a specific problem context, such as root cause isolation, outlier detection, and service level agreement violation. They provide these services by using a specific data set and analytics technique.

With Operations Analytics (OpsA) you generate insights from the IT data in your environment that you, the OpsA administrator, chooses to collect in your network. And because identifying the most useful analytics to derive from the data generally depends on the problem context, the user community provides each data request.

As the OpsA administrator, you configure collections from a diverse set of possible sources. For example, if you have HP Network Node Manager (NNMi) or HP Operations Manager (HPOM), you can configure collections to gather NNMi topology or HPOM events occurring within your network.

See "Table 2: Predefined Data Collection Sources by Collection Type""Data Sources used in Operations Analytics "and "Table 3: Custom Data Collection Sources by Collection Type" for the list of supported data sources.

Note: OpsA requires that you use a configuration template to configure each collection. See "Table 2: Predefined Data Collection Sources by Collection Type" to determine the data sources that have predefined configuration templates. You create Custom Collections for any supported data source that does not have a configuration template provided by OpsA.

OpsA provides predefined collection templates for the data sources shown in the following table:

Table 2: Predefined Data Collection Sources by Collection Type

Predefined Data Collection Sources	Metrics Collection Type	Events Collection Type	Topology Collection Type	Inventory Collection Type
HP BSM RTSM (Configuration Item Inventory)	no	no	no	yes
HP Business Process Monitor (BPM)	yes	no	no	no
HP NNMi Custom Poller	yes	no	no	no

Table 2: Predefined Data Collection Sources by Collection Type, continued

Predefined Data Collection Sources	Metrics Collection Type	Events Collection Type	Topology Collection Type	Inventory Collection Type
HP Network Node Manager iSPI Performance for Metrics Component Health	yes	no	no	no
HP Network Node Manager iSPI Performance for Metrics Interface Health	yes	no	no	no
HP Operations Agent	yes	no	no	no
HP Operations Smart Plug-in for Oracle	yes	no	no	no
HP OMi (Operations Manager i) Events	no	yes	no	no
HP Operations Manager (OM) Events	no	yes	no	no

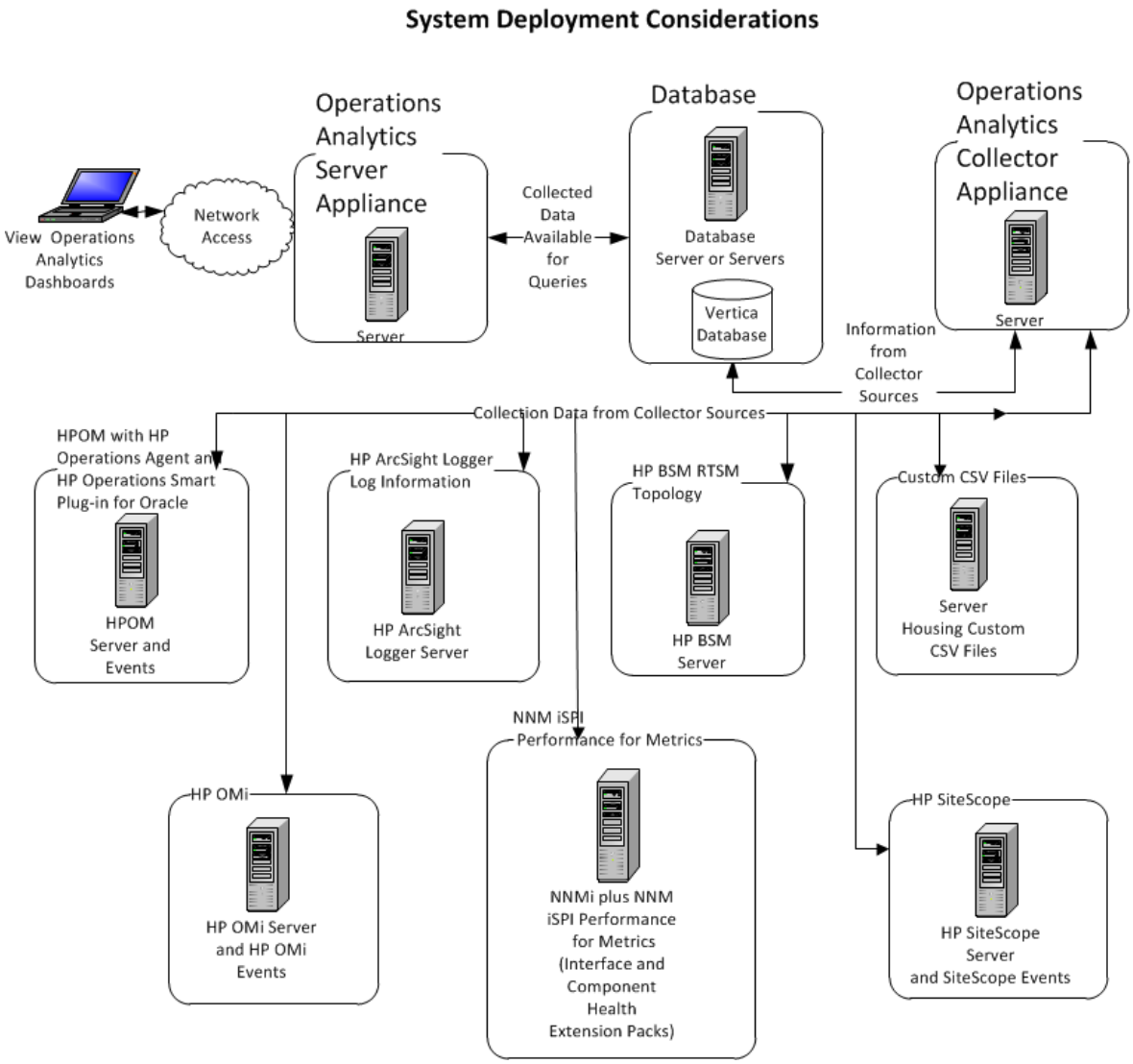
OpsA supports, but does not provide predefined collection templates for the data sources shown in the following table:

Table 3: Custom Data Collection Sources by Collection Type

Custom Collection Data Source	Topology Collection Type	Metrics Collection Type	Structured Logs Collection Type	Undefined Collection Type
HP SiteScope	no	yes	no	no
Structured Logs	no	no	yes	no
Custom CSV Files	no	no	no	yes

Supported Deployments

Review the information shown in the following diagram to begin understanding the data sources used by Operations Analytics (OpsA) and how they are configured together to better plan your OpsA installation.



OpsA supports the deployments described in this chapter.

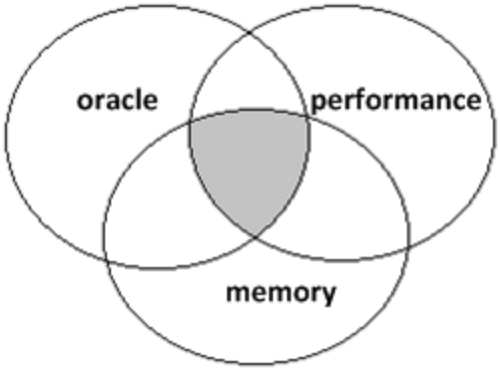
How Tags are Used in Operations Analytics

A tag is a word or phrase that is associated with a metric, topology, event, or log file attribute that is stored as part of a collection in Operations Analytics (OpsA). Tags are provided by OpsA.

Tags are used in the OpsA Phrased Query Language or Analytics Query Language to create an OpsA dashboard. They help to define the following:

- Entity class names for which you want information, such as a database collection tagged with **database** or a metric collection with transaction rates tagged with **transaction** and **performance**.
- Hardware and software components, such as **cpu**, **memory**, **disk**, **interface**, **tablespace**, **process**, and **threads**.
- Metrics or problem areas, such as **utilization**, **availability**, **performance**, and **change**.

OpsA creates an intersection of the tags used to query for a Guided Troubleshooting Dashboard. For example, the query **oracle memory performance** returns only the metrics that are associated with all three tags (**oracle memory performance**) as represented in the following diagram:



The purpose of tagging is to find a suitable set of metrics and logs that relate to a specific question a user might pursue using OpsA query languages. The OpsA administrator defines tags before configuring an OpsA collection or after configuring a collection using the `$OPSA_HOME/bin/opsa-tag-manager.sh` script. See "Creating, Applying, and Maintaining Tags" on page 116 for more information.

Use the following guidelines when configuring tags:

- Collections should only be assigned tags that describe the purpose of the whole domain.
- Typically these tags are the entity class that is associated with a domain.
- Examples of these tags are as follows:
  - Oracle
  - performance
  - memory
- Always use a tag for the most specific entity type that describes the whole collection within which you want to search.

If you add tags to a metric in a collection, tag it as follows:

- Use entity elements that describe the metric or text variable (such as tablespace, process, cpu, or disk).
- Use tags that depict the purpose of the metric (such as performance, availability, error, security, change).
- Use tags that depict a search focus. This tag should only be applied to important metrics that either are prototypical for a set of metrics (cpu) or gives important information about the health of an entity (like error rate for a database).
- Keep the tag length short, but descriptive enough to meet your needs. Tags are limited to 256 characters.

To configure tags when configuring collections, see *"Creating, Applying, and Maintaining Tags" on page 116*.

Chapter 3: Configuring Tenants and Collections

Deploying OpsA is relatively easy and usually takes less than two hours to install and deploy. OpsA depends on other products and solutions as data sources. Some examples are BSM products such as BPM, SiteScope, NNMi, OMi, and Operations Agent. To configure OpsA to connect to and process data from these data sources can take anywhere from a few minutes to a few hours, depending on the data sources. In a general sense, you should be able to deploy OpsA and get started with two or three data sources in less than two days, resulting in having data being collected from these data sources into OpsA for analysis.

A collection defines the data to be collected and corresponds to a database table in which the Operations Analytics (OpsA) Collector Appliance stores the data. To populate the OpsA database with useful information you must configure collections.

Using the Collections Manager to Configure Collections

The recommended method of configuring collections is by using the **Collections Manager** from the OpsA console.

Note: You might not be able to configure all collections by using the **Collections Manager** (see the following examples):

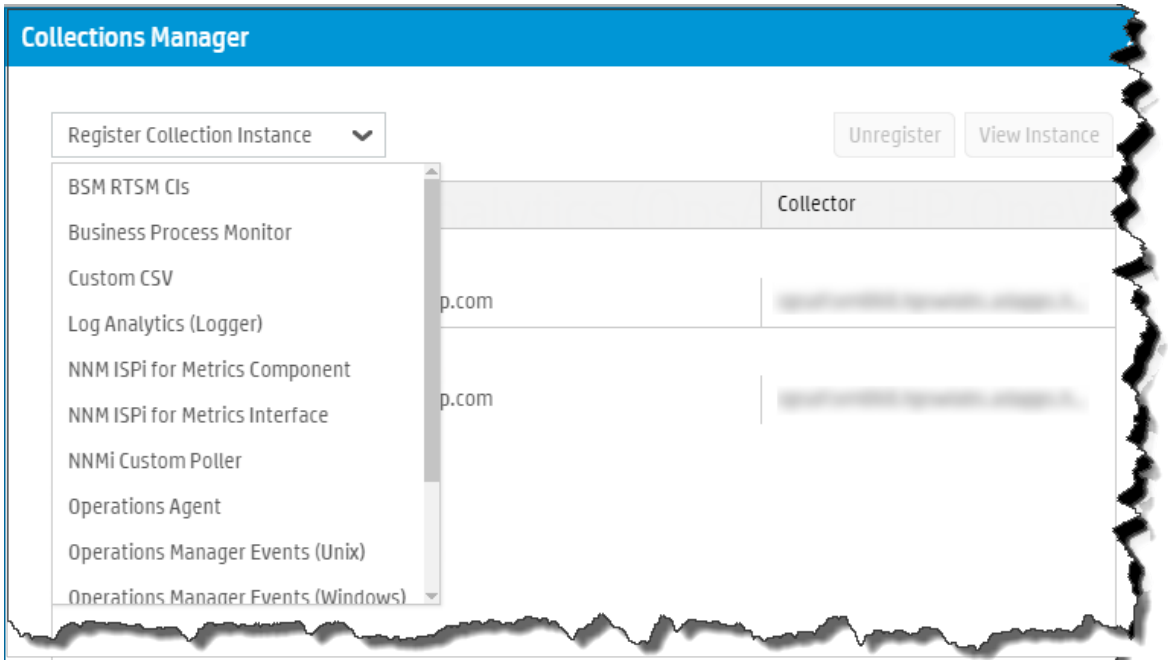
- You cannot use the **Collections Manager** to configure a Structured Log collection.
- Although it would be better to configure a SiteScope collection by using the **Collections Manager**, you cannot use the **Collections Manager** to configure custom monitors in a SiteScope collection.

If the **Collections Manager** does not provide you with all of the options you need for a specific collection, then use the more manual collection configuration instructions (using use the `opsa-collection-setup.sh` script) shown in "Configuring Collections using Predefined Templates " on page 25 or "Configuring Collections for Custom Data Sources " on page 87.

Note: You must register a collection using the instructions shown in "Registering Each Collector Appliance" on page 24 before using the **Collections Manage**r.

Note: You must decide on the approach you want to take regarding tenants before creating your collections. See the instructions shown in "Creating Tenants " on page 20 before using the **Collections Manager**.

See an example of the **Collections Manager** shown below:



Do the following to access the **Collections Manager**:

1. Log on to the OpsA Console with the tenant administrator credentials for your tenant. This would be the **opsatenantadmin** user when using the default tenant.
2. Click the **Settings** menu in the upper right.
3. Click **Collections Manager** to open the Collections Manager.
4. Click **Add Collection Instance** to pull down a list of potential collections and select the collection you want to configure.

The **Collections Manager** supports the following collection configurations:

1. BSM RTSM NNMi Configuration Item (CI) Collection
2. Business Process Monitor Collection
3. Custom CSV Collection
4. Log Analytics (Logger) Collection
5. NNM iSPi Performance for Metrics Component Collection
6. NNM iSPi Performance for Metrics Interface Collection
7. NNMi Custom Poller Collection

8. Operations Agent Collection
9. Operations Manager Events (Unix) Collection
10. Operations Manager Events (Windows) Collection
11. Operations Manager i Events Collection
12. HP Operations Smart Plug-in for Oracle Collection
13. SiteScope Collection
14. Log Analytics (Splunk) Collection

The **Collections Manager** provides you with an **Unregister** button if you no longer want to analyze data for a collection. See *Collections Manager* in the *OpsA Help* for more information.

Using a Manual Approach to Configure Collections

If you prefer to use a more manual method of configuring collections, use the information in this section. The instructions shown in this section cover the following:

1. Registering your collector appliances.
2. Identifying the collections for which you want to collect data.
3. For any of the custom collections you must create a collection template that describes what is being collected.
4. Specifying what nodes to collect against in a node properties file.
5. Creating a collector for each of the collections and assigning those collectors to a specific collector appliance.

See "Terminology Used in this Document" on page 10 and "Supported Deployments" on page 14 for more information about data sources from which OpsA can collect data.

You can also configure collections through command line interaction using the `opsa-collection-config.sh` script. OpsA stores configurations on its file system and not in the OpsA database. If you have multiple OpsA Server Appliances installed, you must designate one of them from which to complete collection configuration. You should back up these files as part of your company's server backup strategy. All collection configuration files are stored in the `/opt/HP/opsa/conf/collection` directory.

For best results, configure your collections in the following order (from less complex to more complex collection configurations):

1. HPOM Events Collection
2. OMi Events Collection

3. HP Operations Agent Collection
4. HP Operations Smart Plug-in for Oracle Collection
5. HP Business Process Monitor Collection
6. NNMi Custom Poller Collection
7. NNM iSPi Performance for Metrics Component Health Collection
8. NNM iSPi Performance for Metrics Interface Health Collection
9. HP BSM RTSM NNMi Configuration Item Collection
10. ArcSight Logger Collection
11. Custom CSV Collection
12. Custom SiteScope Collection
13. Structured Log Collection

Preparing for Configuring Collections

Operations Analytics (OpsA) relies on collected metrics, topology, event, and log file data from a diverse set of possible data sources. An OpsA Collector Appliance contains the software that listens for data coming from a device. Each server that is running the OpsA Collector software must be registered as a Collector Appliance. Customers can register a collector, then use that registered collector for multiple collections.

During the process of collector configurations, you must complete the steps in this section before configuring either custom or predefined collections. After you complete the steps in this section continue to "Configuring Collections using Predefined Templates " on page 25 and "Configuring Collections for Custom Data Sources " on page 87 to finish configuring your collections.

Registering a collector appliance is a two-step process:

1. "Creating Tenants " below
2. "Registering Each Collector Appliance" on page 24

Creating Tenants

Operations Analytics (OpsA) gathers metrics, topology, event, and log file data from a diverse set of possible data sources. Tenants enable you to separate information from these data sources into groups, such as collections, user accounts, and user groups.

For each collection you define for the data sources supported by OpsA, you must define a corresponding Tenant Admin or use the default Tenant, opsa_default, and the associated Tenant Admin user, opsatenantadmin, if you choose to not use a tenant model to separate information.

Important Tenant Information

A collection is automatically associated with a tenant depending on the Tenant Admin user that the OpsA administrator provides as input when running the `$OPSA_HOME/bin/opsa-collection-config.sh` script.

Before creating collections using the **Collections Manager** or the `$OPSA_HOME/bin/opsa-collection-config.sh` script, **you must decide on one of the following options** before proceeding with any collection configuration:

- Use the default Tenant, opsa_default, its corresponding default tenant username (opsatenantadmin), and the password for this user that you selected during installation. If you choose this option. skip directly to "Registering Each Collector Appliance" on page 24.
- Decide on which existing tenant to use.
- Create a new tenant and its corresponding Tenant Admin.

Note: Any user that is associated with a new tenant created by a member of the Super Admin user group cannot see collected information (in any dashboard) from any of the existing predefined collections (for any of the existing tenants, including the opsa_default tenant). After a member of the Super Admin user group creates a new tenant, the tenant admin user associated with that tenant needs to create collections for this new tenant.

When using the `$OPSA_HOME/bin/opsa-collection-config.sh` script, some examples in this document use a predefined Tenant Admin user, opsatenantadmin, for the predefined opsa_default tenant. When defining collections, replace the opsatenantadmin shown in the example with the Tenant Admin user for the collection you are creating.

Note: You can configure a collector to collect data from a data source for only one tenant. So a single collector cannot be used to collect data from a single data source for multiple tenants.

Note: There might be tenant limitations when configuring collections for products that support multiple tenants. Each collector you configure for a collection supports a single tenant, so the data source from which it is collecting must also be for a single tenant.

OpsA provides the following predefined User Groups:

- **Super Admin**: During installation, the opsaadmin user gets created, and assigned to the Super Admin user group. You set the password for this user during installation. The primary responsibility

of users assigned to the Super Admin user group is to add, modify, and delete tenants and users assigned to the Tenant Admin user group. See the *opsa-tenant-manager.sh* reference page (or the Linux manpage) for information about creating and managing tenants.

- **Tenant Admin**: During installation, the opsatenantadmin user gets created, and assigned to the Tenant Admin user group. You set the password for this user during installation.Only a user assigned to the Super admin user group is permitted to create a user assigned to the Tenant Admin user group. The primary responsibility of the Tenant Admin user is to add, modify, and delete users for a specific tenant. See the *opsa-tenant-manager.sh* reference page (or the Linux manpage) for information about creating and managing users for a tenant.

- **User**: During installation, the opsa user gets created, and assigned a normal user role. You set the password for this user during installation. Only a user assigned to the Tenant admin user group is permitted to create a user having a normal user role.This role is for the normal user who can use the Operations Analytics console and has access to data for the user group to which it is assigned. This user account must be unique across all tenants. See *Manage Users* in the *Operations Analytics Help* for more information.

If you plan to use a tenant model, you can create additional tenants from the Operations Analytics console or by using the `$OPSA_HOME/bin/opsa-tenant-manager.sh` script. See *Add a Tenant* in the *Operations Analytics Help* for more information about creating a tenant using the Operations Analytics console. To create a tenant and a Tenant admin user for a collection by using the `opsa-tenant-manager.sh` script, do the following:

1. Run the `$OPSA_HOME/bin/opsa-tenant-manager.sh` command from the Operations Analytics Server Appliance as a user assigned to the Super Admin User Group. See the *opsa-tenant-manager.sh* reference page (or the Linux manpage) for information about managing tenants.

2. Enter **Add a new tenant** and follow the interactive commands to add the new tenant.

3. Run the `$OPSA_HOME/bin/opsa-tenant-manager.sh` command as a user assigned to the Super Admin User Group.

4. Enter **Add a new user** and follow the interactive commands to add a user assigned to the Tenant Admin user group for the newly created tenant.
   See the *opsa-tenant-manager.sh* reference page (or the Linux manpage) or *Manage Users* in the *Operations Analytics Help* for information about managing users.

See the *opsa-tenant-manager.sh* reference page (or the Linux manpage) or *Manage Users* in the *Operations Analytics Help* for information about managing users.

If you do not create a Tenant Admin user while adding a new tenant (as shown above in steps 3 and 4), add the Tenant Admin user for the new tenant later using the `$OPSA_HOME/bin/opsa-user-manager.sh` script. Do the following:

1. Run the `$OPSA_HOME/bin/opsa-user-manager.sh` command.

2. Enter **Add a new user** option.

3. Enter the Super Admin username and password.
4. Enter the Tenant Name for which you must add the Tenant Admin user.
5. Enter the new Tenant Admin user name.
6. Enter the new password for the new Tenant Admin user name.
7. Confirm the password.

The newly added Tenant Admin user is now available to add, modify, and delete users for its specified tenant. See the *opsa-user-manager.sh* reference page (or the Linux manpage) for more information.

Configuring and Managing Logger or Splunk for a Tenant

Use the information in this section to configure and manage Logger or Splunk configurations for a tenant. As mentioned earlier, you can use the default Tenant, opsa_default, and the associated Tenant Admin user, opsatenantadmin, if you choose to not use a tenant model to separate information.

Note: If the log management software is Logger, you can configure more than one Logger for a tenant. If the log management software is Splunk, you can configure only one Splunk for a tenant.

Note: See the Operations Analytics Support Matrix for the supported versions of Splunk and Logger.

Note: When running the following command, use port 443 for Logger and port 8089 for Splunk.

To add an HP ArcSight Logger or Splunk configuration for a tenant, run the following command:
`opsa-logger-config-manager.sh -loginUser <Tenant Admin User> -loginPassword <password> -add -loggerType (arcsight | splunk) -host <hostname> -port <port> -sslEnabled (true | false) -username <user> -password <password>`

To delete an HP ArcSight Logger or Splunk configuration for a tenant, run the following command:
`opsa-logger-config-manager.sh -loginUser <Tenant Admin User> -loginPassword <password> -delete -host <hostname>`

To list existing HP ArcSight Logger or Splunk configurations for a tenant, run the following command:
`opsa-logger-config-manager.sh -loginUser <Tenant Admin User> -loginPassword <password> -list`

To update an existing HP ArcSight Logger or Splunk configuration for a tenant, run the following command: `opsa-logger-config-manager.sh -loginUser <Tenant Admin User> -loginPassword <password> -update -host <hostname> -port <port> -password <password>`

See the *opsa-logger-config-manager.sh* reference page (or the Linux manpage) for more information.

Registering Each Collector Appliance

You must register each Operations Analytics (OpsA) Collector Appliance you plan to use with the OpsA Server Appliance. If you plan to use a tenant model (tenants other than the opsa_default default tenant), you must use the `-tenant` option with the `opsa-collection-config.sh` command. See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

Automatically Creating Alert Collections

All of the alerts generated by OpsA are stored as collections. This collected information is used to display dashboards for these alerts generated over time. Unlike all of the other collections, there is no manual configuration or registration required for the Alerts Collections, as a new Alerts Collection gets created with each newly created OpsA tenant.

See *Alerts* in the *Operations Analytics Help* for more details about the Alerts feature.

An Alerts Collection gets created each time an OpsA Collector is registered to an OpsA Server. When adding new OpsA Servers, as described in "Adding Operations Analytics Server Appliances" on page 167, you must run the following command from the new OpsA Servers to create the alerts collection:

```
$OPSA_HOME/bin/opsa-collection-config.sh -create -collectorhost <collector IP address> -source opsa -domain collection -group alerts -username <tenant admin user> -password <tenant admin password>
```

Checking the Registration Status of a Collector Appliance

To check the registration status of your collector appliance, do the following:

1. `$OPSA_HOME/bin/opsa-collection-config.sh -list -collectorhosts -allversions -username opsatenantadmin`
2. Review the list of registered collectors. If the collector you plan to register is not on the list, you must register it using the instructions in this section.

Registering a Collector Appliance

To register an OpsA Collector Appliance with the OpsA Server Appliance, do the following:

1. Run the following command on the OpsA Collector Appliance to make sure the opsa-collector process is running:

   `$OPSA_HOME//bin/opsa-collector status`

   Look for a message stating the `opsa-collector` process is running. If the message states that the `opsa-collector` process is stopped, restart the process using the following command: `$OPSA_HOME/bin/opsa-collector start`

2. Run the following command from the OpsA Server Appliance:

```
$OPSA_HOME/bin/opsa-collection-config.sh -register -collectorhost
<fully-qualified domain name of the collector appliance host> -port
9443 -username opsatenantadmin -coluser <collector_username> (the
default collector username is opsa) -colpass <collector web service
password> (the default password is opsa)
```

Note: If you have the collector appliance configured to use SSL for data communications, use the `-ssl` option in this command. If you have changed the HTTP user name or password on the OpsA collector appliance, use the `-coluser` and `-colpass` option in this command. You must also use the fully-qualified domain name of the collector appliance host when using this command. See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

Note: The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

Note: The default port to which OpsA listens is 9443. You can modify this port in cases of port conflicts. See "Configuring the HTTP and HTTPS Port for the Operations Analytics Collector Appliance" on page 135 for more information.

If the script communicates successfully with the OpsA Collector Appliance, it registers it in the OpsA Server Appliance database and displays a success message.

Configuring Collections using Predefined Templates

Operations Analytics (OpsA) supports several predefined collection templates. See "Supported Deployments" on page 14 for a list of predefined collection templates.

To configure OpsA to collect data from the supported data sources you plan to use, you must configure collections using a list of predefined collection templates that reside on the OpsA Server Appliance. These predefined collection templates are defined in advance so that administrators only have to configure OpsA collectors to collect data using those predefined collection templates.

You can use the `$OPSA_HOME/bin/opsa-collection-setup.sh` script to assist you when configuring collections that use predefined templates. See the *opsa-collection-setup.sh* reference page (or the Linux manpage) for more information.

Note: Use the `$OPSA_HOME/bin/opsa-collection-setup.sh` script to assist you when first setting up a collection. See the *Operations Analytics Quick Start Guide* for information about how to use the `$OPSA_HOME/bin/opsa-collection-setup.sh` script.

To obtain a copy of the *Operations Analytics Quick Start Guide*, go to:
https://softwaresupport.hp.com/group/softwaresupport/search-result?keyword=

This site requires an HP Passport account. If you do not have one, click the `Create an account` button on the HP Passport Sign in page.

Note: If you want to set up your collections manually or make changes to existing collections, use the collection configuration information in this document. Only use the `$OPSA_HOME/bin/opsa-collection-setup.sh` script to assist you when first setting up a collection.

Important Tenant Information

A collection is automatically associated with a tenant depending on the Tenant Admin user that the OpsA administrator provides as input when running the `$OPSA_HOME/bin/opsa-collection-config.sh` script.

Before creating collections using the **Collections Manager** or the `$OPSA_HOME/bin/opsa-collection-config.sh` script, **you must decide on one of the following options** before proceeding with any collection configuration:

- Use the default Tenant, opsa_default, its corresponding default tenant username (opsatenantadmin), and the password for this user that you selected during installation. If you choose this option. skip directly to "Registering Each Collector Appliance" on page 24.
- Decide on which existing tenant to use.
- Create a new tenant and its corresponding Tenant Admin.

Note: Any user that is associated with a new tenant created by a member of the Super Admin user group cannot see collected information (in any dashboard) from any of the existing predefined collections (for any of the existing tenants, including the opsa_default tenant). After a member of the Super Admin user group creates a new tenant, the tenant admin user associated with that tenant needs to create collections for this new tenant.

When using the `$OPSA_HOME/bin/opsa-collection-config.sh` script, some examples in this document use a predefined Tenant Admin user, opsatenantadmin, for the predefined opsa_default tenant. When defining collections, replace the opsatenantadmin shown in the example with the Tenant Admin user for the collection you are creating.

Configuration Steps

Complete the following configuration steps for the predefined data collection templates that reside on the OpsA Server Appliance:

- "Configuring an HP Operations Smart Plug-in for Oracle Collection" on the next page
- "Configuring an HP Operations Agent Collection" on page 30

- "Configuring an NNM ISPi Performance for Metrics Component Health Collection" on page 35
- "Configuring an NNM ISPi Performance for Metrics Interface Health Collection" on page 40
- "Configuring an HP Operations Manager (OMi) Events Collection" on page 44
- "Configuring an HP Operations Manager (HPOM) Events Collection" on page 49
- "Configuring an HP BSM RTSM Configuration Item (CI) Collection" on page 53
- "Configuring an HP Business Process Monitor Collection" on page 60
- "Configuring ArcSight Logger Out of the Box Smart Connector Collections" on page 67
- "Configuring an NNMi Custom Poller Collection" on page 70

Configuring an HP Operations Smart Plug-in for Oracle Collection

Note: If you do not want to configure this collection by using the **Collections Manager** located in the Operations Analytics console, you can configure this collection using the steps in this section.

After you complete the steps in this section, the HP Operations Smart Plug-in for Oracle Collection collects metrics every 15 minutes, with 5 minute data granularity.

Note: An OpsA Collector Appliance can reliably collect data from approximately 5,000 nodes being monitored by the HP Operations Agent and the HP Operations Smart Plug-in for Oracle.

Gather the following information to prepare for configuring the HP Operations Smart Plug-in for Oracle Collection.

Information to Collect Before Running the opsa-collection-setup.sh Script

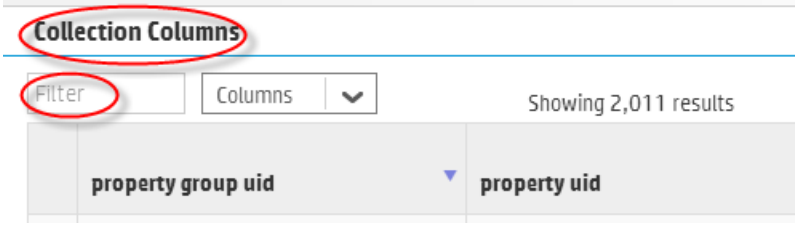
Input Requested by opsa-collection-setup.sh Script	Value
<oanode1.somedomain.com> *<oanode2.somedomain.com>* *<oanode_more.somedomain.com>*	List the fully-qualified domain names of all servers that have the HP Operations Agent and the HP Operations Smart Plug-in for Oracle installed.

Follow the instructions in this section to configure an HP Operations Smart Plug-in for Oracle Collection for Operations Analytics.

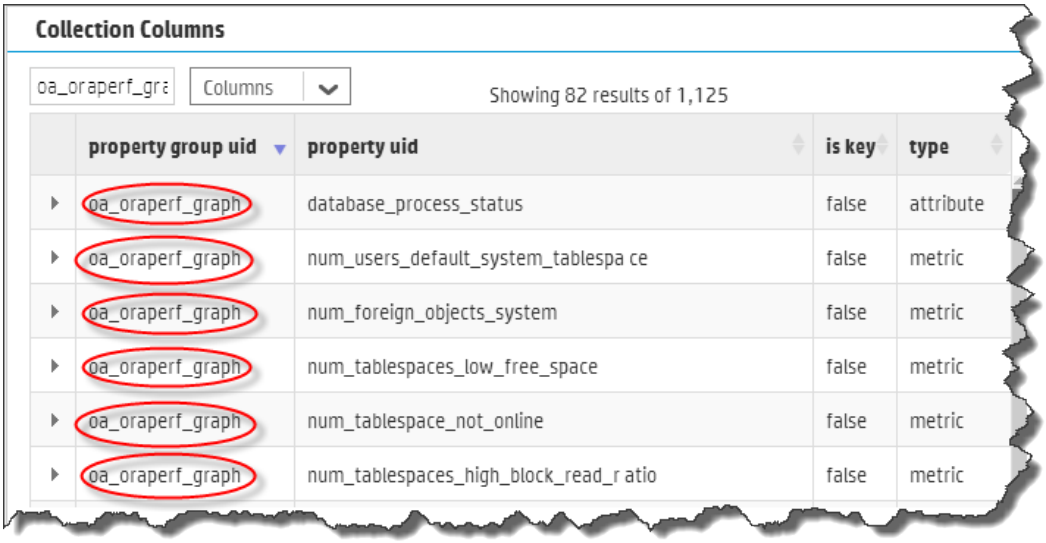
1. Run the interactive `opsa-collection-setup.sh` script.
2. When prompted, enter the tenant admin user name and password.
3. Select the main collector host. In most cases, select `1`.
4. Enter `1` to begin configuring a HP Operations Smart Plug-in for Oracle Collection.
5. Enter the fully-qualified domain names or IP addresses of the servers that have the HP Operations Agent and the HP Operations Smart Plug-in for Oracle installed.
6. Enter `f` or `finish` after you finish entering all of your information.
7. Enter `execute 1` to deploy this collection to the collector host and save this configuration.
8. To check for success, enter `list` and check that the HP Operations Smart Plug-in for Oracle sources were added to the collector host.
9. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

   Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `oa`, a domain of `oraperf`, and a group of `graph` when creating the collection. The resulting property group uid would be `oa_oraperf_graph`.

   a. Type the property group uid (`oa_oraperf_graph`) for this collection in the **Collection ColumnsFilter**:

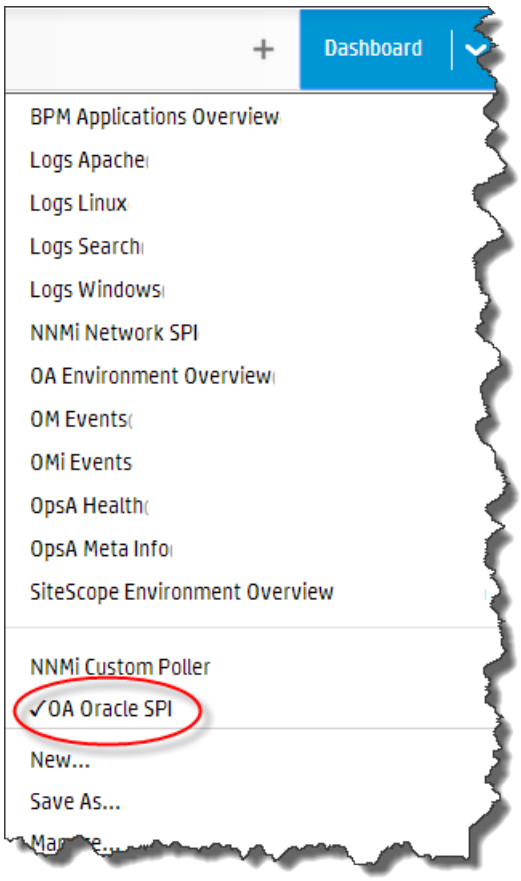


b. After typing property group uid (`oa_oraperf_graph`) for this collection in the **Collection ColumnsFilter**, you should see some information similar to the following for this collection:



10. Create dashboards and query panes for the data you are now collecting. Follow the instructions shown in the *Dashboards and Query Panes* section of the *Operations Analytics Help* for information about creating dashboards and query panes.

11. For this example, assume you created the **OA Oracle SPI** dashboard. From the Operations Analytics console, open the **OA Oracle SPI** dashboard to view some of the collected information for this collection



12. Create AQL functions for the data you are now collecting. See the instructions shown in the *Define Analytic Query Language (AQL) Functions* section of the *Operations Analytics Help* for information about creating AQL functions.For example, using the property group information shown in the **OpsA Meta Info** dashboard, you might create AQL functions similar to the example shown below.

13. If you want to add tags to an HP Operations Smart Plug-in for Oracle Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Configuring an HP Operations Agent Collection

Note: If you do not want to configure this collection by using the **Collections Manager** located in the Operations Analytics console, you can configure this collection using the steps in this section.

The HP Operations Agent Collection collects global system information on the host that is running the HP Operations Agent. After you complete the steps in this section, the HP Operations Agent Collection collects raw metrics every 15 minutes, with 5 minute data granularity.

Note: An OpsA Appliance can reliably collect data from approximately 5,000 nodes being monitored by the HP Operations Agent.

Gather the following information to prepare for configuring the HP Operations Agent Collection.

Information to Collect Before Running the opsa-collection-setup.sh Script

Input Requested by opsa-collection-setup.sh Script	Value
oanode1.somedomain.com	List the fully-qualified domain names of all servers that have the HP Operations Agent and the HP Operations Smart Plug-in for Oracle installed.
oanode2.somedomain.com	List the fully-qualified domain names of all servers that have the HP Operations Agent and the HP Operations Smart Plug-in for Oracle installed.
oanode_more.somedomain.com	Add more servers that have the HP Operations Agent and the HP Operations Smart Plug-in for Oracle installed.

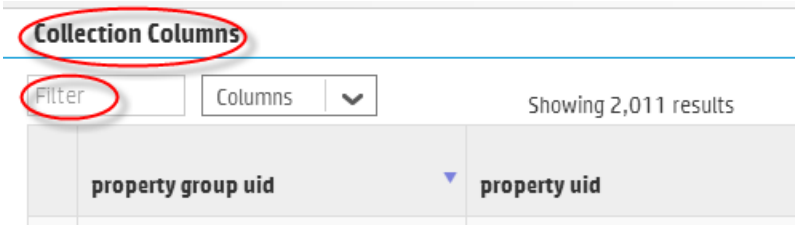
Follow the instructions in this section to configure an HP Operations Agent Collection for Operations Analytics.

1. Run the interactive `opsa-collection-setup.sh` script.
2. When prompted, enter the tenant admin user name and password.
3. Enter `2` to begin configuring an Operations Agent Collection.
4. Follow the prompts to enter the requested information.
5. Enter `f` or `finish` after you finish entering all of your information.
6. Enter `execute 2` to deploy this collection to the collector host.
7. To check for success, enter `list` and check that the HP Operations Agent sources were added to the collector host.
8. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

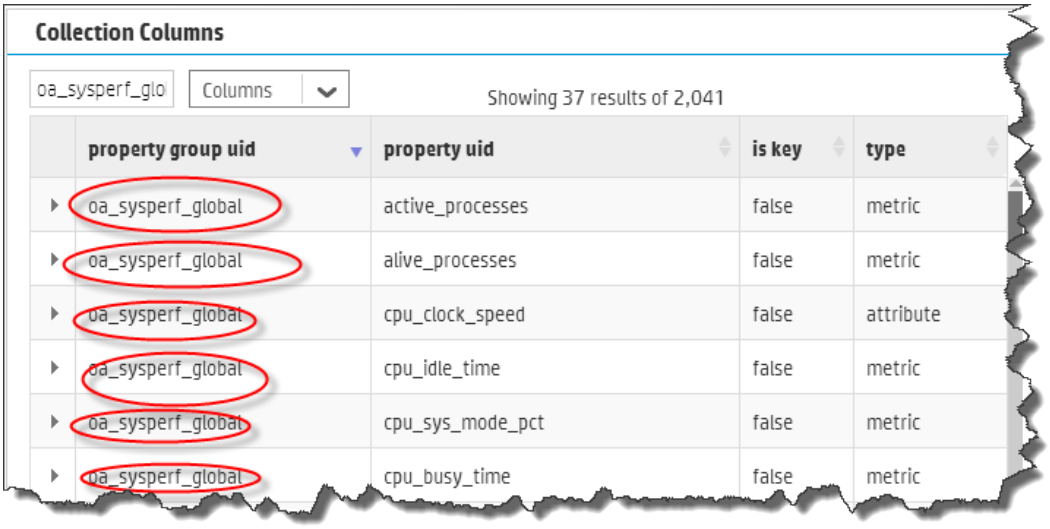
   Note: The property group uid consists of a combination of the `source`, `domain`, and `group`

parameters you used to create the collection. For this collection, you used a name of `oa`, a domain of `sysperf`, and a group of `global` when creating the collection. The resulting property group uid would be `oa_sysperf_global`.

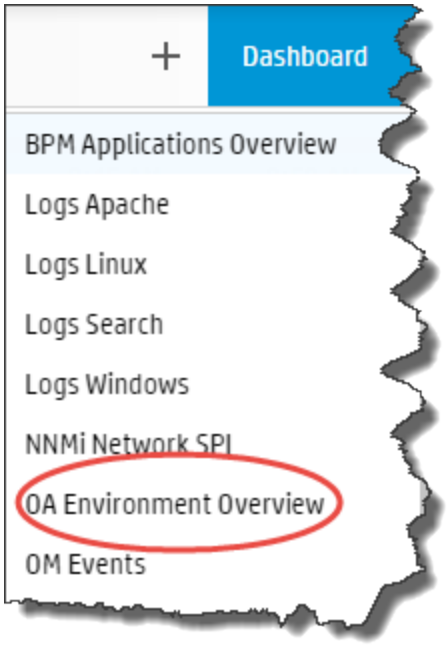
a. Type the property group uid (`oa_sysperf_global`) for this collection in the **Collection ColumnsFilter:**



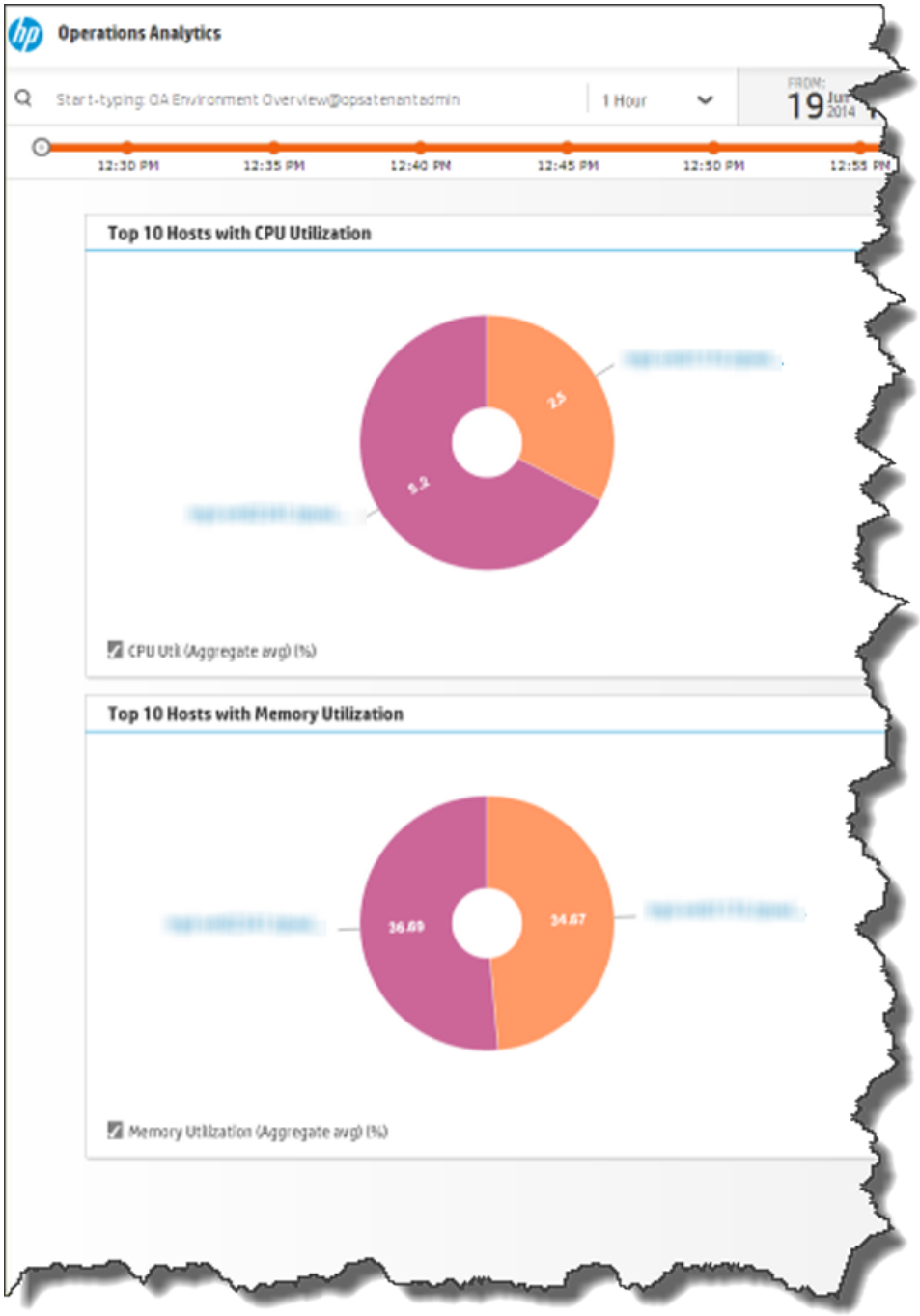
b. After typing property group uid (`oa_sysperf_global`) for this collection in the **Collection Columns Filter**, you should see information for this collection.



9. From the Operations Analytics console, open the **OA Environment Overview** dashboard to view some of the collected information for this collection:



The following is a small sample of HP Operations Agent Collection data provided by the **OA Environment Overview** dashboard.



10. If you want to add tags to an HP Operations Agent Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-*

manager.sh reference page (or the Linux manpage) for more information.

Configuring an NNM ISPi Performance for Metrics Component Health Collection

Note: If you do not want to configure this collection by using the **Collections Manager** located in the Operations Analytics console, you can configure this collection using the steps in this section.

After you complete the steps in this section, the NNMi ISPi Performance for Metrics Component Health Collection reads data from the CSV files within 60 seconds of the file being placed in the source directory.

1. The `opsa` user on the OpsA Collector Appliance must have read and write access to the component health metric files in the OpsA Collector Appliance to move them to the processed directory. The default process directory is `/opt/HP/opsa/data/netcomponent_processed`.

   For example, to configure read and write access to the component health metric files to the OpsA Collector Appliance when the files are located on a Windows server, do the following:

   a. On a Windows server, navigate to **Computer Management** > **System Tools** > **Shares** > **Shared Folders**.

   b. Right-click beneath shares and open the new share wizard.

   c. Create shares for the directories in which the .csv files are stored.

   d. From the OpsA Collector Appliance, add the correct entries to the `/etc/fstab` file. Use the following entries as a model, replacing the IP addresses shown with the IP addresses for the NNMi management server:

   ```
   //10.17.18.19/final /opt/HP/opsa/data/nnm cifs
   username=administrator,password=password,uid=opsa,rw 0 0
   //10.15.14.13/componentfinal /opt/HP/opsa/data/netcomponent cifs
   username=admin,password=passwd,uid=opsa,rw 0 0
   //10.15.14.13/interfacefinal /opt/HP/opsa/data/netinterface cifs
   username=admin,password=passwd,uid=opsa,rw 0 0
   ```

   e. Use the mount -a command to get the directories mounted.

2. For the Collector Appliance to access raw metric information from the NNM iSPI Performance for Metric's component health extension pack, you must export these metrics to CSV files. Run the following command on the NNM iSPI Performance for Metric server to export these metrics to CSV files in the `/csvexports` directory:

- *Windows (Raw Information)*:
  `<Install_Dir>\NNMPerformanceSPI\bin\configureCsvExport.ovpl -p Component_Health -a "Raw,<Target-Dir>"`
- *UNIX: (Raw Information)*:
  `/opt/OV/NNMPerformanceSPI/bin/configureCsvExport.ovpl -p Component_Health -a "Raw,<Target-Dir>"`

Note: You must make the exported component health metrics available on the Operations Analytics (OpsA) Collector Appliance in the `/opt/HP/opsa/data/netcomponent` directory.

If you want to use a different directory than `/opt/HP/opsa/data/netcomponent`, do the following:

a. Edit the following collection template:
   `/opt/HP/opsa/conf/collection/server/config.templates/nnmispi/1.0/netcomponent/component/nnmispi_netcomponent_component_collection.xml`.

b. Specify a different directory for the `sourcedir` attribute.

Note: The `opsa` user on the OpsA Collector Appliance must have read and write access to the component health metric files in the OpsA Collector Appliance to move them to the processed directory. The default process directory is `/opt/HP/opsa/data/netcomponent_processed`.

For example, to configure read and write access to the component health metric files to the OpsA Collector Appliance when the files are located on a Windows server, do the following:

a. On a Windows server, navigate to **Computer Management** > **System Tools** > **Shares** > **Shared Folders**.

b. Right-click beneath shares and open the new share wizard.

c. Create shares for the directories in which the .csv files are stored.

d. From the OpsA Collector Appliance, add the correct entries to the `/etc/fstab` file. Use the following entries as a model, replacing the IP addresses shown with the IP addresses for the NNMi management server::

```
//10.17.18.19/final /opt/HP/opsa/data/nnm cifs
username=administrator,password=password,uid=opsa,rw 0 0
//10.15.14.13/componentfinal /opt/HP/opsa/data/netcomponent
cifs username=admin,password=passwd,uid=opsa,rw 0 0
//10.15.14.13/interfacefinal /opt/HP/opsa/data/netinterface
```

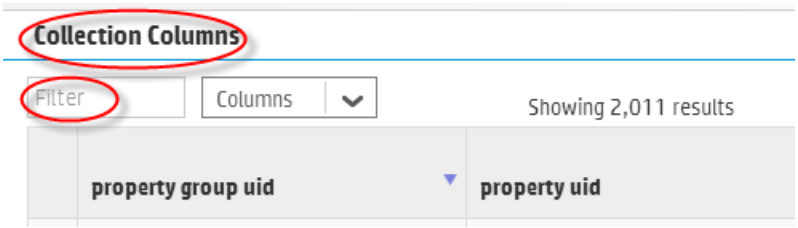
```
cifs username=admin,password=passwd,uid=opsa,rw 0 0
```

e. Use the mount -a command to get the directories mounted.

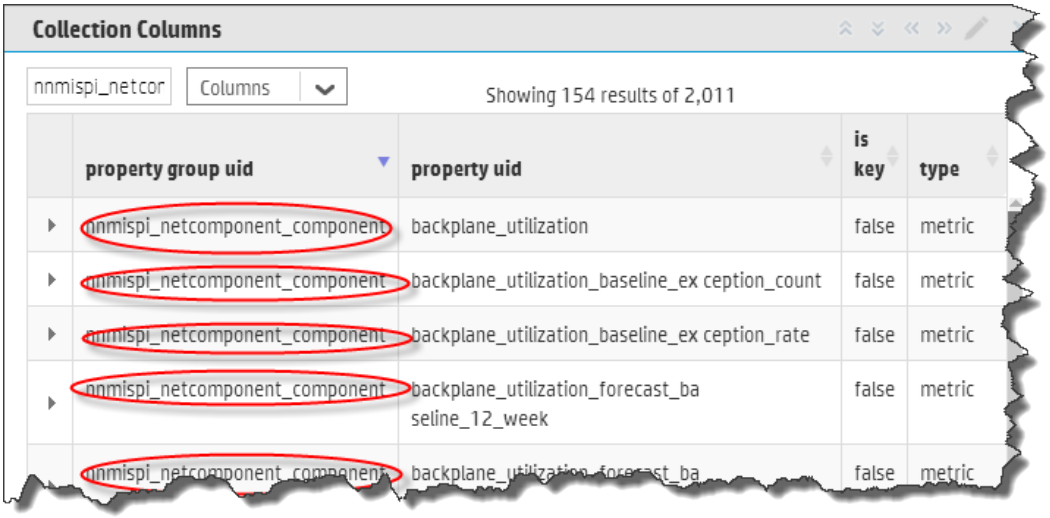
3. Run the interactive `opsa-collection-setup.sh` script.
4. When prompted, enter the tenant admin user name and password.
5. Enter `3` to begin configuring an NNMi Performance SPI Interface and Component Collection.
6. Follow the prompts to enter the requested information.
7. Enter `f` or `finish` after you finish entering all of your information.
8. Enter `execute 3` to deploy this collection to the collector host.
9. To check for success, enter `list` and check that the NNM ISPi Performance for Metrics Component Health Collection sources were added to the collector host.
10. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `nnmispi`, a domain of `netcomponent`, and a group of `component` when creating the collection. The resulting property group uid would be `nnmispi_netcomponent_component`.

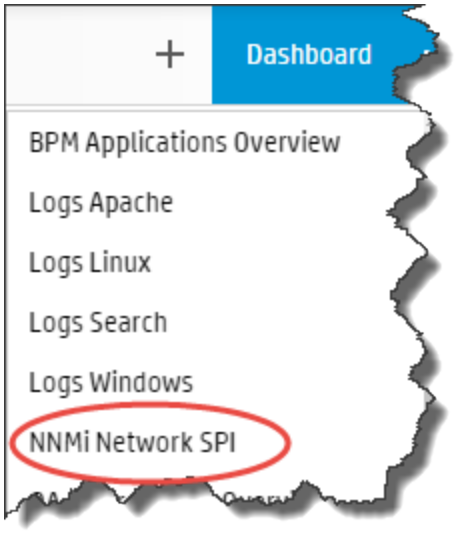
a. Type the property group uid (`nnmispi_netcomponent_component`) for this collection in the **Collection ColumnsFilter:**



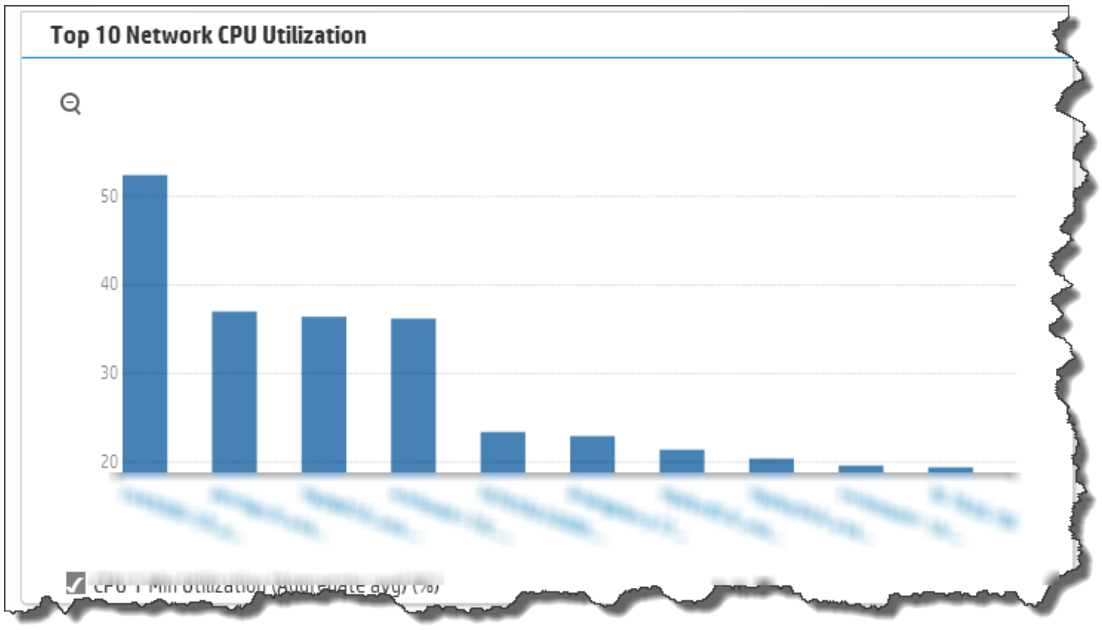
b. After typing property group uid (`nnmispi_netcomponent_component`) for this collection in the **Collection Columns Filter**, you should see information for this collection.



11. From the Operations Analytics console, open the **NNMi Network SPI** dashboard to view some of the collected information for this collection:



The following is a small example of NNM ISPi Performance for Metrics Component Health Collection data provided by the **NNMi Network SPI** dashboard.



12. If you want to add tags to an HP Operations Agent Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Configuring an NNM ISPi Performance for Metrics Interface Health Collection

Note: If you do not want to configure this collection by using the **Collections Manager** located in the Operations Analytics console, you can configure this collection using the steps in this section.

After you complete the steps in this section, the NNMi ISPi Performance for Metrics Interface Health Collection reads data from the CSV files within 60 seconds of the file being placed in the source directory.

1. For the Collector Appliance to access live metric information from the NNM iSPI Performance for Metric's interface health extension pack, you must export these metrics to CSV files. Run the following command on the NNM iSPI Performance for Metric server to export these metrics to CSV files in the `/csvexports` directory:

   - *Windows (Raw Information)*:
     `<Install_Dir>\NNMPerformanceSPI\bin\configureCsvExport.ovpl -p Interface_Health -a "Raw,<Target_Directory">`

   - *UNIX (Raw Information*):
     `/opt/OV/NNMPerformanceSPI/bin/configureCsvExport.ovpl -p Interface_Health -a "Raw,<Target_Directory">`

   Note: You must make the exported interface health metrics available on the Operations Analytics (OpsA) Collector Appliance in the `/opt/HP/opsa/data/netinterface` directory.
   If you want to use a different directory than `/opt/HP/opsa/data/netinterface`, do the following:

   a. Edit the following collection template:
      `/opt/HP/opsa/conf/collection/server/config.templates/nnmispi/1.0/netinterface/interface/nnmispi_netinterface_interface_collection.xml`.

   b. Specify a different directory for the `sourcedir` attribute.

   Note: The `opsa` user on the OpsA Collector Appliance must have read and write access to the interface health metric files in the OpsA Collector Appliance to move them to the processed directory. The default process directory is `/opt/HP/opsa/data/netinterface_processed`.

For example, to configure read and write access to the interface health metric files to the OpsA Collector Appliance when the files are located on a Windows server, do the following:

a. On a Windows server, navigate to **Computer Management** > **System Tools** > **Shares** > **Shared Folders**.

b. Right-click beneath shares and open the new share wizard.

c. Create shares for the directories in which the .csv files are stored.

d. From the OpsA Collector Appliance, add the correct entries to the `/etc/fstab` file. Use the following entries as a model:

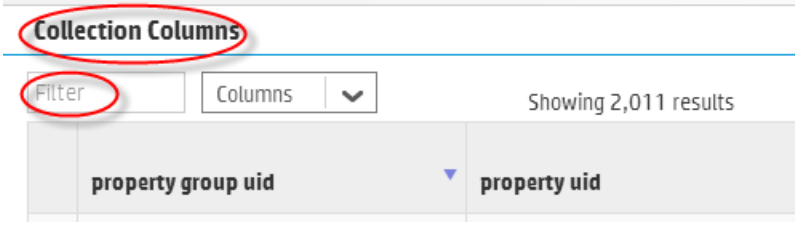
```
//10.17.18.19/final /opt/HP/opsa/data/nnm cifs
username=administrator,password=password,uid=opsa,rw 0 0
//10.15.14.13/componentfinal /opt/HP/opsa/data/netcomponent
cifs username=admin,password=passwd,uid=opsa,rw 0 0
//10.15.14.13/interfacefinal /opt/HP/opsa/data/netinterface
cifs username=admin,password=passwd,uid=opsa,rw 0 0
```

e. Use the mount -a command to get the directories mounted.

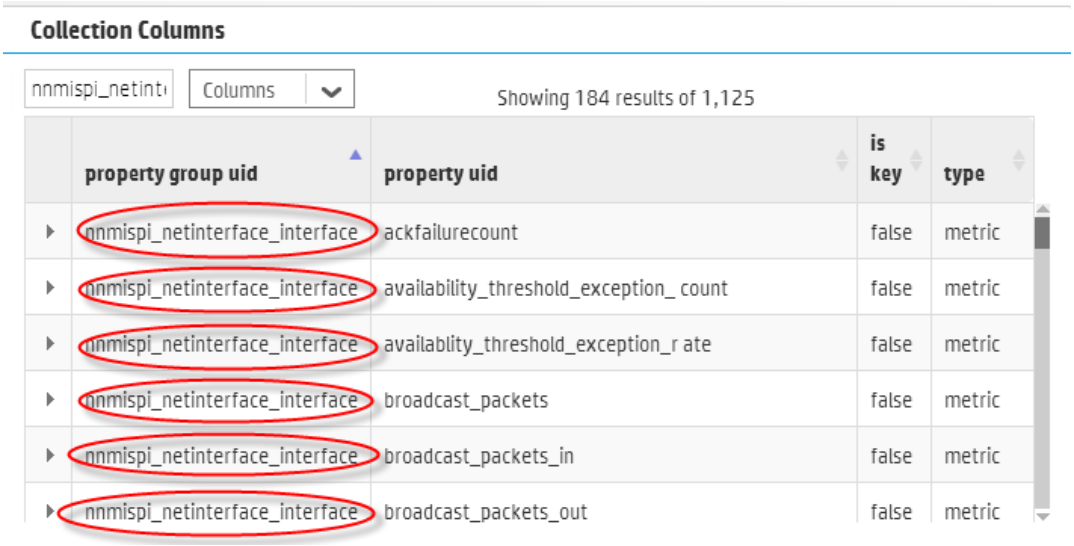
2. Run the interactive `opsa-collection-setup.sh` script.
3. When prompted, enter the tenant admin user name and password.
4. Enter `3` to begin configuring an NNMi Performance SPI Interface and Component Collection.
5. Follow the prompts to enter the requested information.
6. Enter `f` or `finish` after you finish entering all of your information.
7. Enter `execute 3` to deploy this collection to the collector host.
8. To check for success, enter `list` and check that the NNM ISPi Performance for Metrics Interface Health Collection sources were added to the collector host.
9. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `nnmispi`, a domain of `netinterface`, and a group of `interface` when creating the collection. The resulting property group uid would be `nnmispi_netinterface_interface`.

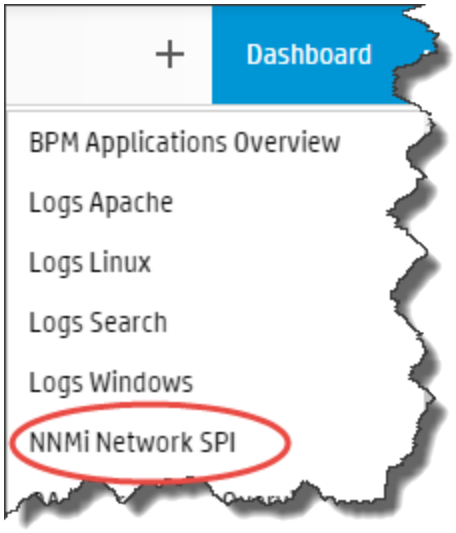
a. Type the property group uid (`nnmispi_netinterface_interface`) for this collection in the **Collection ColumnsFilter:**



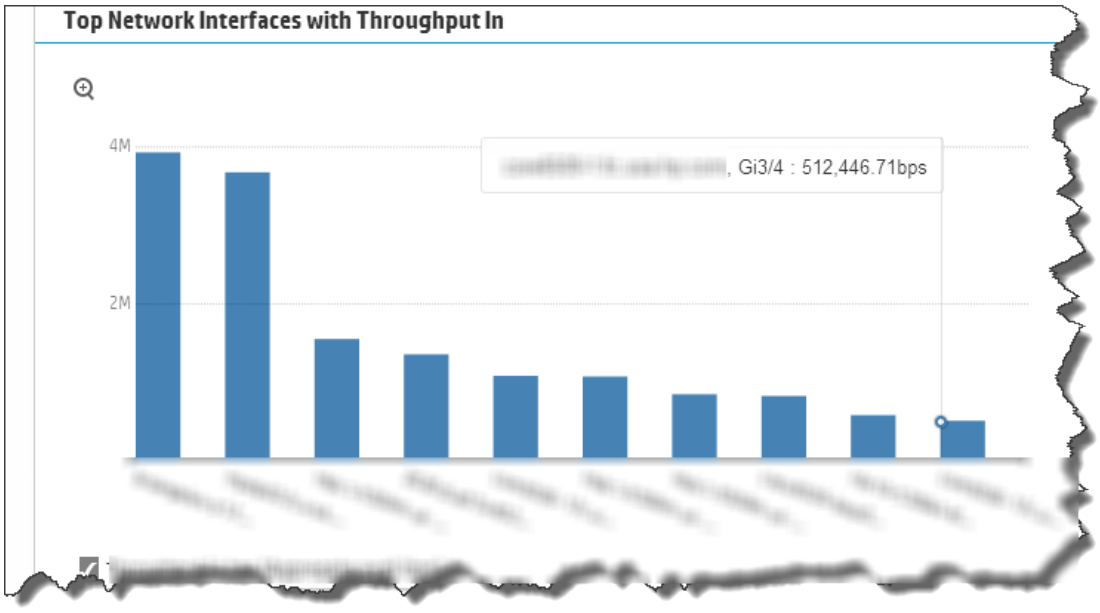
b. After typing property group uid (`nnmispi_netinterface_interface`) for this collection in the **Collection ColumnsFilter**, you should see information in the resulting table:



10. From the Operations Analytics console, open the **NNMi Network SPI** dashboard to view some of the collected information for this collection:



The following is a small example of NNM ISPi Performance for Metrics Interface Health Collection data provided by the **NNMi Network SPI** dashboard.



11. If you want to add tags to an NNM ISPi Performance for Metrics Interface Health Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Configuring an HP Operations Manager (OMi) Events Collection

Note: If you do not want to configure this collection by using the **Collections Manager** located in the Operations Analytics console, you can configure this collection using the steps in this section.

After you complete the steps in this section, the HP OMi Events Collection collects events every 15 minutes, and collects all OMi events that occurred since the last poll.

Note: To support this collection using Oracle RAC, do the following:

1. Copy the `tnsnames.ora` file from the Oracle server to the following locations:

   OpsA Server: `/opt/HP/opsa/conf/collection/tnsnames.ora`

   OpsA Collector: `/opt/HP/BSM/PMDB/config/tnsnames.ora`

2. Rename the `tnsnames.ora` files:

   OpsA Server: `/opt/HP/opsa/conf/collection/bsm-tnsnames.ora`

   OpsA Collector: `/opt/HP/BSM/PMDB/config/bsm-tnsnames.ora`

Note: The OMi collection is also able to accept events from HP Service Health Analyzer (SHA), a component of Business Service Management (BSM). If you have installed SHA on a BSM server and have configured the OMi collection, the OMi collection automatically accepts SHA events. You can then use the collected SHA event data to anticipate and predict IT problems. The following is a short description of SHA:

HP Service Health Analyzer (SHA): HP Service Health Analyzer analyzes abnormal service behavior and alerts IT managers of service degradation before an issue affects their business.

Note: An Operations Analytics (OpsA) Collector Appliance can only collect data for a single HPOM or OMi event source. If you configure more than one HPOM or OMi event source for an OpsA Collector Appliance, it collects the events from only one of the event sources at every collection interval. It cannot be determined from which event source data collection occurs for a given collection interval. To remedy this, configure a separate OpsA Collector appliance for each HPOM or OMi event source.

Setting the Correct Time Zone

If the (OpsA) Collector Appliance is in a different time zone (not in GMT+0), you must specify that time zone in the collection template file before configuring the OMi Events collection.

Do the following if you want to specify a time zone other than GMT

1. Edit the following collection template file:
   `/opt/HP/opsa/conf/collection/server/config.templates/omi/1.0/events/omievents/omi_collection.xml`

   Note: The default template version is 1.0 as shown in this step. Specify the version as appropriate.

2. Specify the OpsA Collector Appliance time zone offset from GMT by by editing the timezone attribute of the following column element names: `timestamp`, `TimeReceivedTimeStamp`, and `TimeCreatedTimeStamp`.

   For example, if the OpsA Collector Appliance is in GMT+3, you must specify a timezone that is 3 hours ahead of GMT. That means the timezone attribute's value needs to be GMT+3 as shown in the following example:

   ```
   <column name="timestamp" position="2" datatype="datetime" length="0"
   datetype="date" format="MM/dd/yyyy HH:mm:ss" timezone="GMT+3 "key="no"
   mapsto="TimeOfStateChangeTimeStamp" value="" label="Timestamp"
   columnname="timestamp" unit="" tags="" type="attribute" />
   ```

3. Save your changes.

After you complete the above steps to specify your time zone in the collection template file, you can use the instructions in the remainder of this section to configure the HP OMi collection.

Configuring the HP OMi Events Collection (Automated Method)

Gather the following information to prepare for configuring the HP OMi Events Collection.

Information to Collect Before Running the opsa-collection-setup.sh Script

Input Requested by opsa-collection-setup.sh Script	Value
OMi Database Host Name	The fully-qualified domain name) for the server housing the OMi database
OMi Database Port Number	The port number to use to connect to the OMi database. **Note:** The default port is 1433, which is suitable for accessing most MS SQL Server installations. Oracle typically uses port 1521.

Information to Collect Before Running the opsa-collection-setup.sh Script, continued

Input Requested by opsa-collection-setup.sh Script	Value
OMi User Name (database user name)	The name of a database user that has READ access to the OMi/Event Management schema in BSM. This is a database user, not a system user. Check with your BSM Administrator or the database administration staff for the proper credentials.
OMi Password (password for database user)	The password for the OMi user name shown above. This is a database password, not a system password.
Database Type	ORACLE or MSSQL
OMi Database Instance name	The instance name of the OMi database. If you are using Oracle RAC, use the service name in this field.
Omi Database Name	The OMi database name

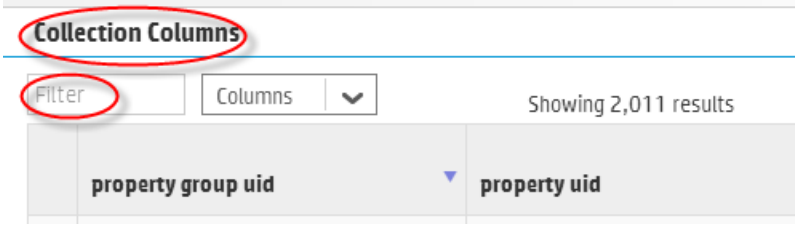
Follow the instructions in this section to configure an HP Operations Manager Events Collection for Operations Analytics.

1. Run the interactive `opsa-collection-setup.sh` script.
2. When prompted, enter the tenant admin user name and password.
3. Enter `4` to begin configuring an HP OMi Events Collection.
4. Follow the prompts to enter the requested information.
5. Enter `f` or `finish` after you finish entering all of your information.
6. After you see a message similar to the following: `Successfully authenticated connection configuration for omi`, enter `execute 4` to deploy this collection to the collector host.
7. To check for success, enter `list` and check that the OMi Source was added to the collector host.
8. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

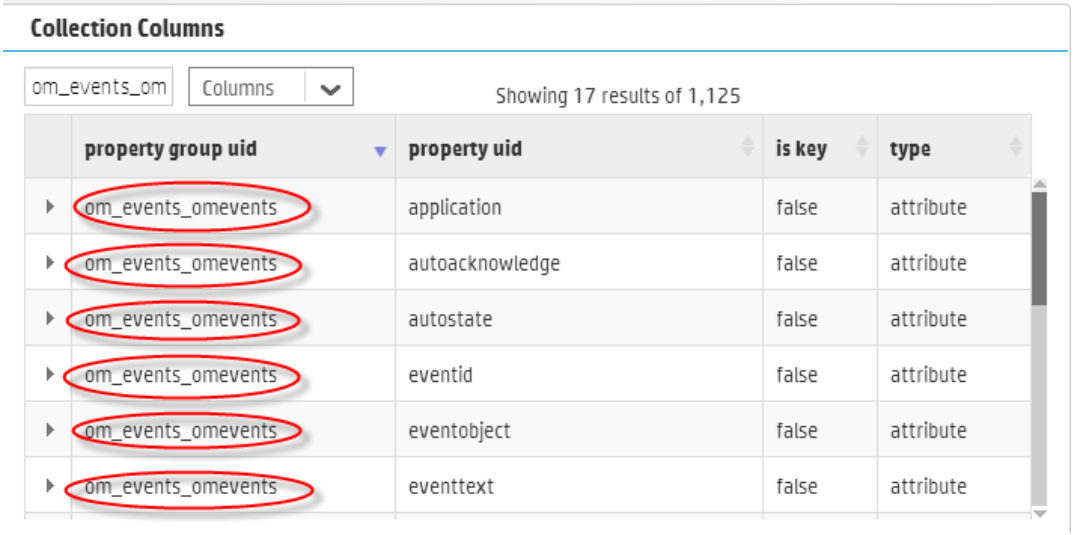
   Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `omi`, a

domain of `events`, and a group of `omievents` when creating the collection. The resulting property group uid would be `omi_events_omievents`.

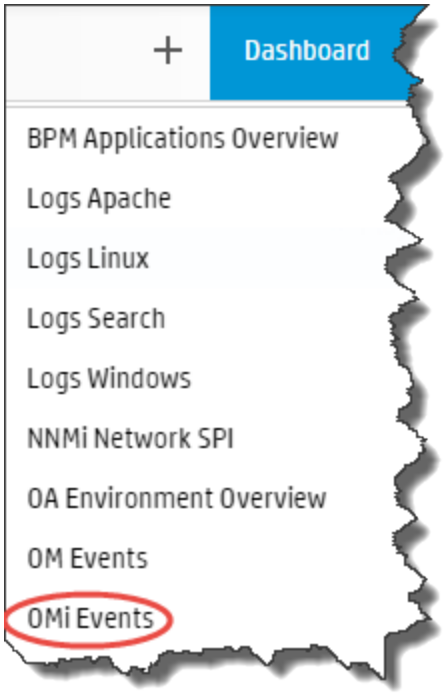
a. Type the property group uid (`omi_events_omevents`) for this collection in the **Collection Columns Filter:**



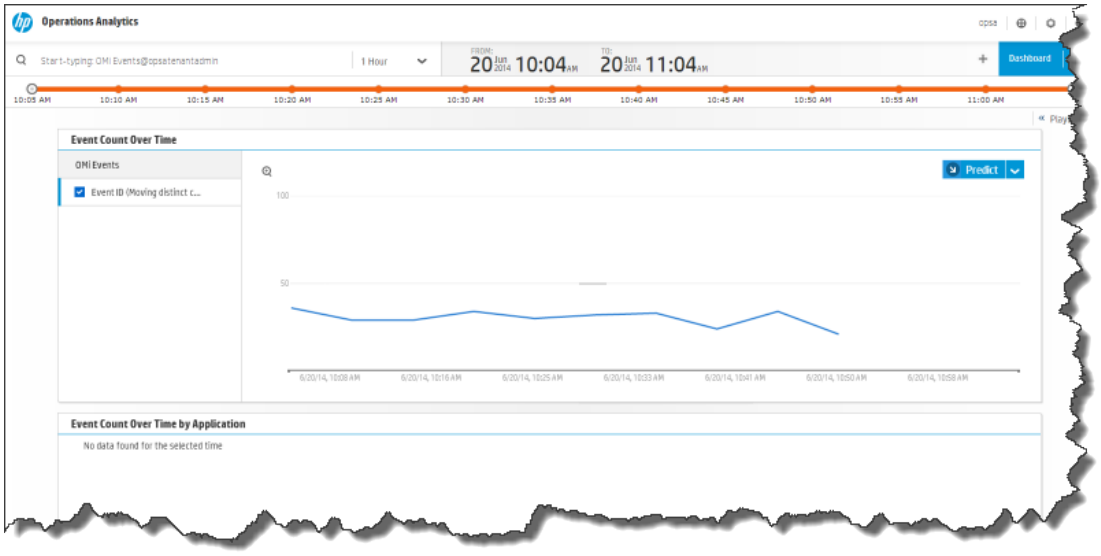
b. After typing property group uid (`omi_events_omievents`) for this collection in the **Collection Columns Filter**, you should see information for this collection.



9. From the Operations Analytics console, open the **OMi Events** dashboard to view some of the collected information for this collection:



The following is a small example of HP Operations Manager Events Collection data provided by the **OMi Events** dashboard.



10. If you want to add tags to an HP Operations Manager Events Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the

opsa-tag-manager.sh reference page (or the Linux manpage) for more information.

Configuring an HP Operations Manager (HPOM) Events Collection

Note: If you do not want to configure this collection by using the **Collections Manager** located in the Operations Analytics console, you can configure this collection using the steps in this section.

After you complete the steps in this section, the HP Operations Manager Events Collection collects events every 15 minutes, and collects all OM events that occurred since the last poll.

Note: To support this collection using Oracle RAC, do the following:

1. Copy the `tnsnames.ora` file from the Oracle server to the following locations:

   OpsA Server: `/opt/HP/opsa/conf/collection/tnsnames.ora`

   OpsA Collector: `/opt/HP/BSM/PMDB/config/tnsnames.ora`

2. Rename the `tnsnames.ora` files:

   OpsA Server: `/opt/HP/opsa/conf/collection/bsm-tnsnames.ora`

   OpsA Collector: `/opt/HP/BSM/PMDB/config/bsm-tnsnames.ora`

For special circumstances related to how Microsoft SQL Server is set up in the HPOM environment, see "Configuring HP Operations Manager (HPOM) (Creating a Database User Account on an HPOM Database Server)" on page 78

Setting the Correct Time Zone

An Operations Analytics Collector Appliance uses the GMT time zone for the HPOM event source. If the HPOM server is in a different time zone, you must specify that time zone in the collection template file before configuring the HPOM collection.

Do the following if you want to specify a time zone other than GMT:

1. Edit the following collection template file:
   `/opt/HP/opsa/conf/collection/server/config.templates/om/1.0/events/om events`

   Note: The default template version is 1.0 as shown in this step. Specify the version as appropriate.

2. Specify the time zone offset from GMT by editing the `timezone` attribute of the following column element names: `timestamp`, `TimeReceivedTimeStamp`, and `TimeCreatedTimeStamp`.

   Specify the total offset between the OpsA Collector Appliance and the OM server (the server from which the OM data comes):

   - If the OpsA Collector Appliance is in GMT+0 (UTC) and the OM server is in GMT+2, the attribute's value needs to be GMT+2.
   - If the OpsA Collector Appliance is in GMT-7 and the OM server is in GMT+0 (UTC), the attribute's value needs to be GMT-7.
   - If the OpsA Collector Appliancer is in GMT-7 and the OM server is in GMT+2, the attribute's value needs to be GMT+9 (the total offset between the servers).

     For this example, set the attribute as follows:

     ```
     <column name="timestamp" position="2" datatype="datetime" length="0"
     datetype="date" format="MM/dd/yyyy HH:mm:ss" timezone="GMT+9" key="no"
     mapsto="TimeOfStateChangeTimeStamp" value="" label="Timestamp"
     columnname="timestamp" unit="" tags="" type="attribute" />
     ```

3. Save your changes.

After you complete the above steps to specify your time zone in the collection template file, you can use the instructions in the remainder of this section to configure the HPOM collection.

Gather the information shown in the following table to prepare for configuring the HP HPOM Events Collection using Operations Manager for Windows:

Information to Gather when using OMW

Field	Value
HPOM Server Name	The fully-qualified domain name of the HPOM server.
HPOM Server Port	`1433`: The port used to connect to the HPOM server.
User Name for the HPOM Server (database user name)	The user name to use for connecting to the HPOM server. The user name to use for connecting to the HPOM database. See "Configuring HP Operations Manager (HPOM) (Creating a Database User Account on an HPOM Database Server)" on page 78 for instructions about creating this user. This is a database user, not a system or HPOM application user.
Database Instance Name	`OVOPS`
Data Source Type	`OM`
Database Type	`MSSQL`
Database Name	`openview`

Gather the information shown in the following table to prepare for configuring the HP HPOM Events Collection using HP Operations Manager for UNIX and Linux:

Information to Gather when using HP Operations Manager for UNIX and Linux

Field	Value
HPOM Server Name	The fully-qualified domain name of the HPOM server.
HPOM Server Port	`1521`: The port used to connect to the HPOM server.
omudbserver.username	`opc_op`: The user name to use for connecting to the HPOM server.
User Name for the HPOM Server (database user name)	The user name to use for connecting to the HPOM server. The user name to use for connecting to the HPOM database. See "Configuring HP Operations Manager (HPOM) (Creating a Database User Account on an HPOM Database Server)" on page 78 for instructions about creating this user. This is a database user, not a system or HPOM application user.
Database Instance Name	`openview` If you are using Oracle RAC, use the service name in this field.
Data Source Type	`OM`
Database Type	`ORACLE`

Follow the instructions in this section to configure an HP Operations Manager Events Collection for Operations Analytics.

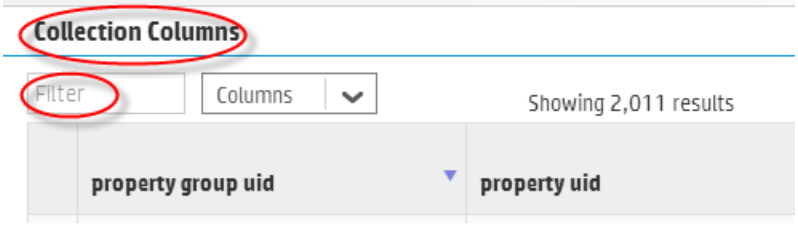
1. Run the interactive `opsa-collection-setup.sh` script.
2. When prompted, enter the tenant admin user name and password.
3. Do one of the following:
   - Enter `5` to begin configuring an HP Operations Manager (Linux, HP-UX, or Solaris) Events Collection.
   - Enter `6` to begin configuring an HP Operations Manager (Windows) Events Collection.
4. Follow the prompts to enter the requested information.
5. Enter `f` or `finish` after you finish entering all of your information.
6. After you see a message similar to the following: `Successfully authenticated connection configuration for om`, enter `execute 5` (for Operations Manager (Linux, HP-UX, or Solaris)) or `execute 6` (for HP Operations Manager (Windows)) to deploy this collection to the collector host.

7. To check for success, enter `list` and check that the OM source was added to the collector host.

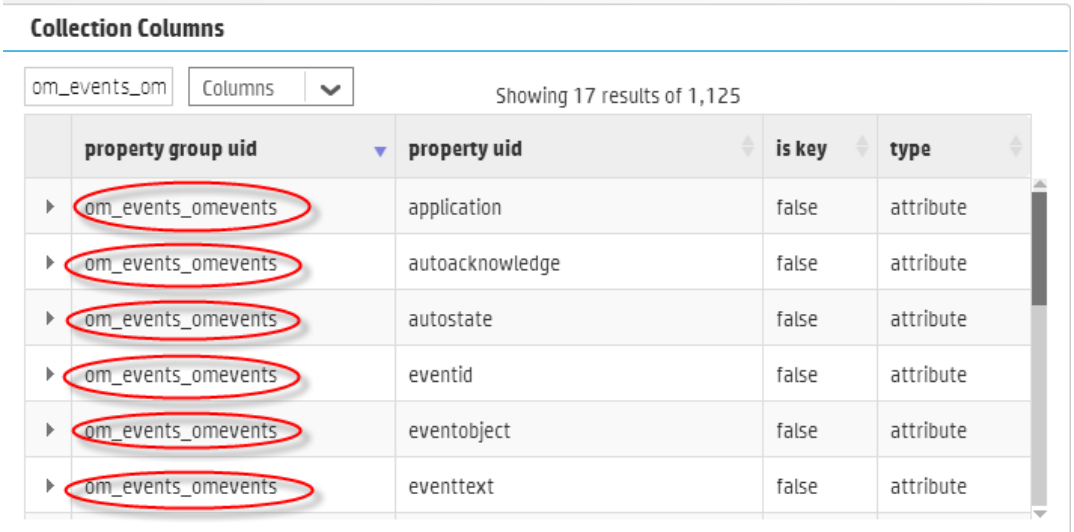
8. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

   Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `om`, a domain of `events`, and a group of `omevents` when creating the collection. The resulting property group uid would be `om_events_omevents`.

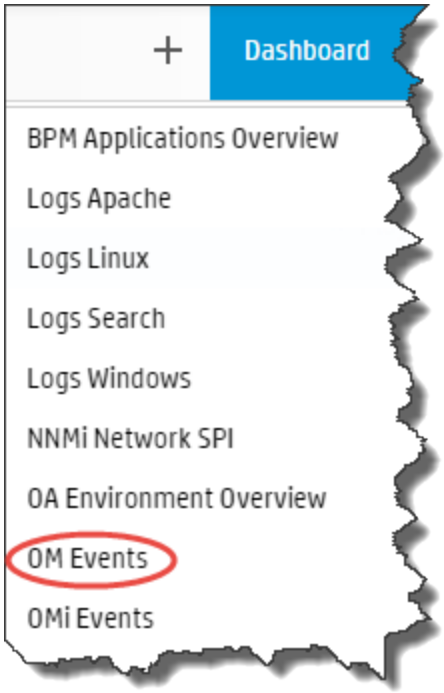
   a. Type the property group uid (`om_events_omevents`) for this collection in the **Collection Columns Filter:**



   b. After typing property group uid (`om_events_omevents`) for this collection in the **Collection Columns Filter**, you should see information for this collection.



9. From the Operations Analytics console, open the **OM Events** dashboard to view some of the collected information for this collection:



10. If you want to add tags to an OM Events Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Configuring an HP BSM RTSM Configuration Item (CI) Collection

Note: If you do not want to configure this collection by using the **Collections Manager** located in the Operations Analytics console, you can configure this collection using the steps in this section.

After you complete the steps in this section, the HP BSM RTSM CI Collection collects data every 6 hours.

Setting the Correct BSM User Name Permissions

When configuring either a BSM RTSM CI collection or a BPM Collection in OpsA you must enter a BSM user name. This BSM user name is used for connecting to the RTSM DPS server, and must be configured for the correct roles.

Note: The user you plan to use for the BPM Collection must be part of the Integration Users Group and include the following OpenAPI required roles:
`CmdbOpenApiQuery`
`CmdbOpenApiClassModel`
`CmdbOpenApiUpdate`
`CmdbOpenApiImpact`

These values could be named as shown below in newer versions:
`RTSMOpenApiQuery`
`RTSMOpenApiClassModel`
`RTSMOpenApiUpdate`
`RTSMOpenApiImpact`

Before competing the remaining configuration steps in this section, do the following to test if the user has the required permissions:

1. Try to log on to BSM as your users using the following URL :
   `http://<BSM>:21212/axis2/services/UcmdbService`

   Note: The log on is successful it you see a web page that shows the following message: `Please enable REST support in WEB-INF/conf/axis2.xml and WEB-INF/web.xml.`

   If your credentials are not accepted you are prompted to enter your user name and password. If this happens, the user does not have the required permissions.

2. If the previous step fails, your user is missing some required permissions. Do not continue until you do the following:

   a. Open the RTSM JMX console using the following URL:
      `http://<BSM>:21212/jmx-console/`

   b. Under the **UCMDB** heading, navigate to **UCMDB:service=Security Services**.

   c. Invoke `setRolesForUser JMX` and give the user either the Admin role or all of the OpenAPI roles:
      Admin role:
      `Admin`

      OpenAPI related roles:
      `CmdbOpenApiQuery, CmdbOpenApiClassModel, CmdbOpenApiUpdate, CmdbOpenApiImpact`

      Note: To prevent making mistakes when entering the role names, retrieve the available

roles by invoking `getAclController JMX` then copy and paste the role names.

3. Repeat step 1 to verify that you correctly made the permissions changes.

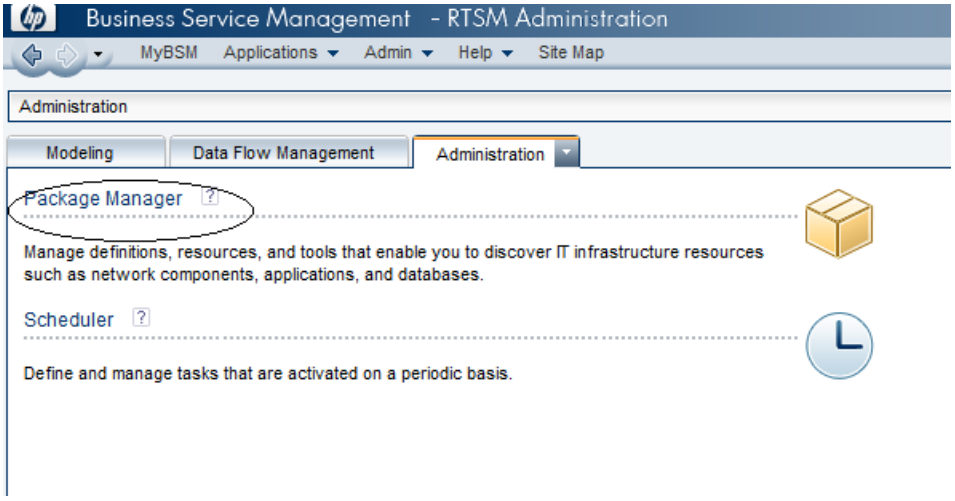
Now the BSM user you tested should have all of the required permissions.

Configuration Steps

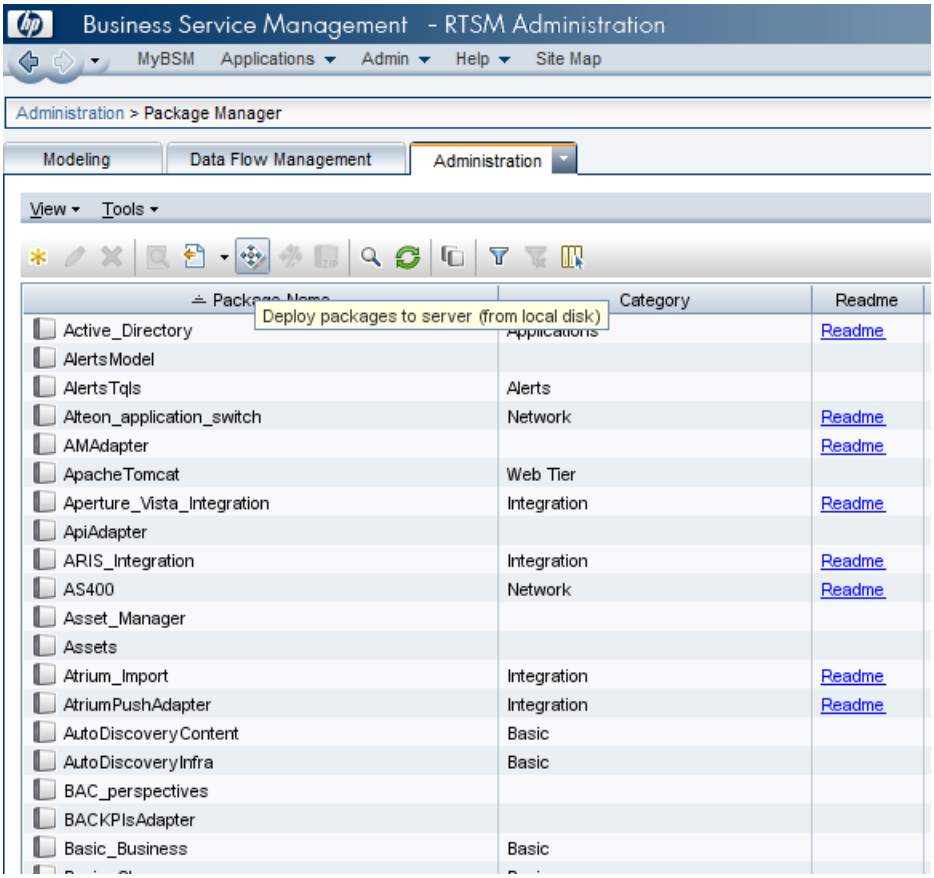
1. Before configuring any of the HP BSM RTSM collections, you must deploy Operations Analytics (OpsA) views on the BSM Server. Do the following:

   a. Copy the `$OPSA_HOME/conf/collection/rtsm_views/OPSA_Views.zip` file to the local server from where the BSM UI is launched.

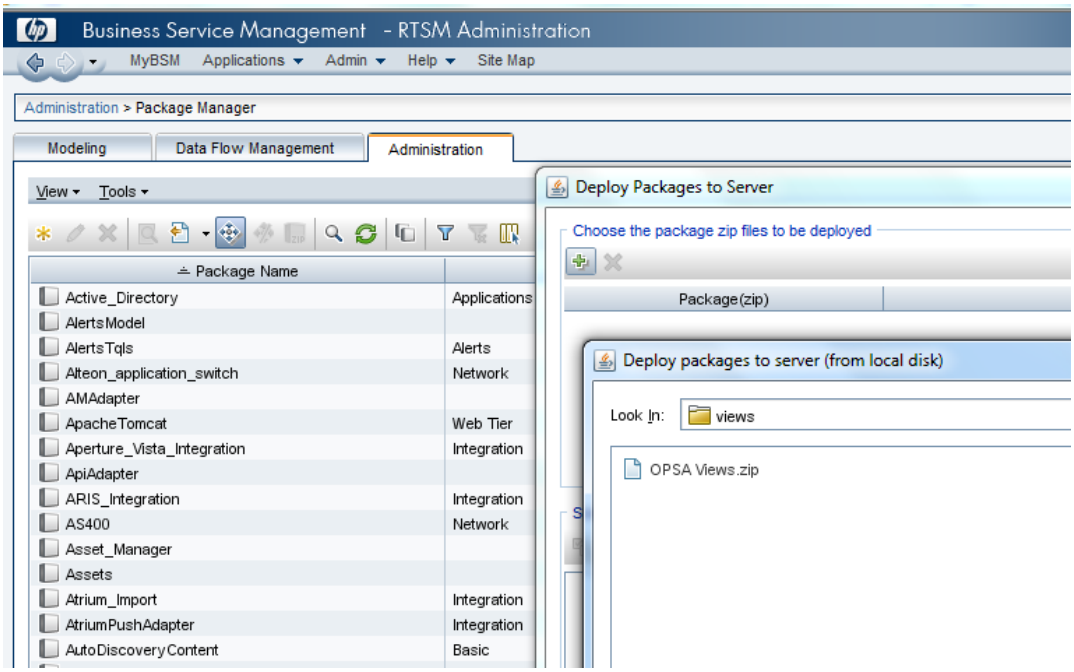
   b. Access the **Package Manager** module through the BSM UI:



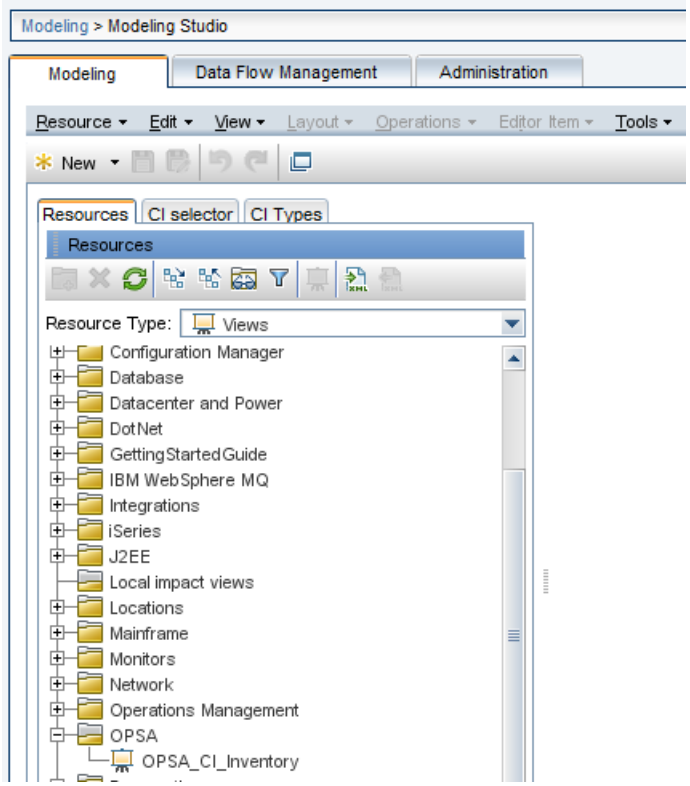
c. Select the **Deploy packages to server (from local disk)** option.



d. Select the **OPSA_Views.zip** file from the local disk as shown in the following screen shot:



e. Once deployed, the views should be visible in the Modeling studio as shown in the following screen shot:



2. Gather the following information to prepare for configuring the HP BSM RTSM CI Collection.

Information to Collect Before Running the opsa-collection-setup.sh Script

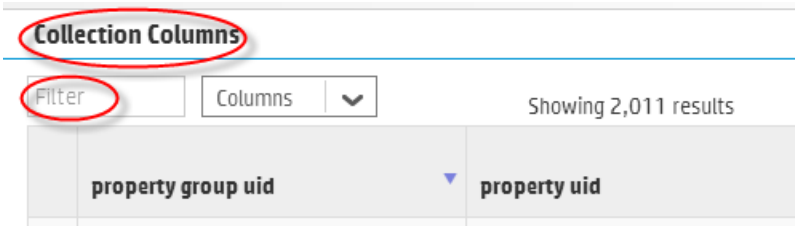
Field	Value
rtsmserver.hostdnsname	The fully-qualified domain name of the RTSM DPS server.
rtsmserver.port	`21212`: The port used to connect to the RTSM DPS server.
rtsmserver.username	`admin`: The user name to use for connecting to the RTSM DPS server.
rtsmserver.datasource_type	`rtsm`

3. Run the interactive `opsa-collection-setup.sh` script.
4. When prompted, enter the tenant admin user name and password.
5. Enter `7` to begin configuring an HP BSM RTSM CI Collection.

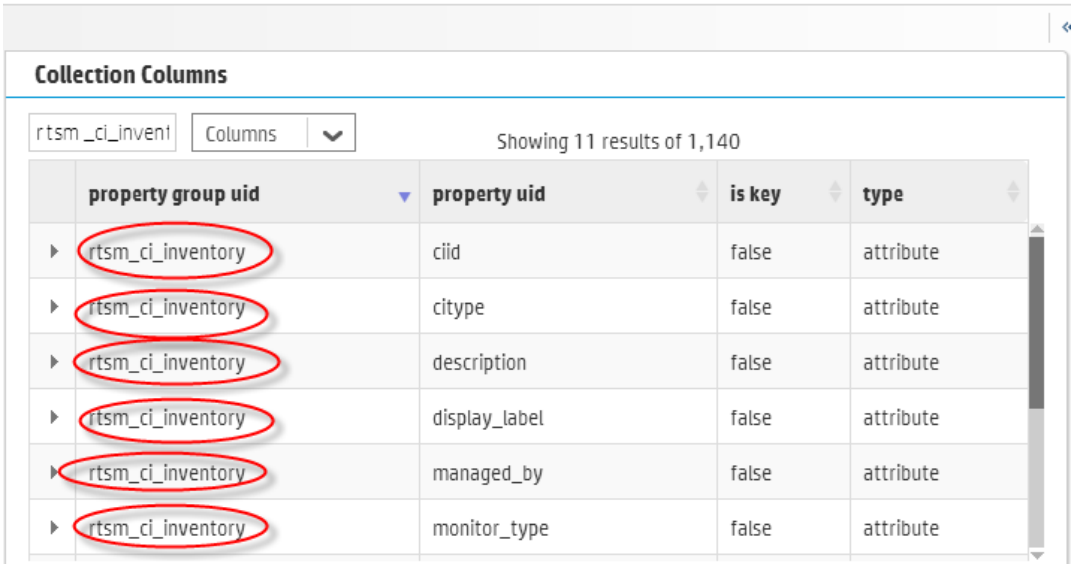
6. Follow the prompts to enter the requested information.
7. Enter `f` or `finish` after you finish entering all of your information.
8. Enter `execute 4` to deploy this collection to the collector host.
9. To check for success, enter `list` and check that the HP BSM RTSM CI Source was added to the collector host.
10. Let the collection run for five minutes or longer. From the OpsA Console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `rtsm`, a domain of `ci`, and a group of `inventory` when creating the collection. The resulting property group uid would be `rtsm_ci_inventory`.

a. Type the property group uid (`rtsm_ci_inventory`) for this collection in the **Collection ColumnsFilter:**



b. After typing the property group uid (`rtsm_ci_inventory`) for this collection in the **Collection Columns Filter**, you should see information for this collection.



11. Create dashboards and query panes for the data you are now collecting. Follow the instructions shown in the *Dashboards and Query Panes* section of the *Operations Analytics Help* for information about creating dashboards and query panes.

12. Create AQL functions for the data you are now collecting. See the instructions shown in the *Define Analytic Query Language (AQL) Functions* section of the *AQL Developer Guide* for more information.

13. If you want to add tags to an HP BSM RTSM CI Collection, use the `opsa-tag-manager.sh` command. See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Configuring an HP Business Process Monitor Collection

Note: If you do not want to configure this collection by using the **Collections Manager** located in the Operations Analytics console, you can configure this collection using the steps in this section.

After you complete the steps in this section, HP Business Process Monitor (BPM) starts sending data to the BPM Collection. The BPM Collection collects metrics related to application transaction response times. The BPM Collection collects data as it arrives from BPM.

Note: You must set the `max heap size` to 3 GB or higher on the Operations Analytics Server Appliance when the following conditions are met:

- OpsA monitors 100 or more BPM applications with 100 or more transactions per application.
- OpsA users make five or more concurrent attempts to open the OpsA default BPM dashboard.

To set the `max heap size`, do the following

1. Edit the `/opt/HP/opsa/jboss/bin/standalone.conf` file.
2. Make the change shown in bold font in the in JAVA_OPTS section:
```
#
# Specify options to pass to the Java VM.
if [ "x$JAVA_OPTS" = "x" ]; then
JAVA_OPTS="-Xms64m -Xmx3072m -XX:MaxPermSize=256m -
Djava.net.preferIPv4Stack=true"
JAVA_OPTS="$JAVA_OPTS -Djboss.modules.system.pkgs=$JBOSS_MODULES_
SYSTEM_PKGS -Djava.awt.headless=true"
else
echo "JAVA_OPTS already set in environment; overriding default
settings with values: $JAVA_OPTS"
fi
```
3. Save your changes.
4. Run the following command to restart the OpsA Server Appliance:
```
$OPSA_HOME/bin/opsa-server restart
```
See the *opsa-server* reference page (or the Linux manpage) for more information.

Note: Connecting two or more Operations Analytics (OpsA) Collector Appliances to the same BSM server host is not a supported configuration.

Complete the following before proceeding:

Note: For this example, the fully-qualified domain name of the BSM DPS server is `servername.location.domain.com`.

Add an entry for the BSM DPS server to the `/etc/hosts` file in the domain in which the OpsA Collector Appliance resides. For example, you would add a line for the BSM DPS server to the `/etc/hosts` file using the following format:
`10.1.2.3 servername.location.domain.com` *`servername`*.

Note: You must use the alias, *servername*, as the BSM DPS host name in the node list file.

Setting the Correct BSM User Name Permissions

When configuring either a BSM RTSM CI collection or a BPM Collection in OpsA you must enter a BSM user name. This BSM user name is used for connecting to the RTSM DPS server, and must be configured for the correct roles.

Note: The user you plan to use for the BPM Collection must be part of the Integration Users Group and include the following OpenAPI required roles:
`CmdbOpenApiQuery`
`CmdbOpenApiClassModel`
`CmdbOpenApiUpdate`
`CmdbOpenApiImpact`

These values could be named as shown below in newer versions:
`RTSMOpenApiQuery`
`RTSMOpenApiClassModel`
`RTSMOpenApiUpdate`
`RTSMOpenApiImpact`

Before competing the remaining configuration steps in this section, do the following to test if the user has the required permissions:

1. Try to log on to BSM as your users using the following URL :
   `http://<BSM>:21212/axis2/services/UcmdbService`

   Note: The log on is successful it you see a web page that shows the following message: `Please enable REST support in WEB-INF/conf/axis2.xml and WEB-INF/web.xml.`

   If your credentials are not accepted you are prompted to enter your user name and password. If this happens, the user does not have the required permissions.

2. If the previous step fails, your user is missing some required permissions. Do not continue until you do the following:

   a. Open the RTSM JMX console using the following URL:
      `http://<BSM>:21212/jmx-console/`

   b. Under the **UCMDB** heading, navigate to **UCMDB:service=Security Services**.

   c. Invoke `setRolesForUser JMX` and give the user either the Admin role or all of the OpenAPI roles:
      Admin role:
      `Admin`

      OpenAPI related roles:

```
CmdbOpenApiQuery, CmdbOpenApiClassModel, CmdbOpenApiUpdate,
CmdbOpenApiImpact
```

Note: To prevent making mistakes when entering the role names, retrieve the available roles by invoking `getAclController JMX` then copy and paste the role names.

3. Repeat step 1 to verify that you correctly made the permissions changes.

Now the BSM user you tested should have all of the required permissions.

Configuration Steps (Automated Method)

To configure a BPM collection, do the following:

Information to Collect and Actions to Take Before Running the opsa-collection-setup.sh Script

Input Requested by opsa-collection-setup.sh Script	Value or Action
Obtain the credentials for accessing the BPM connection:	• The active BSM DPS Server (DPS Hostname) • The RTSM Server Port (21212) • The user name (the RTSM Admin user) **Note**: Admin is the role as it appears in the RTSM console. It is important to use the right role for the integration to work correctly. **Note**: The Operations Analytics integration with BPM can connect to a BSM DPS Server configured for failover. However, if the BSM DPS Server fails, OpsA does not automatically reconnect to the failover DPS server. Operations Analytics will no longer collect BPM data until the active BSM DPS Server is back online.
BSM RTSM Server HostName	The fully-qualified domain name of the BSM RTSM server.

Information to Collect and Actions to Take Before Running the opsa-collection-setup.sh Script, continued

Input Requested by opsa-collection-setup.sh Script	Value or Action
BSM DPS Server	Add an entry for the BSM DPS server to the `/etc/hosts` file in the domain in which the OpsA Collector Appliance resides. For example, you would add a line for the BSM DPS server to the `/etc/hosts` file using the following format: `10.1.2.3 servername.location.domain.com servername`
Server Username	The username for the BSM RTSM server.
Server Password	The password for the BSM RTSM server username.
Server Port number.	The well-known port number for communicating with the BSM RTSM server.

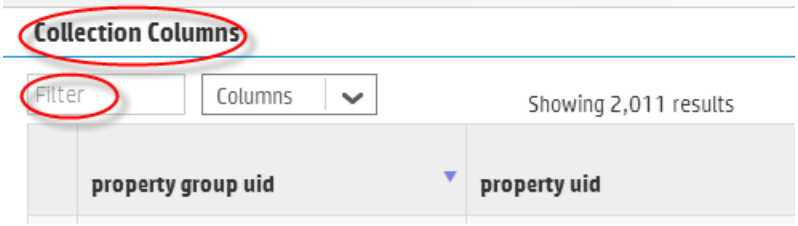
Follow the instructions in this section to configure a BPM Collection for Operations Analytics.

1. Run the interactive `opsa-collection-setup.sh` script.
2. When prompted, enter the tenant admin user name and password.
3. Enter `8` to begin configuring a BPM Collection.
4. Follow the prompts to enter the requested information.
5. Enter `f` or `finish` after you finish entering all of your information.
6. After you see a message similar to the following: `Successfully authenticated connection configuration for BPM`, enter `execute 8` to deploy this collection to the collector host..
7. To check for success, enter `list` and check that the BPM Source was added to the collector host.
8. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

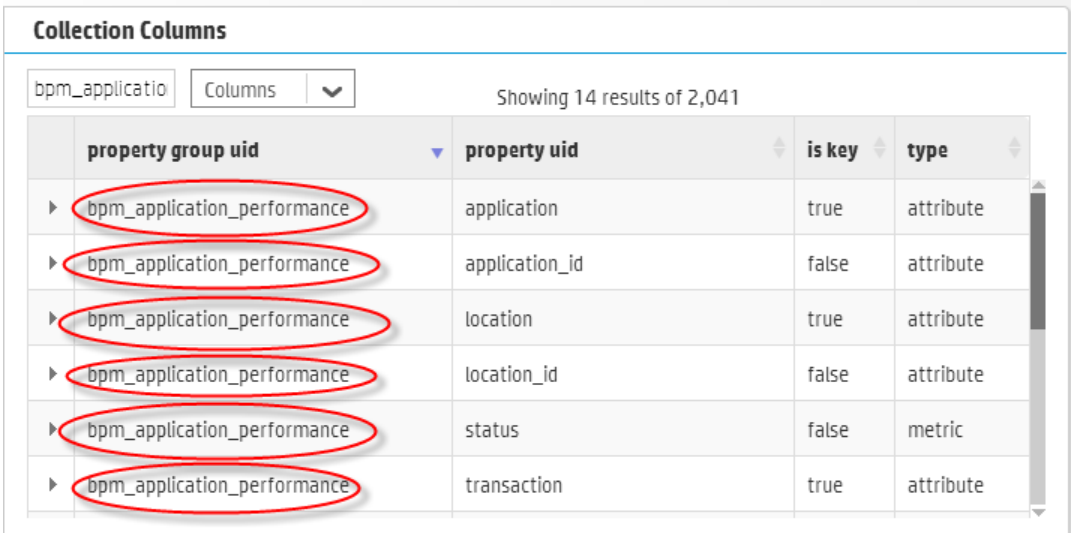
   Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `bpm`, a

domain of `application`, and a group of `performance` when creating the collection. The resulting property group uid would be `bpm_application_performance`.

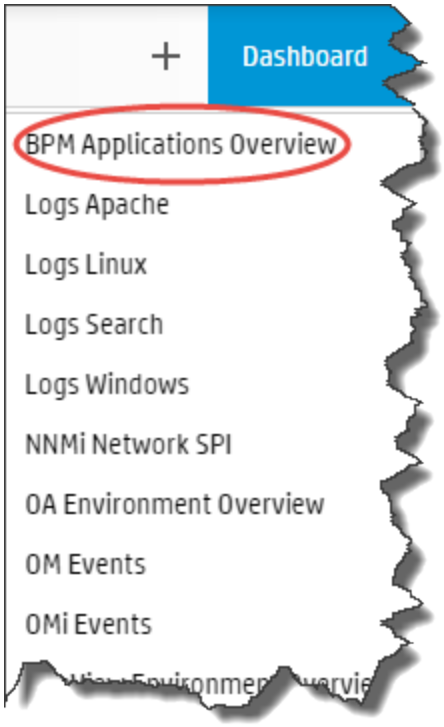
a. Type the property group uid (`bpm_application_performance`) for this collection in the **Collection ColumnsFilter:**



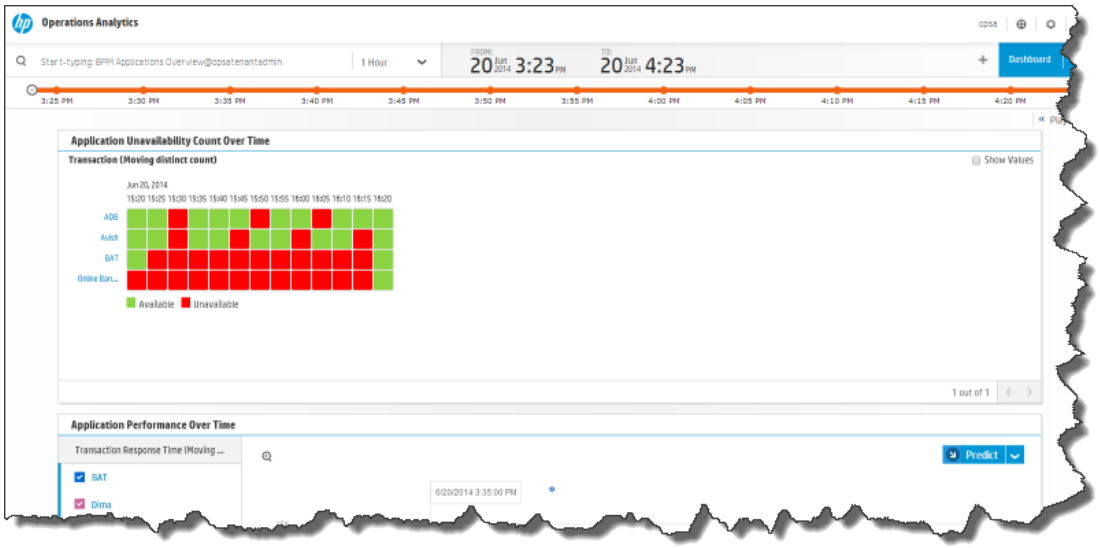
b. After typing property group uid (`bpm_application_performance`) for this collection in the **Collection ColumnsFilter**, you should see information for this collection.



9. From the Operations Analytics console, open the **BPM Applications Overview** dashboard to view some of the collected information for this collection:



The following is a small example of BPM Collection data provided by the **BPM Applications Overview** dashboard.



10. If you want to add tags to an HP Operations Manager Events Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the

opsa-tag-manager.sh reference page (or the Linux manpage) for more information.

Configuring ArcSight Logger Out of the Box Smart Connector Collections

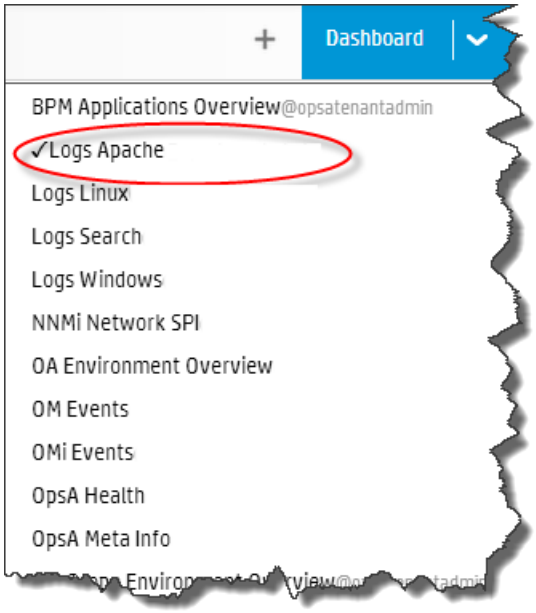
The following steps are required to configure and publish the out of the box SmartConnector Collections. For more information about these connectors, see *Out of the Box Log Content* in the *Operations Analytics Installation Guide.*

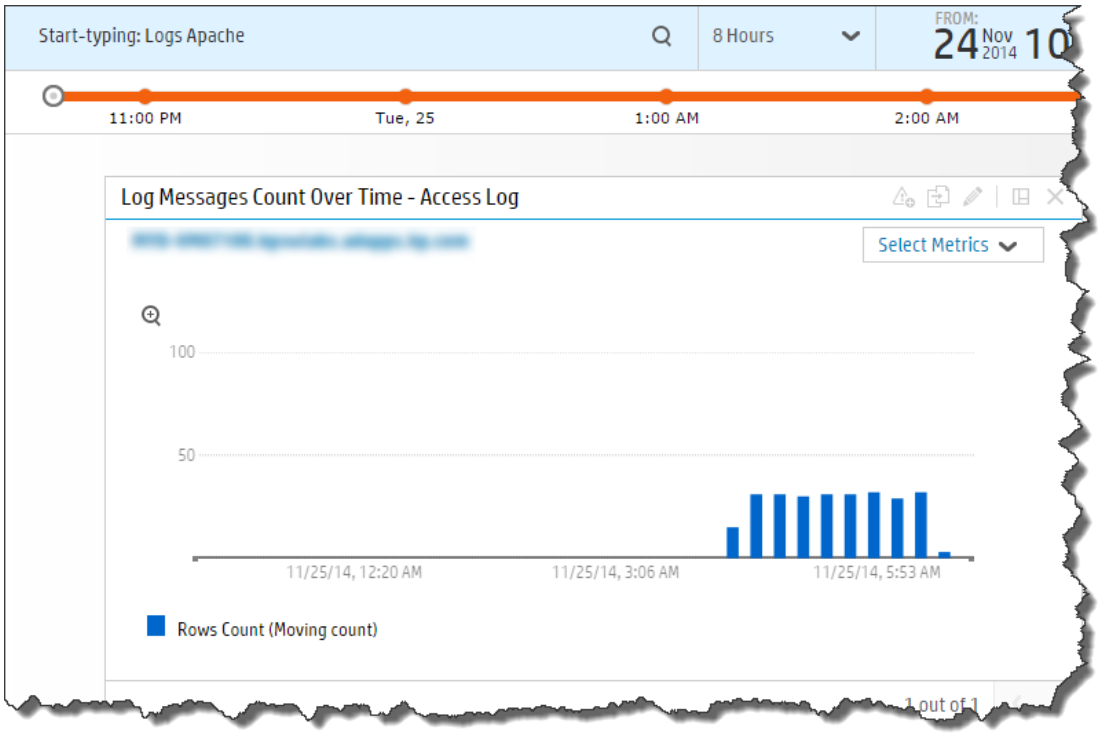
Note: From a resource perspective, there is a limit to the number of Logger sessions supported by Operations Analytics (OpsA) Software. HP strongly recommends that, when you configure Logger collections, you assign those Logger collections to one common OpsA collector. Doing so reduces the number of Logger sessions.

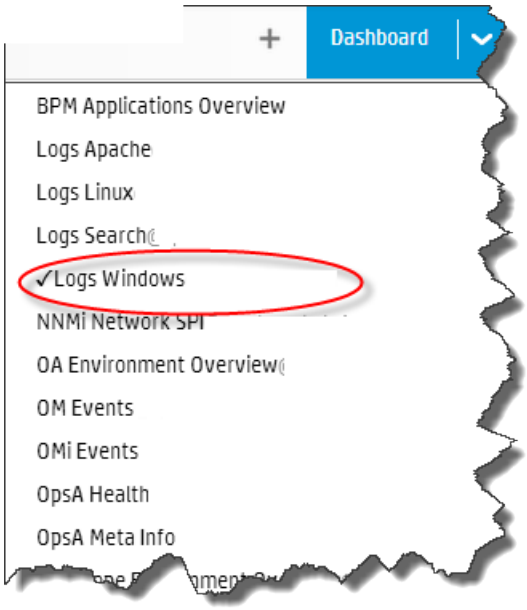
1. Log on to the OpsA Server Appliance as an opsa user.
2. Run the interactive `opsa-collection-setup.sh` script.
3. When prompted, enter the tenant admin user name and password.
4. Enter `9` to begin configuring an ArcSight Logger Collection Configuration.
5. Enter the index number for the SmartConnector you want to configure:
   - 1: Apache Access
   - 2: Apache Error
   - 3. Linux Syslog
   - 4. Windows Event
6. Enter `f` or `finish` after you finish entering all of your information.
7. Enter `execute 9` to deploy this collection to the collector host.
8. To check for success, enter `list` and check that the sources you requested were added to the collector host.
9. Exit the `opsa-collection-setup.sh` script.
10. To test that the process ran correctly wait five minutes, open the following log file on the OpsA Collector server, and confirm that it does not contain any errors:

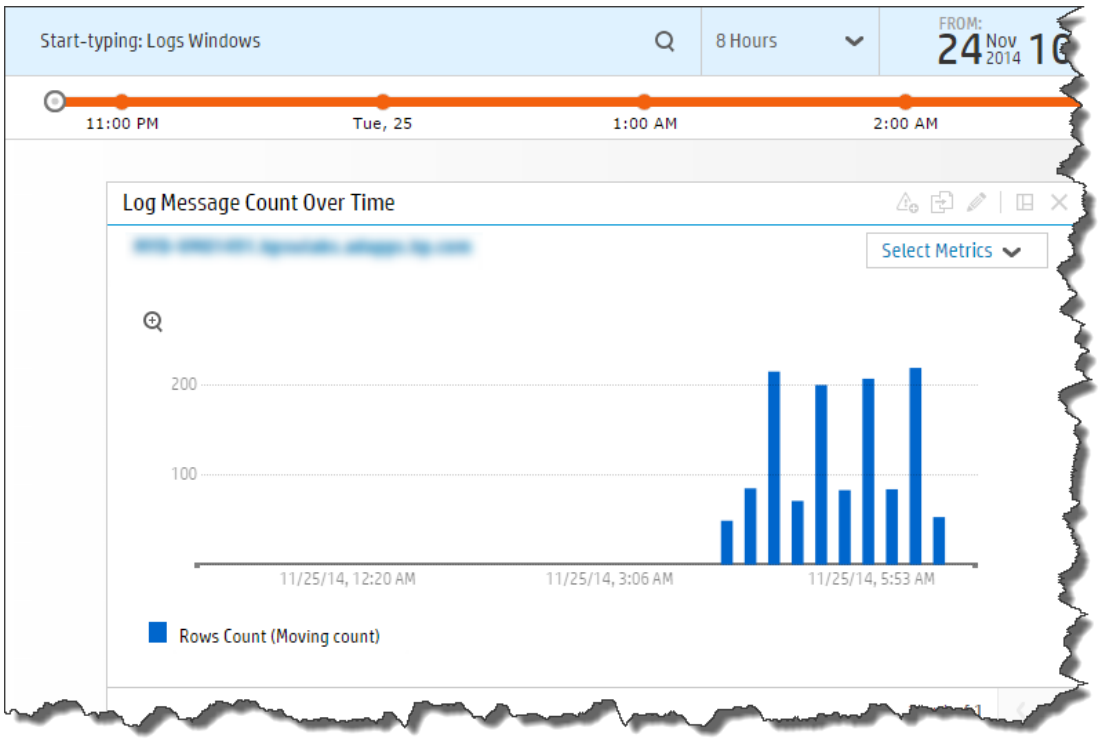
    `$OPSA_HOME/log/opsa-collector.log`

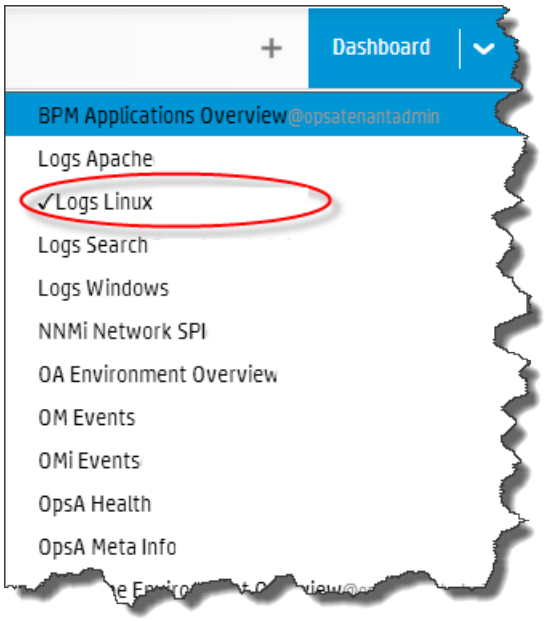
11. From the Operations Analytics console, open the following dashboards to view some of the collected information for these collections:






Dashboard
BPM Applications Overview
Logs Apache
Logs Linux
Logs Search
✓Logs Windows
NNMi Network SPI
OA Environment Overview
OM Events
OMi Events
OpsA Health
OpsA Meta Info


Start-typing: Logs Windows
8 Hours
FROM:
24 Nov 2014 10
11:00 PM
Tue, 25
1:00 AM
2:00 AM
Log Message Count Over Time
Select Metrics
200
100
11/25/14, 12:20 AM
11/25/14, 3:06 AM
11/25/14, 5:53 AM
Rows Count (Moving count)



Configuring an NNMi Custom Poller Collection

Note: If you do not want to configure this collection by using the **Collections Manager** located in the Operations Analytics console, you can configure this collection using the steps in this section.

After you complete the steps in this section, the NNMi Custom Poller Collection reads data from the CSV files within 60 seconds of the file being placed in the source directory. You can use an NNMi Custom Poller Collection to collect numeric metrics from any NNMi Custom Poller MIB expression.

The NNMi Custom Poller collection template is generic, as it stores a MIB instance and a MIB value. This MIB value id defined as a `float` data type. Since Custom Poller can poll any type of MIB value, you must only send Custom Poller CSV files that contain numeric MIB values to this collection .

If you must create an Operations Analytics (OpsA) collection that matches what is being collected, you must create a custom CSV collection template. For example, you might have a Custom Poller Collection for network interface errors. To use this Custom Poller Collection, create a Custom CSV Collection, adding the appropriate tags and labels to identify the data for that collection. See "Configuring a Custom CSV Collection" on page 88 for more information.

1. To enable NNMi to export Custom Poller collections, do the following:

   a. Using the NNMi console, enable NNMi to export custom poller collections to make the metrics from your collections available for OpsA . Configuring NNMi to export custom poller collections enables NNMi to export metrics, such as CSV files, into the following directory:

      - *Windows*:
        `<Install_Dir>\ProgramData\HP\HP BTO Software\shared\nnm\databases\custompoller\export\final`
      - *UNIX*:
        `/var/opt/OV/shared/nnm/databases/custompoller/export/final`

        See the *HP Network Node Manager i Deployment Reference*, the *HP NNMi Help*, or the *HP Network Node Manager i Software Step-by-Step Guide to Custom Poller White Paper* for more information.

   b. The default configuration for the custom poller collection template is for OpsA to read all of the files having file names that match the `*.csv*` or `*.gz*` pattern. If you need the collector to read a different set of files, the OpsA administrator must edit the appropriate custom poller collector template file and specify a different file pattern. To change the pattern, edit the custom poller collection template and make the value changes you must make to the `filepattern=` tag.

Note: You must make the files exported from the `/var/opt/OV/shared/nnm/databases/custompoller/export/final` directory on NNMi available on the OpsA Collector Appliance in the `/opt/HP/opsa/data/nnm` directory.

If you want to use a different directory than `/opt/HP/opsa/data/nnm`, do the following:

a. Edit the following collection template:
   `/opt/HP/opsa/conf/collection/server/config.templates/nnm/1.0/netperf/mib/nnm_netperf_mib_collection.xml`.

b. Specify a different directory for the `sourcedir` attribute.

Note: The `opsa` user on the OpsA Collector Appliance must have read and write access to the NNMi files on the OpsA Collector Appliance to move them to the processed directory. The default process directory is `/opt/HP/opsa/data/nnm_processed`.

For example, to configure read and write access to the NNMi files to the OpsA Collector Appliance when the files are located on a Windows server, do the following:

a. On a Windows server, navigate to **Computer Management** > **System Tools** > **Shares** > **Shared Folders**.

b. Right-click beneath shares and open the new share wizard.

c. Create shares for the directories in which the .csv files are stored.

d. From the OpsA Collector Appliance, add the correct entries to the `/etc/fstab` file. Use the following entries as a model:

```
//10.17.18.19/final /opt/HP/opsa/data/nnm cifs
username=administrator,password=password,uid=opsa,rw 0 0
//10.15.14.13/componentfinal /opt/HP/opsa/data/netcomponent
cifs username=admin,password=passwd,uid=opsa,rw 0 0
//10.15.14.13/interfacefinal /opt/HP/opsa/data/netinterface
cifs username=admin,password=passwd,uid=opsa,rw 0 0
```

e. Use the mount -a command to get the directories mounted.

2. Run the following command from the OpsA Server Appliance to create the collector configuration:

```
$OPSA_HOME/bin/opsa-collection-config.sh -create -collectorhost
<fully-qualified domain name of the collector host> -source nnm -
domain netperf -group mib -username opsatenantadmin
```

Note: The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

Note: The `opsa-collection-config.sh` script uses the values of `source`, `domain`, and `group` to select the right predefined collection template to create the desired collection configuration.

3. Run the following command from the OpsA Server Appliance to validate the collection configuration you just created:

```
$OPSA_HOME/bin/opsa-collection-config.sh -list -allversions -
collectorhosts -username opsatenantadmin
```

To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its version, domain, and group.

4. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance:

```
$OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost
```

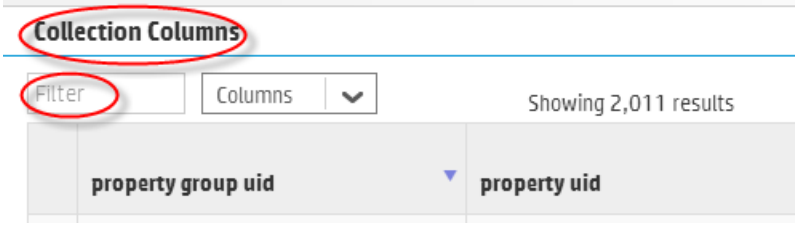
`<fully-qualified domain name of the collector host>` `-username opsatenantadmin`

The `-publish` option uploads the collection configuration you created to the OpsA Collector Appliance. To verify that the collector configuration published successfully, look for a message stating that the publish was successful and that a table was successfully created.

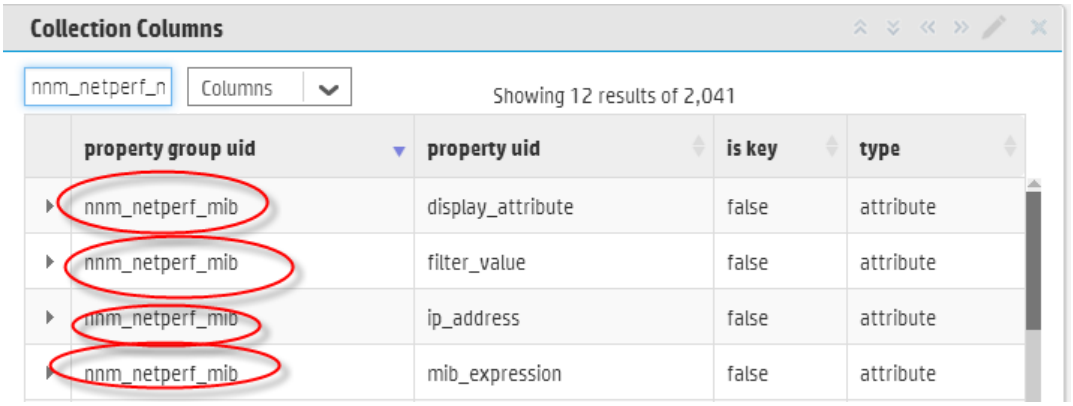
5. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

> **Note:** The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this example, you would have used a name of `nnm`, a domain of `netperf`, and a group of `mib` when creating the collection. The resulting property group uid would be `nnm_netperf_mib`.

   a. Type the property group uid (`nnm_netperf_mib`) for this collection in the **Collection ColumnsFilter:**

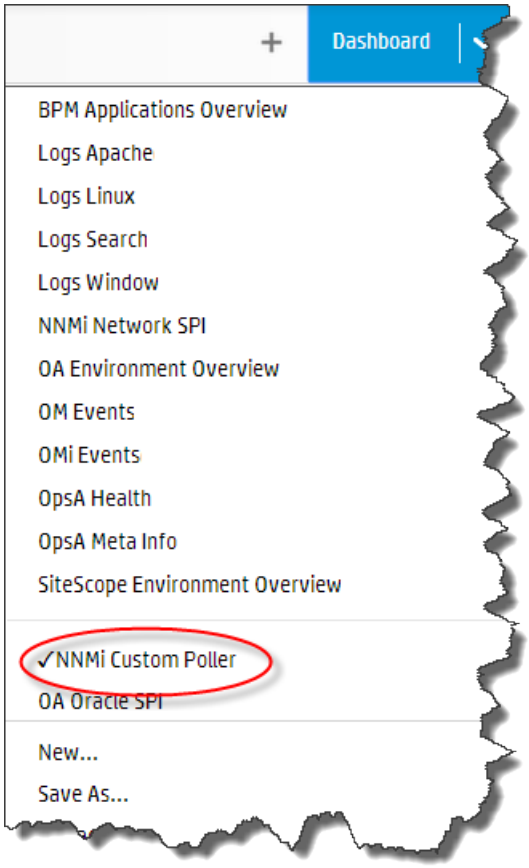


   b. After typing property group uid (`nnm_netperf_mib`) for this collection in the **Collection Columns Filter**, you should see information for this collection.



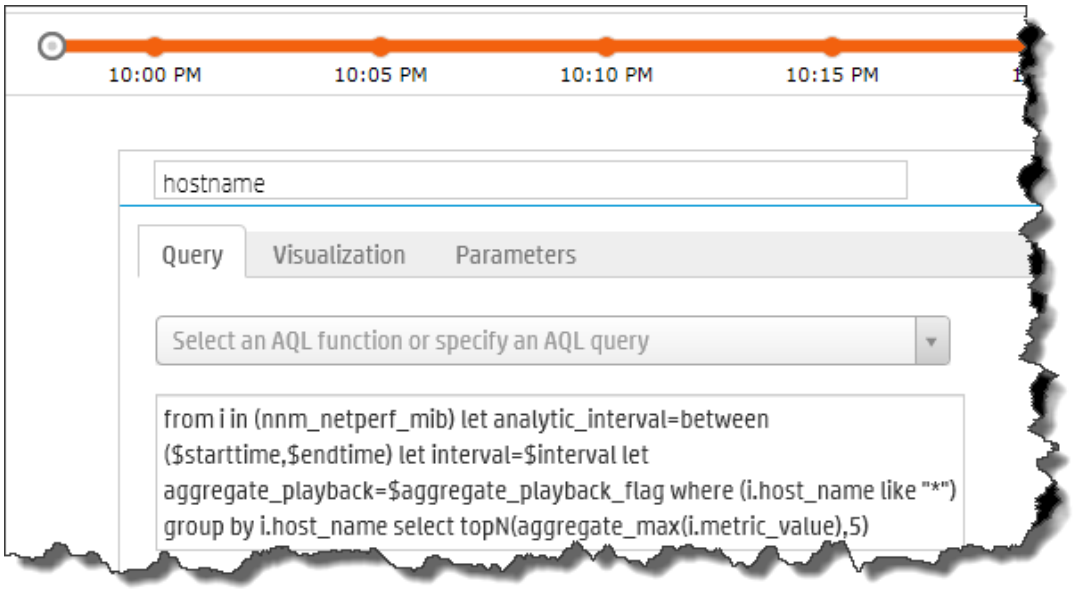
6. Create dashboards and query panes for the data you are now collecting. Follow the instructions shown in the *Dashboards and Query Panes* section of the *Operations Analytics Help* for information about creating dashboards and query panes.

7. For this example, assume you created the **NNMi Custom Poller** dashboard. From the Operations Analytics console, open the **NNMi Custom Poller** dashboard to view some of the collected information for this collection:

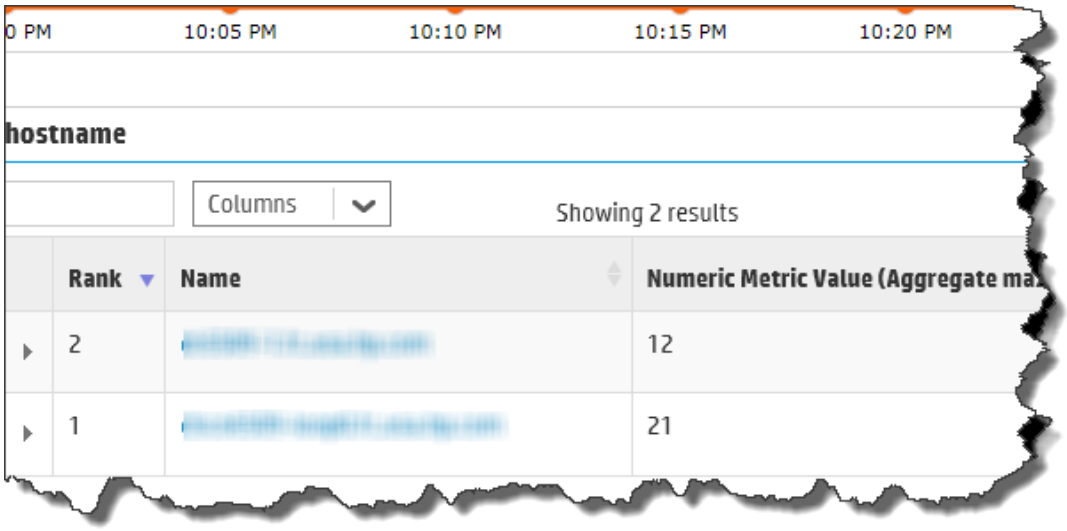


8. Create AQL functions for the data you are now collecting. See the instructions shown in the *Define Analytic Query Language (AQL) Functions* section of the *Operations Analytics Help* for information about creating AQL functions. For example, using the property group information shown in the **OpsA Meta Info** dashboard, you might create AQL functions similar to the examples shown below.

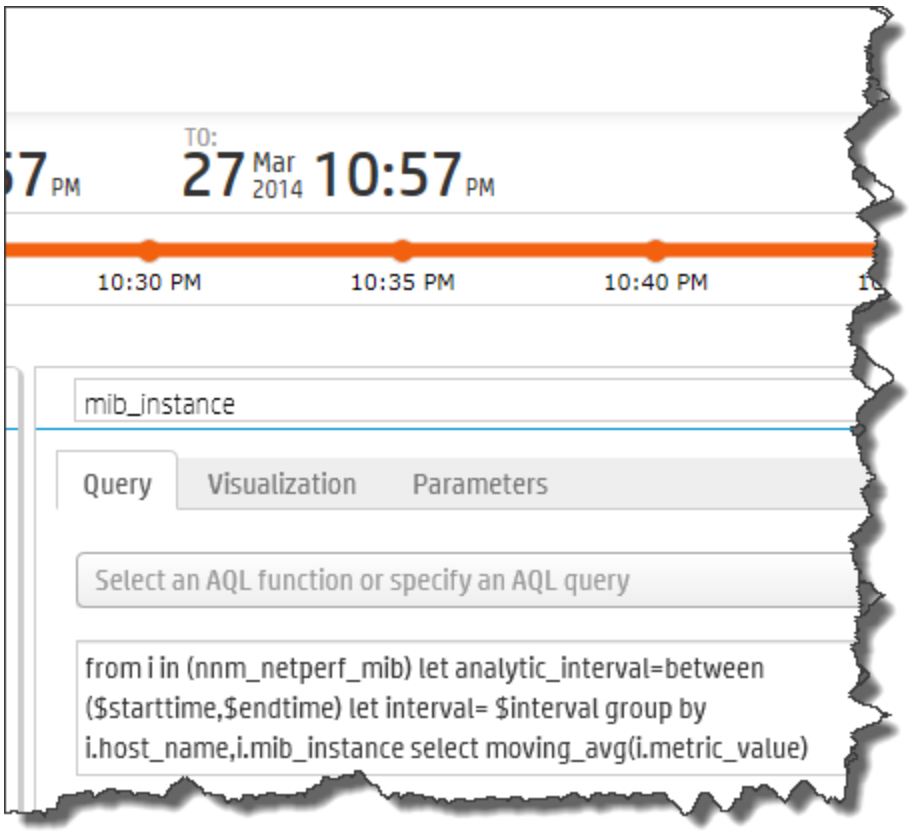
   For the following hostname AQL function:



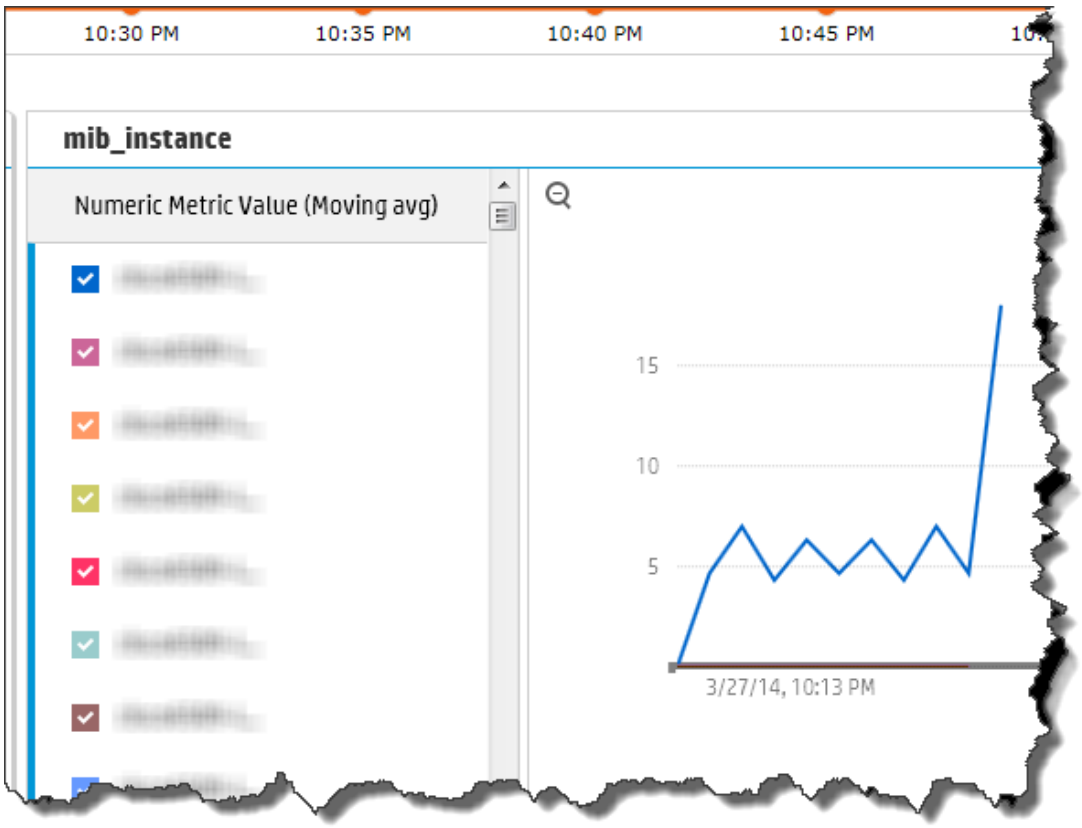
   You might see the following result:



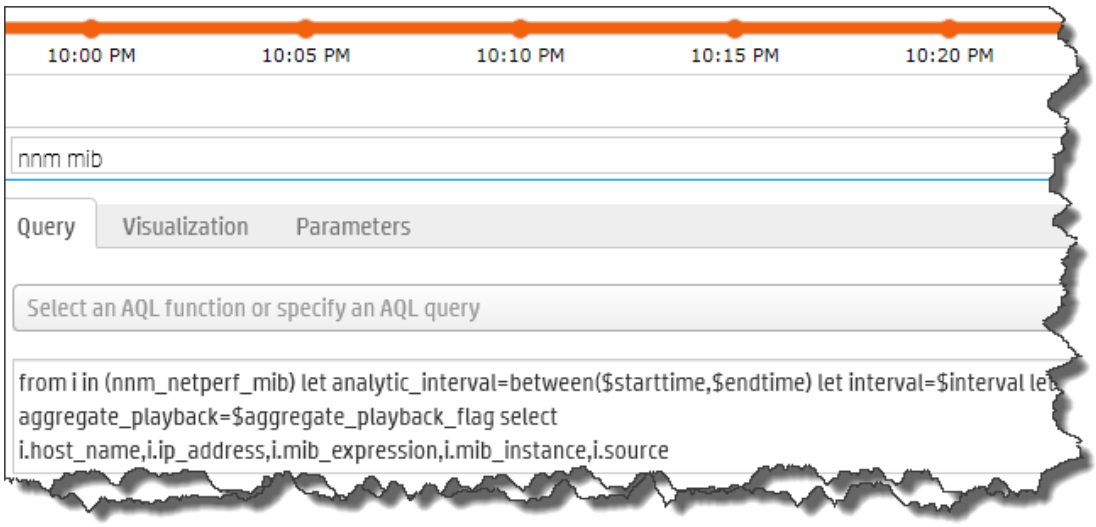
For the following mib_instance AQL function:



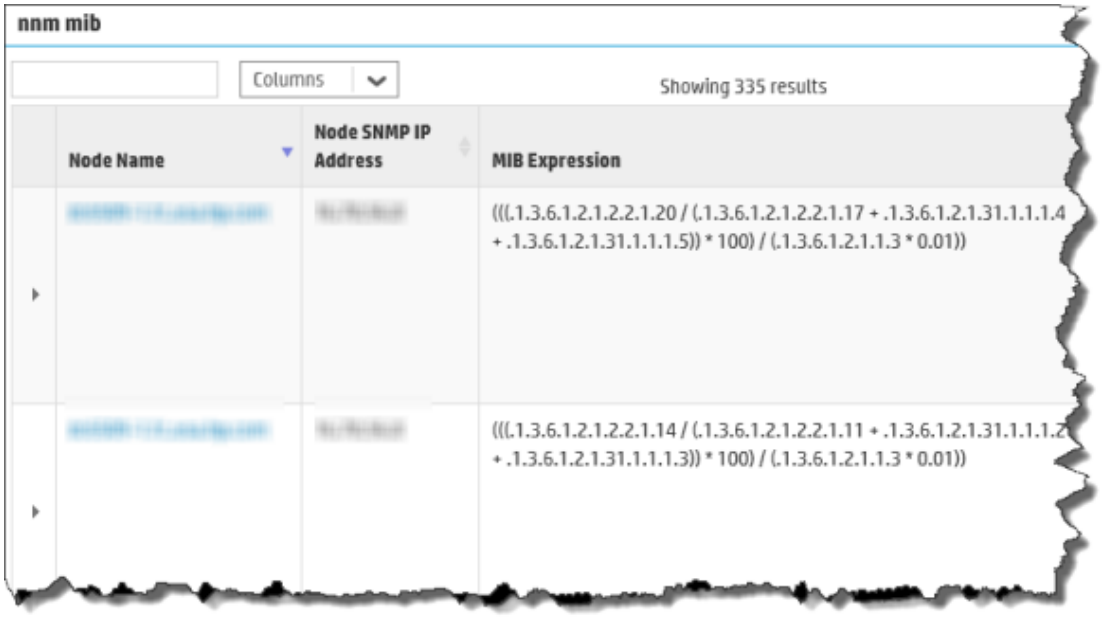
You might see the following mib_instance (metric) result:



For the following nnm_mib AQL function:



You might see the following nnm_mib (mib expression) result:



9. If you want to add tags to a NNMi Custom Poller Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Configuring HP Operations Manager (HPOM) (Creating a Database User Account on an HPOM Database Server)

Performing this task depends on how Microsoft SQL Server is set up in the HPOM environment and how you can configure the HP Embedded Collector to communicate with the HPOM database server. There are two possible scenarios:

- **Scenario 1**: HPOM for Windows 8.x/9.x is installed on one system with Microsoft SQL Server 2005 or Microsoft SQL Server 2008 installed on the same system or a remote system. The HP Embedded Collector, which is installed on another system, can be configured to connect to SQL Server either through Windows authentication or SQL Server authentication (mixed-mode authentication). The authentication method defined in SQL Server can be used in the HP Embedded Collector to configure the HPOM database connection.

- **Scenario 2**: HPOM for Windows 8.x uses Microsoft SQL Server 2005 Express Edition that is embedded with it by default. Similarly, HPOM for Windows 9.x uses the embedded Microsoft SQL Server 2008 Express Edition by default. The authentication mode in this scenario is Windows NT authentication. However, in this case, a remote connection between SQL Server and the HP Embedded Collector is not possible. Therefore, you must create a user account for the HP Embedded Collector so that mixed-mode authentication is possible in this scenario.

Before creating the user account, you must first enable mixed-mode authentication. For the steps, see the Enable Mixed Mode authentication after installation section in the Microsoft Support KB article at the following URL: **http://support.microsoft.com/kb/319930**

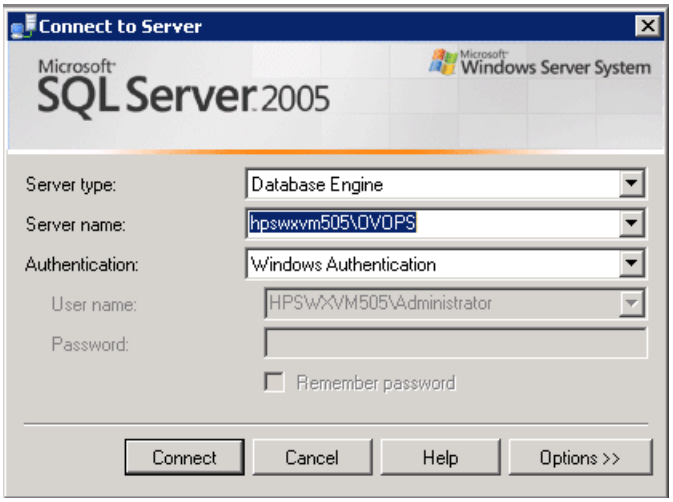
To create a user name and password for authentication purposes, perform the following steps. If you are using Microsoft SQL Server 2008, the steps are similar to the following steps performed in SQL Server 2005:

1. Create a user name and password:

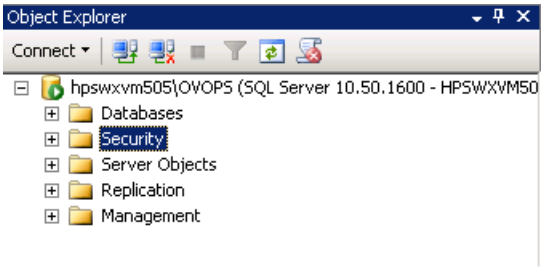
   a. Log on to the HPOM system with embedded Microsoft SQL Server 2005.

   b. The Microsoft SQL Server Management Studio window opens. Click **Start** > **Programs** >**Microsoft SQL Server 2005** > **SQL Server Management Studio**. If SQL Server Management Studio is not installed on your system, you can download it from the Microsoft web site using the following URL: `http://www.microsoft.com/downloads/en/details.aspx?FamilyID=c243a5ae-4bd1-4e3d-94b8-5a0f62bf7796`.

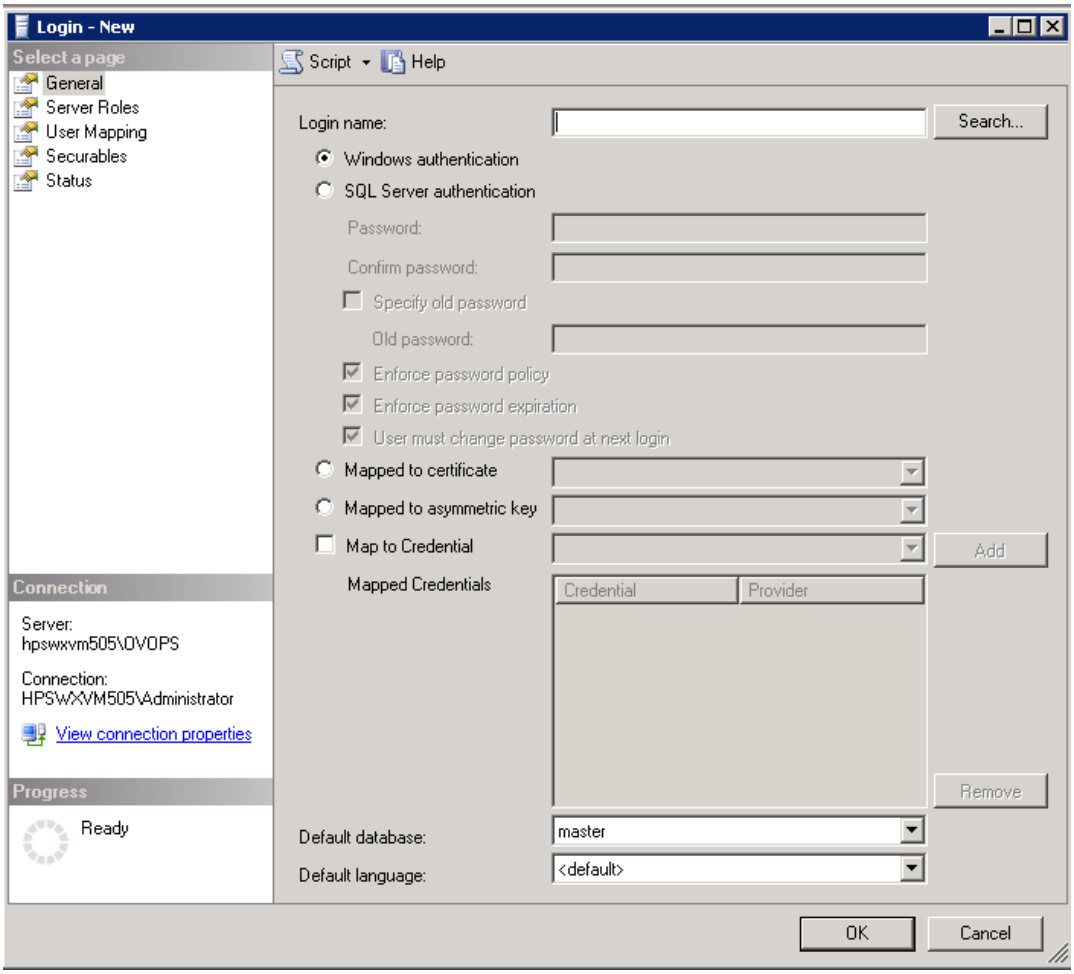
   c. In the **Connect to Server** dialog box, select **NT Authentication** in the **Authentication** list, then click **Connect**.



   d. In the **Object Explorer** pane, expand **Security**.



e. Right-click **Logins** and click **New Login**. The Login - New dialog box opens.



f. In the **Login** name field, type a user name. Specify the other necessary details.

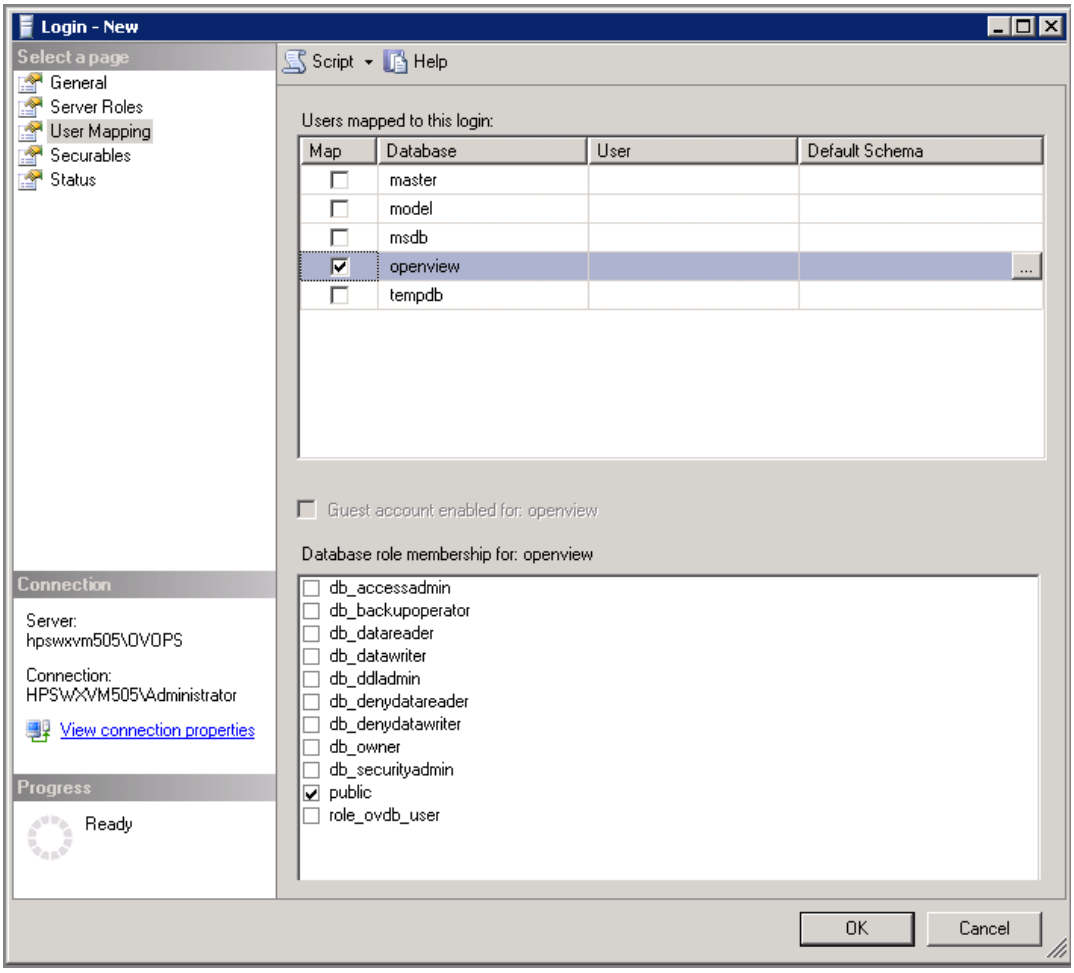
g. Select the **SQL Server authentication** radio button.

h. In the **Password** field, type the password.

i. In the **Confirm password** field, retype the password. You might want to disable the password enforcement rules to create a simple password.

j. Click **User Mapping**.

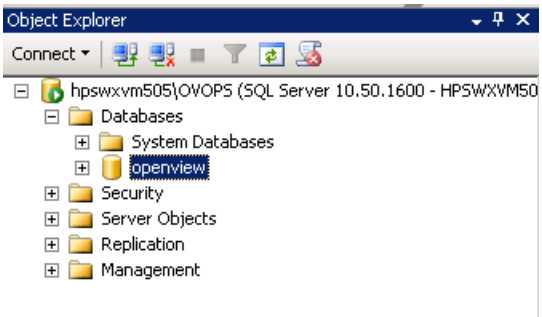
k. Under **Users mapped to this login**, select the check box next to **openview**.



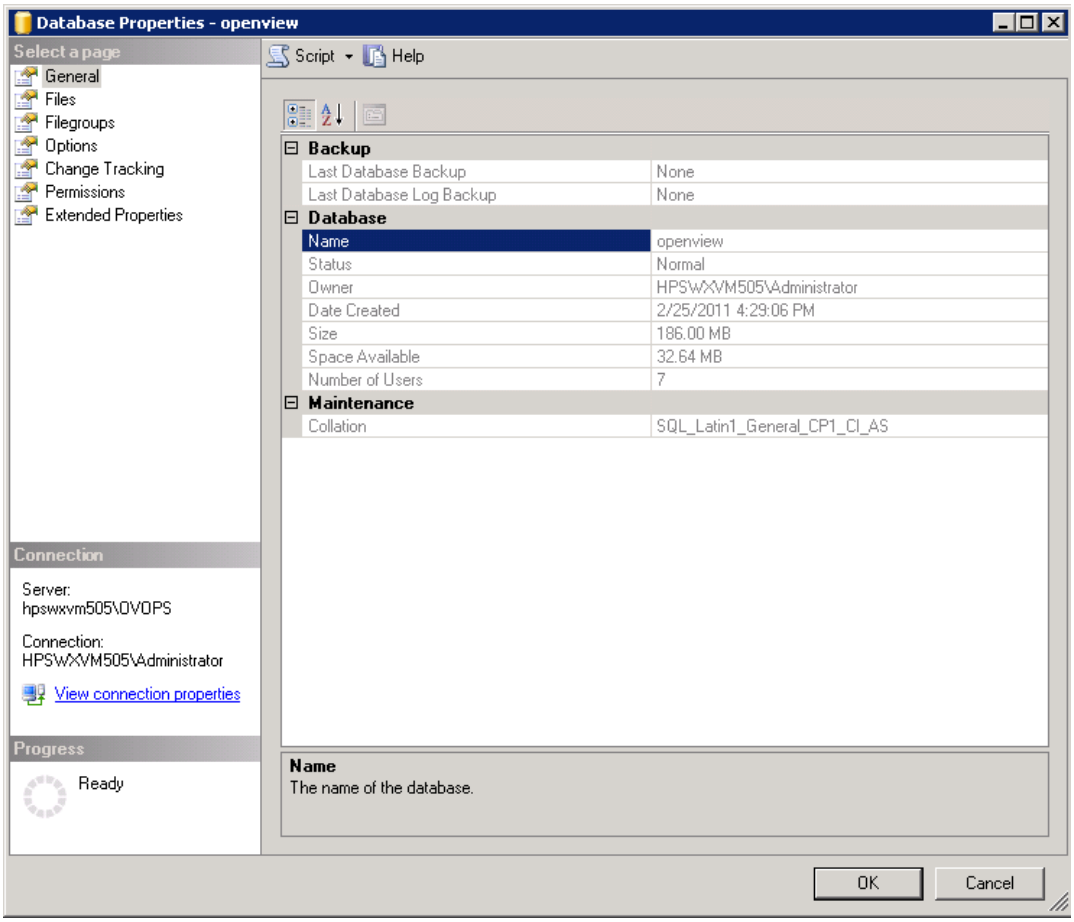
l. Click **OK** to create the user name and password.

2. The database user must have at least the **Connect** and **Select** permissions. To enable the **Connect** and **Select** permissions for the newly created user account, follow these steps:

   a. In the **Object Explorer** pane, expand **Databases**.



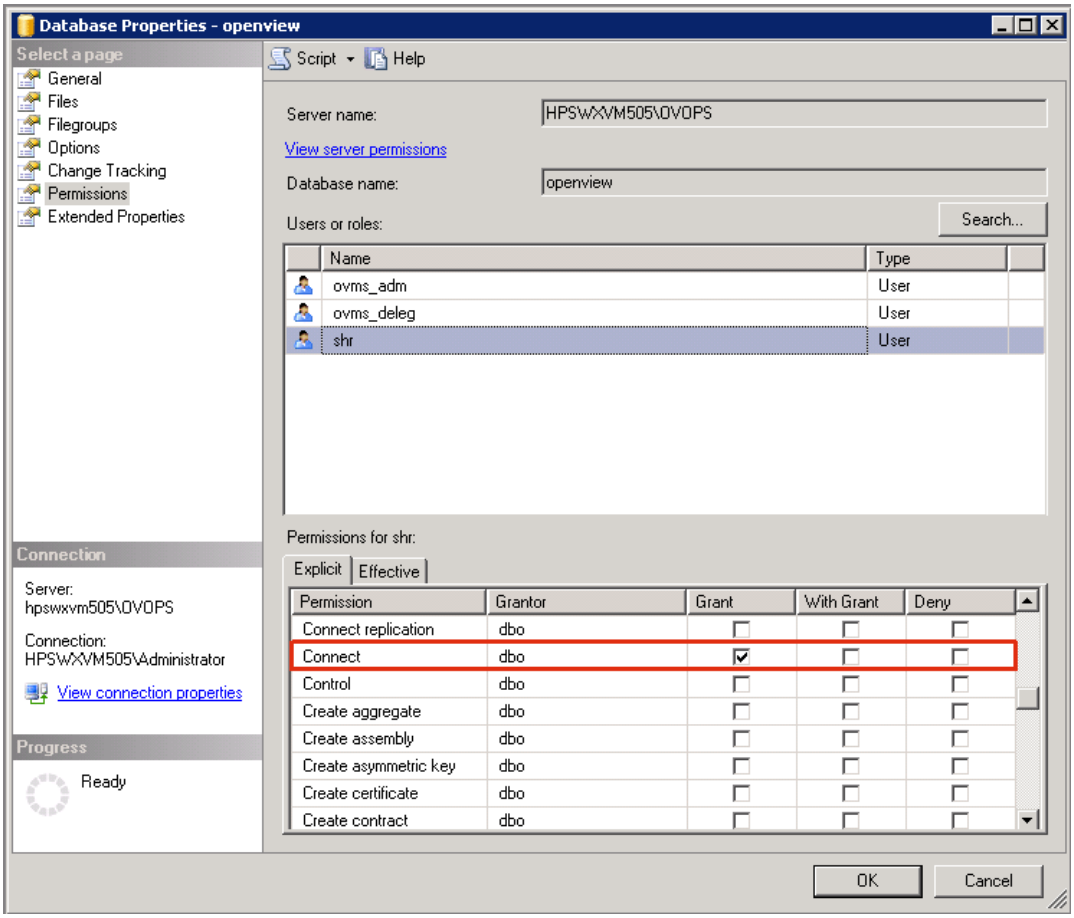
b. Right-click **openview** , then click **Properties**. The Database Properties - openview dialog box opens.



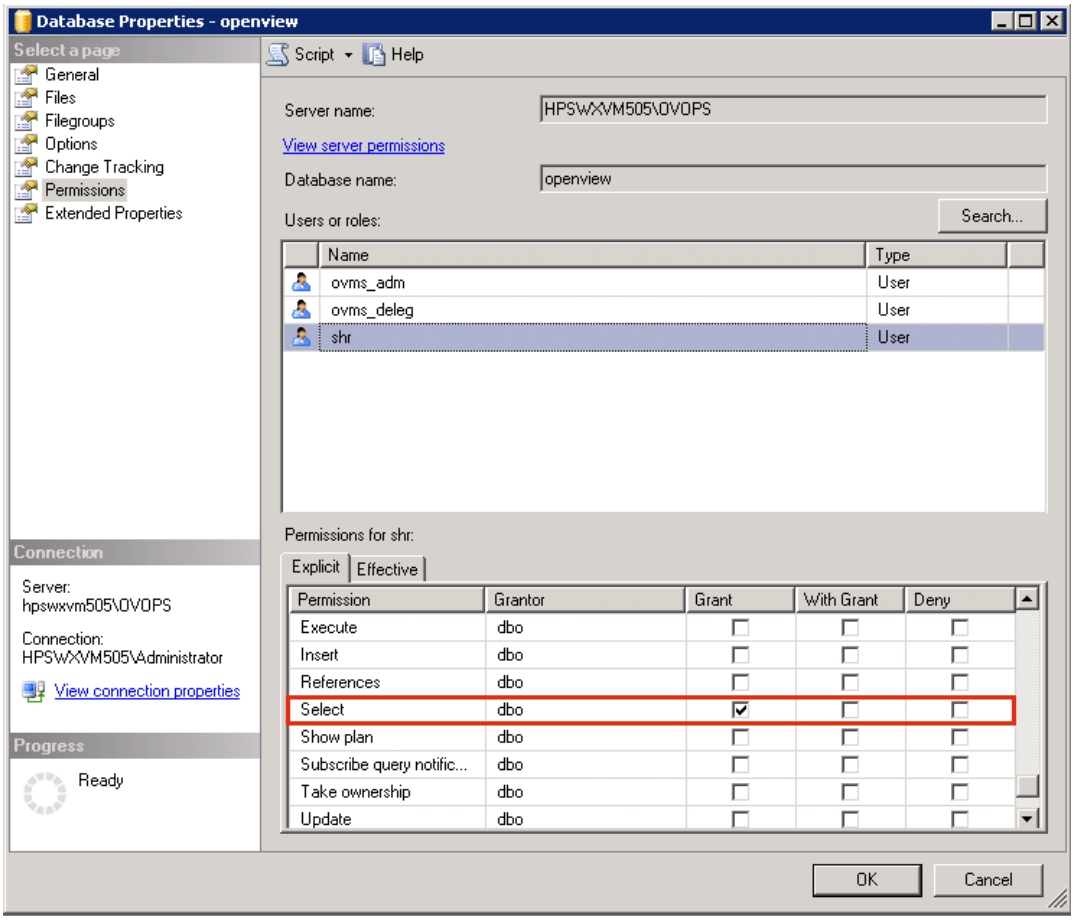
c. Under the **Select a page** pane, click **Permissions**.

d. Under **Users or roles**, click the newly created user account.

e. Under **Explicit permissions for test**, scroll down to the **Connect** permission, then select the **Grant** check box for this permission.



f. Scroll down to the **Select** permission and select the **Grant** check box for this permission.

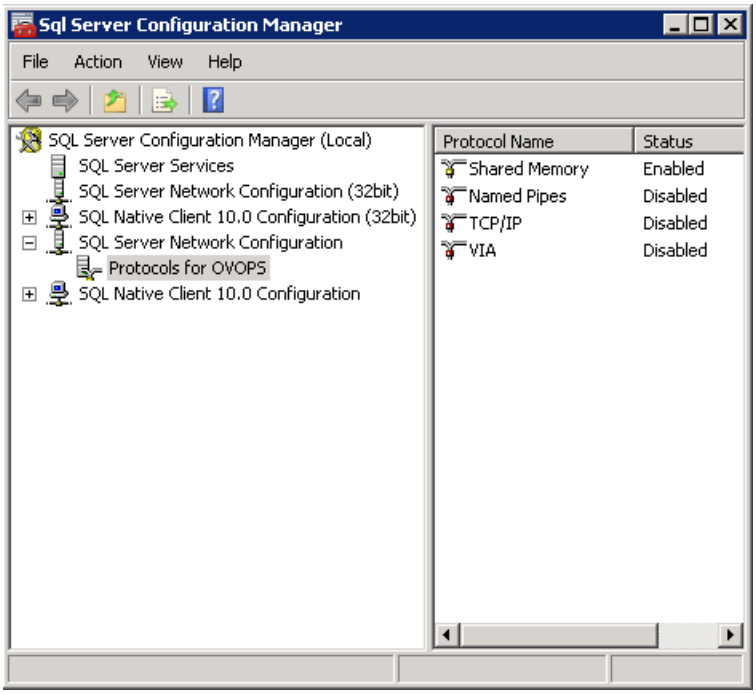


g. Click **OK**.

3. Check for the HPOM server port number:

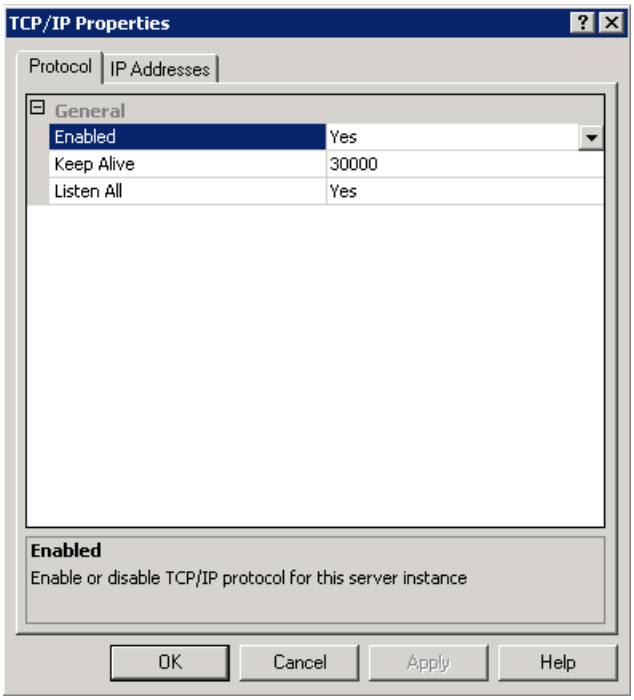
a. Click **Start** > **Programs** > **Microsoft SQL Server 2005** > **Configuration Tools** > **SQL Server Configuration Manager**. The SQL Server Configuration Manager window opens.

b. Expand **SQL Server Network Configuration** and select **Protocols for OVOPS**. If the instance name has been changed, select the appropriate instance name.



c. On the right pane, right-click **TCP/IP**, then click **Enable**.

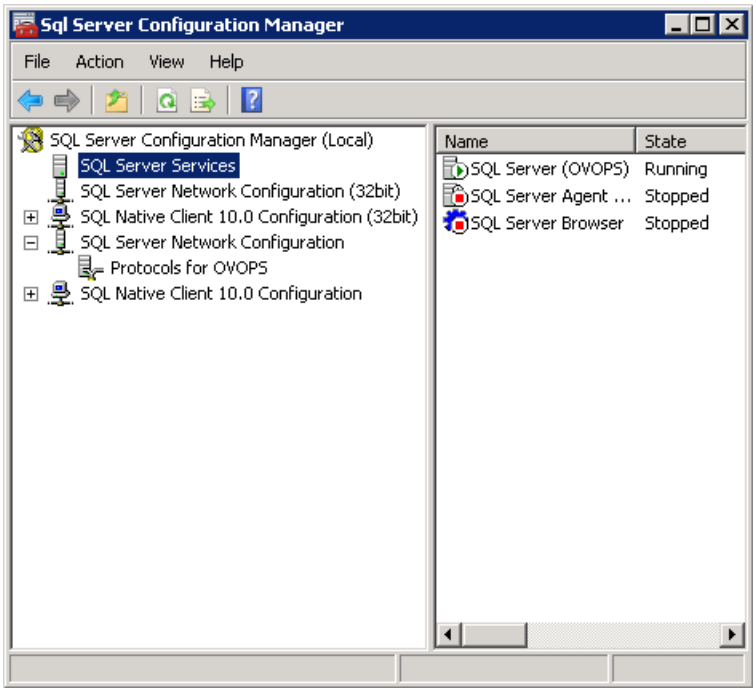
d. Right-click **TCP/IP** again, then click **Properties**. The TCP/IP Properties dialog box opens.



e. On the **IP Addresses** tab, under the **IPAll**, note the port number.

4. Restart the HPOM database server:

a. In the SQL Server Configuration Manager window, click **SQL Server Services**.



b. On the right pane, right-click **SQL Server (OVOPS)**, then click **Restart**.

You can use the newly created user name, password, and the observed instance name and port number when configuring the HPOM data source connection in the Administration Console.

You can perform these steps by using the command prompt utility, osql. For more information, see the Microsoft Support KB article at the following URL: http://support.microsoft.com/kb/325003

Configuring Collections for Custom Data Sources

Operations Analytics (OpsA) relies on collected metrics, topology, event, and log file data from a diverse set of possible data sources. An OpsA Collector Appliance contains the software that listens for data coming from a device. Each server that is running the OpsA Collector software is configured as a Collector Appliance.

To configure OpsA to collect data from the supported custom data sources you plan to use, you must configure collections by creating collection templates that reside on the OpsA Server Appliance. The instructions in this section explain how to configure OpsA to begin collecting data for the custom data sources you plan to use.

Navigate to the instructions for the custom data source or sources you plan to use:

- "Configuring a Custom CSV Collection" below
- "Configuring a Custom SiteScope Collection" on page 95
- "Configuring a Structured Log Collection" on page 101

Configuring a Custom CSV Collection

Note: If you do not want to configure this collection by using the **Collections Manager** located in the Operations Analytics console, you can configure this collection using the steps in this section.

Operations Analytics (OpsA) supports predefined collection templates for configuring data collections using the data sources described in "Configuring Collections using Predefined Templates " on page 25.

To collect data from sources that do not use predefined collection templates, consider configuring a Custom CSV collection. Use the following list to determine if a Custom CSV collection might work for you:

- The data source must provide Comma-separated values (CSV) data. CSV data is the only method that OpsA provides to collect data (instead of those predefined or custom collection methods described in " Configuring Tenants and Collections " on page 17
- The data source must collect CSV data based on time. There has to be a time and date column for each row in the CSV file. Both the time and date must be in that same column.

  Note: If this is not true, you must merge these columns before creating the collection.

- The data source cannot exceed 1549 data columns. If you try to create a Customer CSV collection containing more than 1549 data columns, the collection creation will fail.
- The maximum supported CSV file size is 500 MB.
- Data from the CSV data source must be accessible to the OpsA Collector Appliance.
- The CSV file can be local or remote to the collector and is assumed to be available in the source directory at regular intervals.
- Each column should have a header.
- Column names should not contain any spaces.
- The CSV file will be used to create a table in Vertica. Vertica objects include tables, views, and columns. Your CSV file must use the following naming conventions:

- A column name must be from 1 to 128 characters long.
- A column name must begin with a letter (A through Z), diacritic marks, or non-Latin characters (200-377 octal).
- A name cannot begin with an underscore (_). Leading underscores are reserved for system objects.
- Names are not case sensitive. For example, `CUSTOMER` and `Customer` represent the same names. However, if you enclose a name in quotation marks, it is case sensitive.

  Note: Object names are converted to lowercase when they are stored in the Vertica database.

- A name cannot match a Vertica reserved word such as `WHERE`, `VIEW`, `Table`, `ID`, `User`, or `Query`.
- A name cannot match the another Vertica object that has the same type.
- The Maximum number of columns cannot exceed 1549 in a CSV file as that value is a Vertica limitation when creating a table.

- The CSV file format has to be uniform for a single collection. For example, if you create a collection with 10 columns, the subsequent files that are provided for import within OpsA must have same format, including column names and data types.

For an example of data you might choose to collect using a Custom CSV collection, see the *Creating a Content Pack for Operations Analytics* White Paper at https://hpln.hp.com.

Note: Do the following to locate the *Creating a Content Pack for Operations Analytics* White Paper:

1. Select **Products**.
2. Navigate to the **Operations Intelligence (OpsA)** product, then click **Operations Intelligence**.
3. Click **Resources**, then locate the *Creating a Content Pack for Operations Analytics* White Paper.

Important Prerequisite Steps

Complete the following prerequisite work before configuring your Custom CSV Collection using the steps in "Configuration Steps" on page 92:

1. Tenants enable you to separate information from these data sources into groups, such as collections, user accounts, and user groups. When running the commands in this chapter, the tenant model you select affects which Tenant Admin user you will use. Use one of the following tenant models:

   - **Default Tenant**: If you plan to use the default tenant, opsa_default, use opsatenantadmin as the tenant admin user and opsatenantadmin as the default tenant admin when running the commands in this chapter.

   - **Use your own Tenant**: If you plan to configure a new tenant or use an existing tenant (other than the Default Tenant), see "Creating Tenants " on page 20. If you use this option, you will need to use the tenant admin user and password you created when running the commands in this chapter.

2. For the Custom CSV Collection, your data must be available in CSV format. If your data is not available in CSV format, you must find a way to convert the data, or the Custom CSV Collection will not work for you.

3. Choose the `<filename>.csv` file you want to load into the OpsA database. For OpsA, consider that most `<filename>.csv` files for a CSV collection will have a CSV file with a header and at least one row of data. For this example, assuming this sample file name is `your_file.csv`, copy the `your_file.csv` file to the `/tmp` directory.

   For OpsA, consider that most `<filename>.csv` files for a CSV collection will have a CSV file with a header and at least one row of data. For example, the header could include two columns: one with data and one with the time and date.

   Note: The `your_file.csv` sample file contains a good sample of data. OpsA uses this sample data to determine the data types and meta data to place in the `<your_template_name>.xml` sample file used in these instructions.

   Do not include any of the values from the following table in the header, as Vertica does not permit these values to be used as header names

Reserved Words for Vertica (do not use these values in the header name)

Index Letter	Value
A	ALL, ANALYSE, ANALYZE, AND, ANY, ARRAY, AS, ASC
B	BINARY, BOTH
C	CASE, CAST, CHECK, COLUMN, CONSTRAINT, CORRELATION, CREATE, CURRENT_DATABASE, CURRENT_DATE, CURRENT_SCHEMA, CURRENT_TIME, CURRENT_TIMESTAMP, CURRENT_USER

Reserved Words for Vertica (do not use these values in the header name), continued

Index Letter	Value
D	DEFAULT, DEFERRABLE, DESC, DISTINCT, DO
E	ELSE, ENCODED, END, EXCEPT
F	FALSE, FOR, FOREIGN, FROM
G	GRANT, GROUP, GROUPED
H	HAVING
I	IN, INITIALLY, INTERSECT, INTERVAL, INTERVALYM, INTO
J	JOIN
K	KSAFE
L	LEADING, LIMIT, LOCALTIME, LOCALTIMESTAMP
M	MATCH
N	NEW, NOT, NULL, NULLSEQUAL
O	OFF, OFFSET, OLD, ON, ONLY, OR, ORDER
P	PINNED, PLACING, PRIMARY, PROJECTION
R	REFERENCES
S	SCHEMA, SEGMENTED, SELECT, SESSION_USER, SOME, SYSDATE
T	TABLE, THEN, TIMESERIES, TO, TRAILING, TRUE
U	UNBOUNDED, UNION, UNIQUE, UNSEGMENTED, USER, USING
W	WHEN, WHERE, WINDOW, WITH, WITHIN

4. Choose the following parameter values to use when running the `opsa-csv-template-gen.sh` script:

   - **name**: Choose a name that accurately describes the data you plan to collect. For example, you might choose the name `mycsv` for the source.
   - **domain**: Choose a domain that accurately describes a domain in which the data you plan to collect resides. For example, you might choose the domain `birds`, to support the example in this section.
   - **group**: Choose a group that accurately describes the group for which you plan to collect data. For example, you might choose the domain `eagle`, to support the example in this section.

Note: To specify a time zone that supports Daylight Savings Time, use the desired daylight savings time value you need as the `timezone` attribute. See "Daylight Savings Time Codes" on page 294 for a list of valid timezone attributes.

Note: You must run the `opsa-csv-template-gen.sh` script before running the `opsa-collection-setup.sh` script shown in the next section.

Note: The `opsa-csv-template-gen.sh` script uses the `-sourcedir <source_directory>` command option to configure the absolute path of the existing source directory for collection input files, That directory must exist on the OpsA Collector Appliance. Copy the `<yourdata>.csv` data files for the collection you plan to create into this directory.

Note: As a practical example, you might have a requirement to collect data from CSV files that keep generating periodically. For this example, suppose these files use some incrementing suffix, such as `abc1.csv`, `abc2.csv`, `abc3.csv` and so on. To meet this requirement, place these CSV files in the path you choose with the `-sourcedir <source_directory>` option when you run the `opsa-csv-template-gen.sh` script.

To expand this example, if you need to further limit the CSV file collections to use **only** the `abc1.csv`, `abc2.csv`, `abc3.csv` and so on files (ignoring the other files in the <*source_directory*>), use the `-filepattern <yourfilepattern>` option when you run the `opsa-csv-template-gen.sh` script. For this example, you might run the `opsa-csv-template-gen.sh` script along with the `-filepattern abc*.csv` and the `-sourcedir /opt/HP/opsa/data/mydata` options to limit this CSV collection to read **only** the `abc1.csv`, `abc2.csv`, `abc3.csv` and so on files from the `/opt/HP/opsa/data/mydata` directory.

The following is an example of the `opsa-csv-template-gen.sh` command with all of the options shown: `opsa-csv-template-gen.sh -inputfile <Full path to the sample CSV> -name <Name of the collection> -domain <Domain Name> -group <Group Name> -sourcedir <Folder path to where the input CSVs will be located > -datecolumn <Name of the Date column in the CSV> -dateformat <Date Format of the date> -timezone <GMT of the data source> -filepattern <Only files that comply with this pattern will be processed> -grouptype <metrics or attributes> -key <Define key columns>`

See the *opsa-csv-template-gen.sh* reference page (or the Linux manpage), for more information.

Configuration Steps

Gather the following information to prepare for configuring the Custom CSV Collection.

Information to Collect Before Running the opsa-collection-setup.sh Script

Input Requested by opsa-collection-setup.sh Script	Value
Domain Name	Choose a domain that accurately describes a domain in which the data you plan to collect resides.
Group Name	Choose a group that accurately describes the group for which you plan to collect data.
Valid Custom CSV Template	The template file name for the file you generated in the prerequisites section (using the `opsa-csv-template-gen.sh` script).

After you complete the steps in this section, the Custom CSV Collection reads data from the CSV files within 60 seconds of the file being placed in the source directory.

For several examples of data you might choose to collect using a Custom CSV collection, see the *Creating a Content Pack for Operations Analytics* White Paper at https://hpln.hp.com.

Note: Do the following to locate the *Creating a Content Pack for Operations Analytics* White Paper:

1. Select **Products**.
2. Navigate to the **Operations Intelligence (OpsA)** product, then click **Operations Intelligence**.
3. Click **Resources**, then locate the *Creating a Content Pack for Operations Analytics* White Paper.

1. Run the interactive `opsa-collection-setup.sh` script.
2. When prompted, enter the tenant admin user name and password.
3. Enter `11` to begin configuring a Custom CSV Collection.
4. Enter `add` to configure a new Custom CSV instance.
5. Follow the prompts to enter the requested information.
6. Enter `back` when you are finished.
7. Enter `execute 11` to deploy this collection to the collector host.
8. To check for success, enter `list` and check that the Custom CSV sources were added to the collector host.
9. Let the collection run for five minutes or longer. From the Operations Analytics console, view the

OpsA Meta Info dashboard. Look for the **property group uid** for the collection you just created and published.

Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection.

10. If you want to add tags to a Custom CSV Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Troubleshooting the Custom CSV Collection

If you suspect problems with your Custom CSV Collection, do the following:

1. To check the registration status of your collector appliance, do the following:
   a. `$OPSA_HOME/bin/opsa-collection-config.sh -list -collectorhosts -allversions -username opsatenantadmin`

   b. Review the list of registered collectors. If the collector you plan to register is not on the list, you must register it using the instructions in this section.

2. View the collected data to make sure it is what you expect. If it is not, continue checking the remaining items in this list.

3. Review the content of the `your_file.csv` file and the associated `<your_template_name>.xml` file to make sure it is configured to collect the right data.

4. You must use a CSV file for the Custom CSV Collection. Check the `<filename>.csv` file you loaded into the OpsA database. For OpsA, consider that most `<filename>.csv` files for a CSV collection will have a CSV file with a header and at least one row of data.

5. Check the quality of the data you are collecting. If it is not what you expected, review the content of the `<filename>.csv` file you loaded into the OpsA database, as it might not be collecting the right data for you.

Removing the Registration and Data for a Custom CSV Collection

To remove the registration for a Custom CSV Collection, do the following:

1. Run the following command:
   `$OPSA_HOME/bin/opsa-collection-config.sh -unregister -source custom -group group -domain domain -collectorhost <fully-qualified domain name of collector host>`

Note: If you remove the registration for this CSV collection, and do not complete the remaining steps, remember the following important information:

- The collected data remains intact and is not removed.
- If you decide to register this collection again, you must not reuse the `your_file.csv` file, (or whatever csv file name you used to create the collection template), as you run the risk of duplicating the original collection data.
- It is a best practice to complete all of these removal steps to avoid collecting duplicate data.

2. After unregistering this Custom CSV Collection, remove the collection from the database using the following command:
`$OPSA_HOME/bin/opsa-collection-config.sh -purgecollection -source custom -domain domain -group group -collectorhost <fully-qualified domain name of collector host> -username opsatenantadmin`
See the *opsa-collection-config.sh* reference page (or the Linux manpage), for more information.

Note: The command in this step also removes all Custom CSV Collection data for the specified tenant from the OpsA database.

Note: After unregistering a Custom CSV Collection, the data remains intact. This means that you can register a Custom CSV Collection that you removed and resume that Custom CSV Collection.

3. Remove the data. For example, for the NOAA example, you would remove the `$OPSA_HOME/data/noaaCustom_processed` directory.
4. Remove the data. For example, for the NOAA example, you would remove the `$OPSA_HOME/data/noaaCustom_processed` directory.

See the *opsa-collection-config.sh* reference page (or the Linux manpage) and "Removing a Collection Registration for a Tenant" on page 170 for more information.

Configuring a Custom SiteScope Collection

Note: If you do not want to configure this collection by using the **Collections Manager** located in the Operations Analytics console, you can configure this collection using the steps in this section.

After you complete the steps in this section, SiteScope starts sending data to the Custom SiteScope Collection. The Custom SiteScope Collection collects data as it arrives from SiteScope.

The following table shows the monitor types currently supported by the Custom SiteScope Collection:

Note: If a SiteScope monitor type has only unsupported counters configured, OpsA ignores that

monitor type when creating the collection. OpsA does not support monitor counter names longer than 128 characters. If a supported monitor's counter name is longer than 128 characters, OpsA ignores that counter.

Supported Monitor Types

Apache	Port
BACIntegrationConfiguration	SAPPerformance
BACIntegrationStatistics	Script
Composite	Service
ConnectionStatisticsMonitor	SiebelApplicationServer
CPU	SolarisZones
DatabaseCounter	SQLQuery
DHCP	SSLCertificatesStatus
Directory	Sybase
DiskSpace	UnixResources
DNS	URLContent
DynamicDiskSpace	URLMonitor
File	URLSequenceMonitor
FTPMonitor	VMware
HealthServerLoadMonitor	VMwareHostCPUMonitor
HyperVMonitor	VMwareHostMemoryMonitor
JMXMonitor	VMwareHostStateMonitor
LDAPMonitor	VMwareHostStorageMonitor
LogEventHealthMonitor	WebServer
LogMonitor	WebService
Memory	WebSphere
MicrosoftWindowsEventLog	WindowsPerformance
MQStatusMonitor	WindowsResources

Supported Monitor Types, continued

MSActiveServerPages	WindowsServicesState
MSIISServer	XMLMetrics
MSSQLServer	
MSWindowsMediaServer	
NetworkBandwidthMonitor	
Oracle	
Ping	

Note: You must set the `max heap size` to 3 GB or higher on the Operations Analytics Server Appliance when OpsA collects data from more than 1000 SiteScope monitors. To set the `max heap size`, do the following:

1. Edit the `/opt/HP/opsa/jboss/bin/standalone.conf` file.
2. Make the change shown in bold font in the in JAVA_OPTS section:

```
#
# Specify options to pass to the Java VM.
if [ "x$JAVA_OPTS" = "x" ]; then
JAVA_OPTS="-Xms64m -Xmx3072m -XX:MaxPermSize=256m -
Djava.net.preferIPv4Stack=true"
JAVA_OPTS="$JAVA_OPTS -Djboss.modules.system.pkgs=$JBOSS_MODULES_
SYSTEM_PKGS -Djava.awt.headless=true"
else
echo "JAVA_OPTS already set in environment; overriding default
settings with values: $JAVA_OPTS"
fi
```

3. Save your changes.
4. Run the following command to restart the OpsA Server Appliance:

```
$OPSA_HOME/bin/opsa-server restart
```

   See the *opsa-server* reference page (or the Linux manpage) for more information.

Gather the following information to prepare for configuring the Custom SiteScope Collection.

Information to Collect Before Running the opsa-collection-setup.sh Script

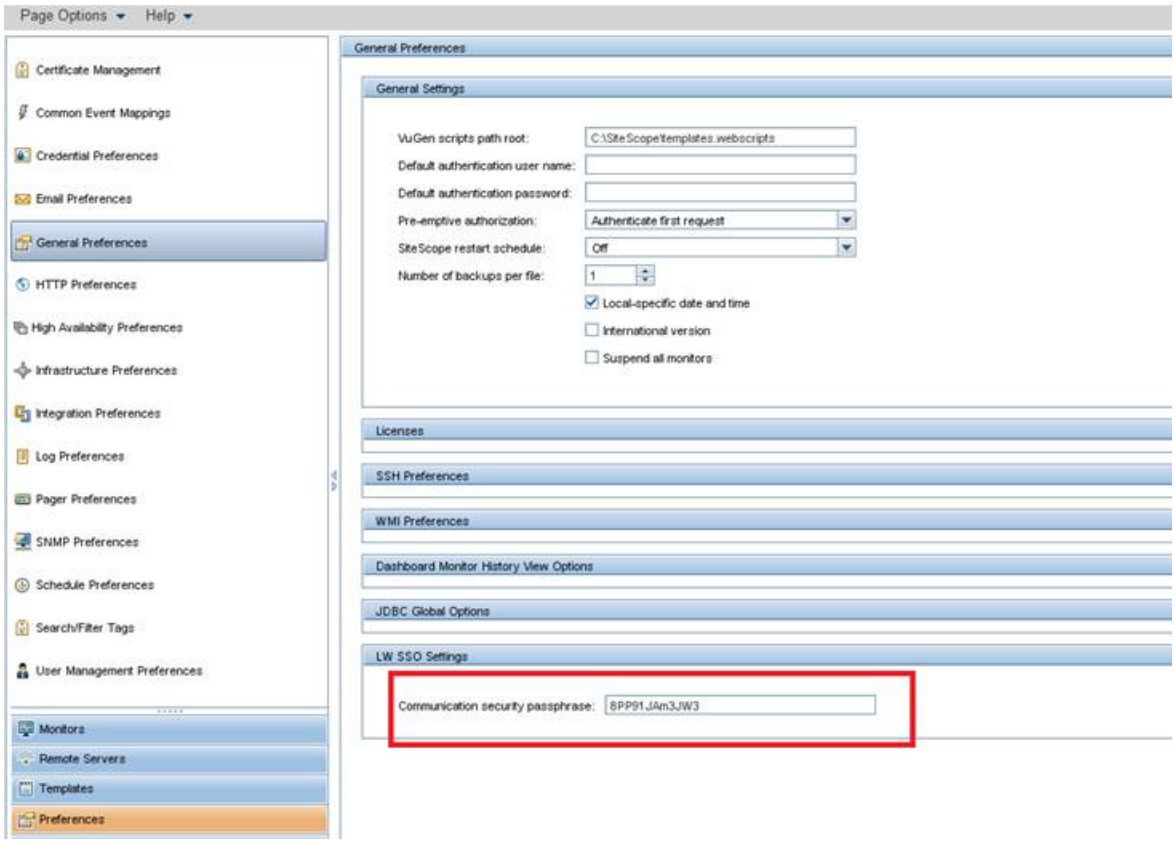
Input Requested by opsa-collection-setup.sh Script	Value
OpsA Collector Node	The fully-qualified domain name or the IP address of the common collector that will collect data from the SiteScope servers. Do not use `localhost` or `127.0.0.1`.
SiteScope Server Hostname	The IP Address or fully-qualified domain name of the SiteScope server for which you are configuring a collection.
SiteScope Server LW SSO Token	The default value of the initstring used for SSL communication with the SiteScope server. **You can obtain this initString from the SiteScope screen shown below this table.**
SiteScope Server Password	The password for the user name.
SiteScope Server Port Number	.The port used to connect to the SiteScope server. Set this if a server does not use the default.port value. The default is 8080.
SiteScope Server Username	The default user name used to connect to the SiteScope server. This is typically `admin`. This field might be set to empty (no value).
Use Collector IP Address (true/false)	When set to `true`, the OpsA integration URL (this link resides in SiteScope) uses the OpsA Collector Appliance IP address and not the OpsA Collector Appliance hostname.

Information to Collect Before Running the opsa-collection-setup.sh Script, continued

Input Requested by opsa-collection-setup.sh Script	Value
Use SSL (true/false)	Set this field to `true` to enable SSL communication with the SiteScope server. If you set this value to `true`, you must export the certificate from the OpsA Collector Appliance and import this certificate on each SiteScope server. If SiteScope is enabled with SSL, do the following: 1. Copy the SiteScope server's root server certificate to each OpsA Server and give the file full permissions. 2. Run the following command: `keytool -importcert -alias <certificate alias> -file <certificate file> -keystore /opt/HP/opsa/jdk/jre/lib/security/cacerts` See *Configuring SiteScope to Use SSL* in the *HP SiteScope Deployment Guide* and the *opsa-collector-manager.sh* reference page for more information.

Finding the initstring in SiteScope

Note: For the integration with OpsA to work correctly, you must enter the communication security passphrase (LW SSO Token) as shown in the example in following graphic. If this field is empty, see the SiteScope documentation for the instructions to set a value in this field.



Follow the instructions in this section to configure a Custom SiteScope Collection for Operations Analytics.

Note: Configuring the Custom SiteScope Collection by using the `opsa-collection-setup.sh` script shown in this section (or by using the Collections Manager in the OpsA console) automatically tags the root group in SiteScope in a way that data from all monitors will be sent to OpsA.

If you want to manually select (tag) the monitors from which to receive data, (instead of using the `opsa-collection-setup.sh` script or the Configuration Manager in the OpsA console), you must do the following:

1. Follow the instructions in"Configuring a Custom SiteScope Collection (Detailed Method)" on page 281 and use the `-ignoretag` option when running `opsa-sis-collector-auto-conf.sh` script.
2. Follow the steps in the *Configuring SiteScope for Integrating Data with Operations Analytics (Manual Method)* section in this manual to manually tag the integration and the desired monitors.

1. Run the interactive `opsa-collection-setup.sh` script.
2. When prompted, enter the tenant admin user name and password.

3. When prompted, enter the index of the collector host for which to configure SiteScope.
4. Enter `10` to begin configuring a Custom SiteScope Collection.
5. Enter `add` to configure a new SiteScope instance.
6. Follow the prompts to enter the requested information.
7. Enter `back` when you are finished.
8. Enter `execute 10` to deploy this collection to the collector host.
9. To check for success, enter `list` and check that the Custom SiteScope sources were added to the collector host.
10. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

    Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection.

11. If you want to add tags to a Custom SiteScope Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Configuring a Structured Log Collection

Structured logs are fragments of log file data read by Operations Analytics (OpsA) from HP ArcSight Logger. This log information is stored (as collections) in OpsA. These collections exist so that users can perform analytics on the log file contents. For example, users might want to query for all outliers by host name and application for a particular time range.

Note: To configure Log Analytics to work with Logger or Splunk collections, see *Configuring Log Analytics* in the *Operations Analytics Quick Start Guide*.

After you complete the steps in this section, the Structured Log Collection collects data every 5 minutes.

Configuring Logger to Forward CEF Messages to Operations Analytics

The feature being configured in this section is also known as the **TCP Forwarding** feature. After you complete the configuration instructions in this section, the performance of the Collector appliance significantly improves. You will also observe more real-time log messages in OpsA.

Note: The TCP Forwarding feature does not support a secured connection between Logger and the OpsA Collector Appliance.

Note: The configuration shown in this section changes the default way that Logger passes data to the OpsA Collector Appliance. By default the OpsA Collector Appliance pulls data from Logger (instead of Logger pushing it to the OpsA Collector Appliance).

If you want Logger to forward near real-time CEF messages to OpsA configure do the following:

Note: If you complete the instructions in this section, you can configure OpsA to receive messages from Logger:

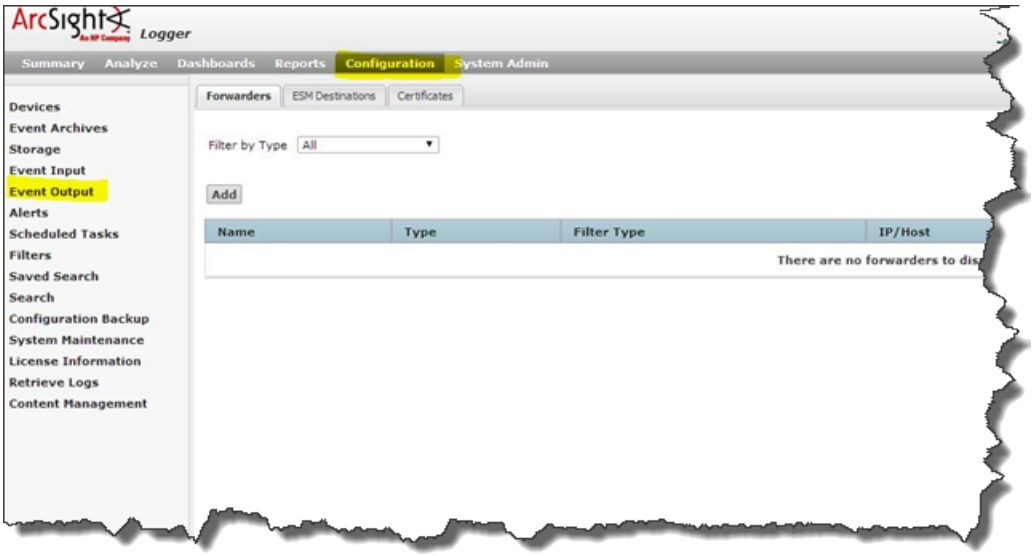
- You will be able to use the `-passive` mode with the `opsa-collection-config.sh` script when publishing a structured log collection. The `-passive` mode is used to support OpsA's out of the box log collections.

  For example, after configuring this feature, you can use a command similar to the following when publishing your collection (notice the bold **-mode passive** option):
  `opsa-collection-config.sh -create -nodelist /tmp/arcsight-log-stream.properties -collectorhost <collector-host> -source arcsight -domain log -group stream -username opsatenantadmin` **-mode passive**

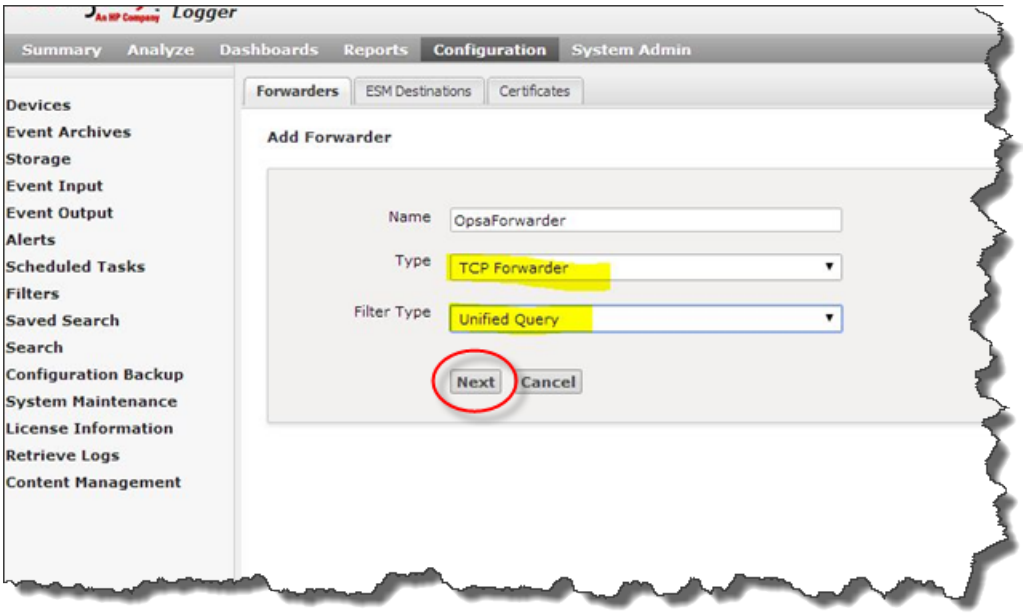
- You will not be able to use the `-active` mode with the `opsa-collection-config.sh` script when publishing a structured log collection.

  For example, after configuring this feature, **you cannot use a command similar to the following** when publishing your collection (notice the bold **-mode active** option):
  `opsa-collection-config.sh -create -nodelist /tmp/arcsight-log-stream.properties -collectorhost <collector-host> -source arcsight -domain log -group stream -username opsatenantadmin` **-mode active**

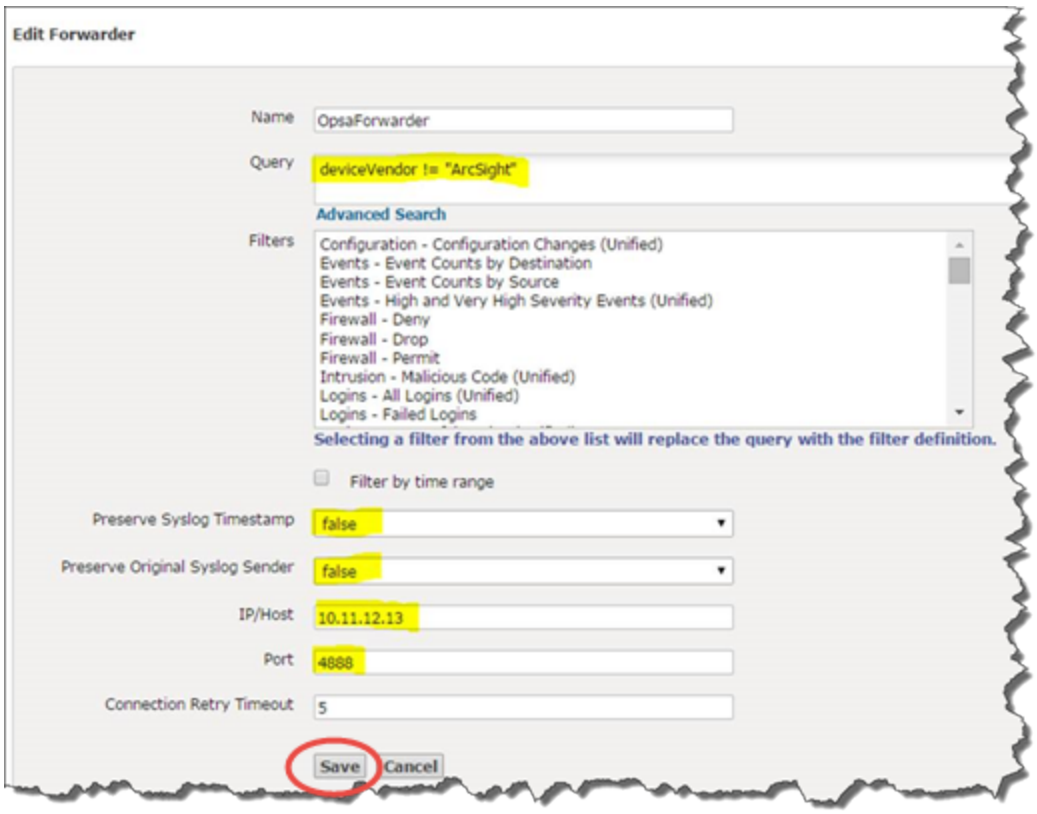
1. From Logger, navigate to **Configuration** > **Event Output**.



2. Select the **Forwarders** tab; set the values following the example highlighted below; then click **Next**:



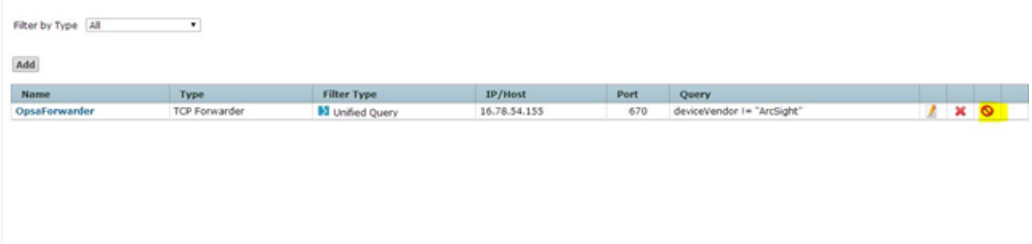
3. Enter the values following the example highlighted below; then click **Save**:

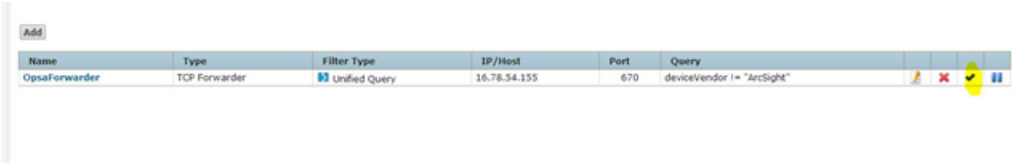


Note:

- The query highlighted above is for HP ArcSight Logger. Make sure the query you configure represents the collection you plan to configure.
- Change the highlighted IP address (in the **IP/Host** field) to the IP address of the OpsACollector Appliance for the collection you plan to configure.

4. Enable the new configuration by clicking the highlighted area as follows:





5. Now Logger should forward near real-time CEF messages to OpsA.

> **Note:** If you completed these instructions, you must use the `-mode passive` option with the opsa-collection-config.sh script when publishing the associated structured log collection. See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

Configuring the Maximum Logger Sessions

By default the `logger.max.sessions` property is set to a value of 5 in an OpsA Collector Appliance and 25 for an OpsA Server Appliance . This means there can be a maximum of 5 Logger session per Logger host in an OpsA Collector Appliance and 25 Logger sessions per Logger host in an OpsA Server Appliance.

To set the maximum number of Logger sessions for the OpsA Collector and Server Appliances, do the following on both the OpsA Collector Appliance and the OpsA Server Appliance:

1. Edit the `$OPSA_HOME/conf/opsa-config.properties` file.
2. Set the `logger.max.sessions` property to the desired value.

> **Note:** The sum of the values you set in the OpsA Collector and Server Appliances must not exceed 30.

3. Save your work.

4. If you changed the `logger.max.sessions` property on the OpsA Server Appliance, restart the `opsa-server` service by running the following command from the OpsA Server Appliance:
   `$OPSA_HOME/bin/opsa-server restart`

   See the *opsa-server* reference page (or the Linux manpage) for more information.

5. If you changed the `logger.max.sessions` property on the OpsA Collector Appliance, restart the `opsa-collector` service by running the following command from the OpsA Collector Appliance:
   `$OPSA_HOME/bin/opsa-collector restart`

   See the *opsa-collector* reference page (or the Linux manpage) for more information.

If you have two or more OpsA Server Appliances configured in a distributed environment, you must spread the available 25 HP ArcSight Logger sessions across both OpsA Server Appliances. If you do not configure these session values correctly, one appliance might control all of the sessions, while the remaining appliance cannot control any sessions.

Note: If you are not planning to configure structured log collections, then set the `logger.max.sessions` property on the OpsA Server Appliance to 30. Doing so enables OpsA to use all of the Logger sessions for rawlog searches in the Operations Analytics console.

Note: As mentioned above, from a resource perspective, there is a limit to the number of Logger sessions supported by HP Operations Analytics Software. HP strongly recommends that, when you configure Logger collections, you assign those Logger collections to one common OpsA collector. Doing so reduces the number of Logger sessions.

Configuring a Structured Log Collection

To configure an OpsA Structured Log Collection, do the following:

1. Using HP ArcSight Logger, define the search query to determine the data you want to collect. For example, you might create the following search query in HP ArcSight Logger based on the ArcSight's WebLogic SmartConnector:
   ```
   agentType = "weblogic_multi_file" AND deviceVendor CONTAINS "Oracle"
   | fields + startTime agentHostName sourceHostName name bytesIn
   bytesOut deviceAction requestMethod requestUrl
   ```

2. A node list file contains details about the sources from which you plan to collect information. There are sample nodelist files located in the `$OPSA_HOME/conf/collection/sample` directory. OpsA administrators can use these sample files to publish the node list file. The sample node list file for the Structured Log collection is either `sample_ArcSight_node.properties` or `sample_Splunk_node.properties`.

Complete the following steps from the OpsA Server Appliance for the Structured Log collection:

a. Copy the appropriate node list file from the `$OPSA_HOME/conf/collection/sample` directory to some location, such as `/tmp/mynodelist.properties`

> **Note:** Select the template file pertaining to the type of collection you are configuring.

b. Edit the `/tmp/mynodelist.properties` file; add information according to what is written in the sample file; then save your work.

HP ArcSight Logger: For example, using ArcSight's WebLogic SmartConnector example shown earlier, you would specify the HP ArcSight Logger hostname and search query:

```
server.names = arcsightserver
##node properties for 'Arcsight'
arcsightserver.hostdnsname = <fully-qualified domain name of the
HP ArcSight Logger server>
```
arcsightserver.query = agentType = \"weblogic_multi_file\" AND deviceVendor CONTAINS \"Oracle\" | fields + startTime agentHostName sourceHostName name bytesIn bytesOut deviceAction requestMethod requestUrl

> **Note:** Although not shown in this example, always use `deviceReceiptTime` as a field in the `mynodelist.properties` file.

Below are some helpful steps to help configure information in the bold font shown above:.

i. Confirm that HP ArcSight Logger is receiving the messages you expect.

ii. Verify that HP ArcSight Logger is processing the log messages into the correct fields. For example, make sure that the `agent_severity` and `message` fields are being populated as expected. If HP ArcSight Logger is not parsing the messages into fields properly, then you might need to correct the configuration for the connector, receiver, or parser. See the *HP ArcSight Logger Administrator's Guide* for more information.

iii. Use the HP ArcSight Logger Analyze/Search facility to fine tune your row selection. This corresponds to the configuration entries that reside before the bar character (|). HP ArcSight Logger has a powerful parsing mechanism. You can tune HP ArcSight Logger to choose the logs messages that interest you while ignoring those messages that are not interesting. HP ArcSight Logger tuning is important, as many of the HP ArcSight Logger receivers can receive logs from multiple sources.

iv. The configuration entries that reside before the bar character (|) that you add in this step select the data (rows) to be collected.

v. The text following the `fields +` keyword (after the bar character (|) that you add in this step) sets the column names. After you are satisfied with the messages, work on the fields. In HP ArcSight Logger, add `| fields + F1 F2 F3` to select the columns you

would like to send to OpsA. You can do all this experimenting in HP ArcSight Logger.

vi. Test the entire string from this step in the HP ArcSight Logger Analysis Search and adjust the string for the desired results before continuing.

> **Note:** You must remove the \ characters before testing the string in the HP ArcSight Logger.

vii. When you are satisfied after working with these tuning tips, place the entire search expression in the `/tmp/mynodelist.properties` file. You must backslash any quotes you used.

Splunk: If you are using Splunk instead of HP ArcSight Logger, specify the following information in the `/tmp/mynodelist.properties` file:

```
server.names = splunkserver
##node properties for 'splunk'
splunkserver.hostdnsname = opsasplunk.fc.usa.hp.com
splunkserver.port = 8089
splunkserver.username = admin
splunkserver.username = opsa
splunkserver.use_ssl = false
splunkserver.encryptionStatus = true
```

c. Save your work.

3. Run the following command from the OpsA Server Appliance if you think there might be an existing structured log collection template you can use. Running this command shows you the available predefined templates: `$OPSA_HOME/bin/opsa-collection-config.sh -list -templates -username opsatenantadmin`

4. If there is no existing structured log collection template, do the following from the OpsA Server Appliance to create one:

   a. Review the following HP ArcSight Logger collection templates:
   `/opt/HP/opsa/conf/collection/sample/config.templates/arcsight/1.0/apache/access/apache_access.xml`
   `/opt/HP/opsa/conf/collection/sample/config.templates/arcsight/1.0/log/structuredlog/arcsight_collection.xml`
   `/opt/HP/opsa/conf/collection/sample/config.templates/splunk/1.0/log/structuredlog/splunk_collection.xml`

   b. Copy one of these templates to a temporary location; then edit the file to create the collection template you need for your structured log collection. Suppose that, for this example, we call this file `mystructuredlog.xml`.

Note: Considering the ArcSight's WebLogic SmartConnector example shown earlier, you might run the following command:

```
cp
/opt/HP/opsa/conf/collection/sample/config.templates/arcsight/1.
0/log/structuredlog/arcsight_collection.xml
/tmp/mystructuredlog.xml
```

c. Edit the `mystructuredlog.xml` file:

i. Considering the ArcSight's WebLogic SmartConnector example shown earlier, you might change the `domain`, `tags`, `group`, and `label` attributes for the `collectiongroup` elements as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<collectiongroup domain="weblogic"
tags="log,arcsight,weblogic,access" group="access" group_
type="log" label="WebLogic Access Log">
<collector type="arcsight" version="5.5.0"
collectionintervalinseconds="300">
<sourcegroup name="default" granularityinseconds="300">
<source name="arcsightQuery" value="" type="query" />
</sourcegroup>
</collector>
```

Note: Although not shown in this example, if you see a `mapsto` item in your collection template file, note its value, as it shows the associated column name in HP ArcSight Logger. See the following table for more information.

Note: If you are using Splunk as your logger application you do not need to edit the `splunk_collection.xml` file.

Mapping a Column Name to Attribute Values (Examples)

Column Name	Attribute Value
timestamp	deviceReceiptTime
agentHostName	agentHostName
sourceHostName	sourceHostName
name	name
bytesIn	bytesIn
bytesOut	bytesOut

Mapping a Column Name to Attribute Values (Examples), continued

deviceAction	deviceAction
requestMethod	requestMethod
requestUrl	requestUrl

ii. Save your work.

d. Copy the `mystructuredlog.xml` file to `/opt/HP/opsa/conf/collection/server/config.templates/<arcsight | splunk>/<version from template file><domain from template files>/<group from template files/mystructuredlog.xml`

Considering the ArcSight's WebLogic SmartConnector example shown earlier, you might copy the `mystructuredlog.xml` file to a new weblogic folder:

```
/opt/HP/opsa/conf/collection/server/config.templates/arcsight/1.0/weblogic/access
```

5. Run the following command from the OpsA Server Appliance to create the collector configuration:
`$OPSA_HOME/bin/opsa-collection-config.sh -create -nodelist /tmp/mynodelist.properties -collectorhost <fully-qualified-domain-name of collector host> -source splunk|arcSight -domain <domain from template files> -group <domain from template files> -username opsatenantadmin`

Considering the ArcSight's WebLogic SmartConnector example shown earlier, you would run the following command:
`$OPSA_HOME/bin/opsa-collection-config.sh -create -nodelist /tmp/mynodelist.properties -collectorhost <fully-qualified domain name of the collector server> -source arcSight -domain weblogic -group access -username opsatenantadmin`

Note: The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

Look for a success message similar to the following:

```
Successfully created the collectorhost '<fully-qualified domain name of the collector server>' configuration.
<fully-qualified domain name of the collector server> base directory:
/opt/HP/opsa/conf/collection/config.files/<fully-qualified domain name of the collector server>/opsa_
default/1.0/arcsight/1.0/weblogic/access
Successfully published the node list for this collector host.
```

6. Check the `$OPSA_HOME/log/collection_config.log` file (or `opsa.log` file) for errors. Correct these errors before continuing.

7. Run the following command from the OpsA Server Appliance to validate the collection configuration you just created:

   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -list
   -allversions -collectorhosts -username opsatenantadmin
   ```

   To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its version, domain, and group.

   If you encounter any errors, look in the `/opt/HP/opsa/log/collection_config.log` file and review the logs carefully to understand and fix any errors.

8. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance:
   `$OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost <fully-qualified domain name of the collector host> -username opsatenantadmin -mode active|passive`

   Note: If you followed the instructions in "Configuring Logger to Forward CEF Messages to Operations Analytics" on page 101, you configured Logger to send CEF message to OpsA and must use the `-mode passive` option in this command. If you want OpsA to actively request log information (the original product behavior), use the `-mode active` option (the default option) in this command.

   For example, you can use a command similar to the following when publishing your collection (notice the bold **-mode passive** option):
   `$OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost <mycollector.company.com> -username opsatenantadmin` **-mode passive**

   See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

   The `-publish` option uploads the collection configuration you created to the OpsA Collector Appliance. To verify that the collector configuration published successfully, look for a message stating that the publish was successful and that a table was successfully created.

   Considering the ArcSight's WebLogic SmartConnector example shown earlier, you might see something similar to the following:

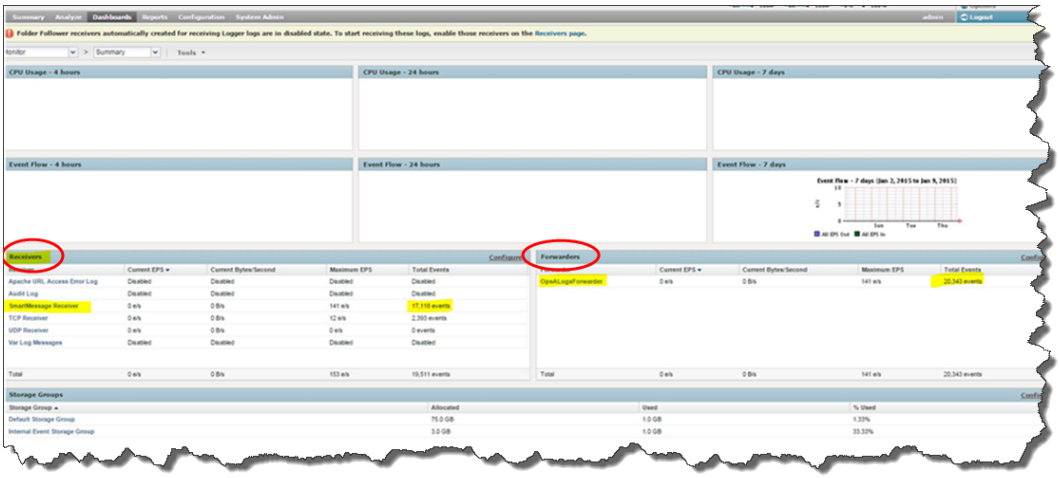
   ```
   Creating the collection database tables for the source:arcsight
   domain:weblogic group:access and tenant:opsa_default
   Successfully created table using
   ```

```
/opt/HP/opsa/conf/collection/config.files/<fully-qualified domain name of the
collector host>/opsa_
default/1.0/arcsight/1.0/weblogic/access/metaData.xml for tenant
opsa_default
Registering the collection policy for the source:arcsight
domain:weblogic group:access and tenant:opsa_default into the
database
Successfully registered collection policy for source
collector:arcsight tenant:opsa_default- 1.0 Domain:weblogic
Group:access
Registering the list of sources for the source:arcsight
domain:weblogic group:access and tenant:opsa_default into the
database
Successfully registered nodes for <fully-qualified domain name of the
collector host>-opsa_default-weblogic-access in the Operations
Analytics database
```

If you encounter any errors, look in the `/opt/HP/opsa/log/collection_config.log` file and review the logs carefully to understand and fix any errors.

9. Do the following to verify that the collection is working:

   a. Open **Arcsight Logger** > **Dashboards** and notice the **Receivers** and **Forwarders** panes.



   b. Review the information shown in the **Receivers** pane to determine if Logger is receiving any CEF messages from the smart connectors.

   c. Review the information shown in the **Forwarders** pane to determine if Logger is forwarding the CEF messages to OpsA Collector Appliance.

   d. Do the following to verify that the OpsA Collector Appliance is listening for log messages:

Since the OpsA Collector Appliance is listening on port 4888, use your favorite command to check if the port is open. For example, you might run `netstat -a | grep 4888` to see if the OpsA Collector Appliance is successfully listening on port 4888.

10. Look in the `/opt/HP/opsa/log/loader.log` file to see that it is processing the contents of the data being collected. Considering the weblogic example shown earlier, you might see something similar to the following:

```
2014-02-15 15:16:53 DEBUG [pool-1-thread-19] LoadDataCmd:512 -
archive file :/opt/HP/opsa/data/archive/opsa_
default/data~~arcsight~~weblogic~~access~~-2014-02-15_15-16-
49.782.csv
2014-02-15 15:16:53 DEBUG [collection data dir watcher]
DataLoader:240 - received notification
for/opt/HP/opsa/data/load/opsa_
default/data~~arcsight~~weblogic~~access~~-2014-02-15_15-16-
49.782.csv
```

Note: You can also test for success in several other ways:

Use a database management software tool to see if the table has been created, and that it is being populated with the expected columns.

If you do not see the table, check to see that the csv data files are automatically created for you on the OpsA Collector Appliance. Look in the following directories:

- `$OPSA_HOME/data/load/opsa_default`
- `$OPSA_HOME/data/archive/opsa_default`

When viewing these data files, if you see the columns you expect, but no rows, you might need to correct the configuration for the connector, receiver, or parser.

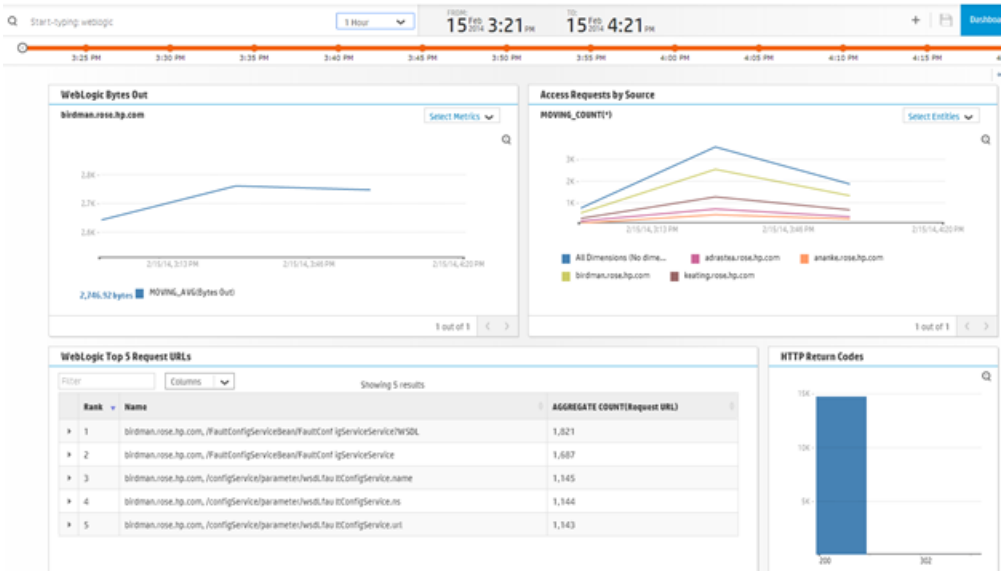
11. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published. **Note**: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. Considering the ArcSight's WebLogic SmartConnector example shown earlier, you used a name of `arcsight`, a domain of `weblogic`, and a group of `access` when creating the collection. The resulting property group uid would be `arcsight_weblogic_access`.

Note: OpsA does not display events, such as binary events, that contains unprintable characters.

12. Create dashboards and query panes for the data you are now collecting. Follow the instructions

shown in the *Dashboards and Query Panes* section of the *Operations Analytics Help* for information about creating dashboards and query panes.

Considering the ArcSight's WebLogic SmartConnector example shown earlier, you might create a dashboard similar to the following:



13. Create AQL functions for the data you are now collecting. See the instructions shown in the *Define Analytic Query Language (AQL) Functions* section of the *Operations Analytics Help* for information about creating AQL functions.

    Considering the weblogic example shown earlier, you might create the following AQL functions:

    WebLogic Bytes Out

    ```
    from i in (arcsight_weblogic_access) let analytic_interval=between
    ($starttime, $endtime) let interval=$interval group by
    i.agenthostname select moving_avg(i.bytesout)
    ```

    Access Requests by Source

    ```
    from i in (arcsight_weblogic_access) let analytic_interval=between
    ($starttime, $endtime) let interval=$interval group by
    i.sourcehostname select moving_count(i)
    ```

    WebLogic Top 5 Request URLs

    ```
    from i in (arcsight_weblogic_access) let analytic_interval=between
    ($starttime,$endtime) let interval=$interval select i.agenthostname,
    i.requesturl, topN(aggregate_count(i.requesturl),5)
    ```

    HTTP Return Codes

    ```
    from i in (arcsight_weblogic_access) let analytic_interval=between
    ($starttime,$endtime) let interval=$interval group by i.deviceaction
    select aggregate_count(i.deviceaction)
    ```

Note: OpsA processes data field strings used by the Structured Log Collection. There is a limit to

the length of data field strings OpsA can process. Although rare, if OpsA cannot process a complete data field string, it trims the string to a length that it can successfully process.

Note: If you stop a Structured Log Collection (or it stops collecting data for any reason) for an extended period of time, and you restart the collection, OpsA gradually recovers the data from HP ArcSight Logger. OpsA recovers the last five hours of data from the time of the restart.

Chapter 4: Creating, Applying, and Maintaining Tags

Operations Analytics (OpsA) supports three types of tags:

- Property Group Tags: OpsA administrators add these tags to an entire collection.
- Link Tags: Link Tags are special tags used to relate collection information. OpsA administrators add these tags to define a link between collections, creating the same link tag for each collection they want to link together.
- Property Tags: OpsA administrators add these link tags to one or more properties (or database columns) for a specific collection.

To manage tags, use the opsa-tag-manager.sh script. See the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Note: Tags, property uids, and property group uids are not case sensitive. They are always converted into lowercase.

Adding Tags

Use the following command to add tags:

- **Property Tags**: `$OPSA_HOME/bin/opsa-tag-manager.sh -add_tags -type property -file /opt/HP/opsa/tmp/property_tags.csv -username <opsatenantadmin>`
- **Property Group Tags**: `$OPSA_HOME/bin/opsa-tag-manager.sh -add_tags -type property_group -file /opt/HP/opsa/tmp/property_group_tags.csv -username opsatenantadmin`
- **Link Tags**: `$OPSA_HOME/bin/opsa-tag-manager.sh -add_tags -type link -file /opt/HP/opsa/tmp/link_tags.csv -username opsatenantadmin`

Listing Tags

Use the following command to list tags:

- **Property Tags**: `$OPSA_HOME/bin/opsa-tag-manager.sh -list_tags -type property [-propertygroup_id ID] [-property_id ID] -username opsatenantadmin`

- **Property Group Tags**: `$OPSA_HOME/bin/opsa-tag-manager.sh -list_tags -type property_group [-propertygroup_id ID] -username opsatenantadmin`
- **Link Tags**: `$OPSA_HOME/bin/opsa-tag-manager.sh -list_tags -type link -username opsatenantadmin`

Deleting Tags

Do not delete any pre-existing tags used for pre-defined collection templates, as that might disrupt these collections.

Use the following command to delete tags:

- **Property Tags**: `$OPSA_HOME/bin/opsa-tag-manager.sh -delete_tags -type property -propertygroup_id <property group id> -tag_name <list of comma-separated tags> -username opsatenantadmin`
- **Property Group Tags**: `$OPSA_HOME/bin/opsa-tag-manager.sh -delete_tags -type property_group -propertygroup_id <property group id> -tag_name <list of comma-separated tags> -username opsatenantadmin`
- **Link Tags**: `$OPSA_HOME/bin/opsa-tag-manager.sh -delete_tags -type link -propertygroup_id <property group id> -rel_propertygroup_id <property group uid of source> -username opsatenantadmin`

Chapter 5: Communicating Collection Names and Meta Data Information to your Users

One way for operators to view the tags and property groups available to them is to View the **OpsA Meta Info** dashboard, which displays all of the active collections and the tags being used. The information in this dashboard provides operators with a lot of the information they need for more effective queries. See *Dashboards Provided by Operations Analytics* in the *Operations Analytics Help* for more information.

The following example uses the `$OPSA_HOME/bin/opsa-tenant-manager.sh` script. This example uses a predefined Super Admin user (opsaadmin) and the password for this user that you set during installation.

To create a list of the collections and tags your users will be interested in, you can also do the following:

1. To list all of the tenants configured for an Operations Analytics (OpsA) Server Appliance, run the `$OPSA_HOME/bin/opsa-tenant-manager.sh -list -loginUser opsaadmin -loginPassword <password>` command from the OpsA Server Appliance. Make a list of the tenants shown in the command output for your users. See the *opsa-tenant-manager.sh* reference page (or the Linux manpage) for more information.

2. To list all of the published collectors and collections for a tenant, run the `$OPSA_HOME/bin/opsa-collection-config.sh -list -collectorhosts -username opsatenantadmin` command from the OpsA Server Appliance. Make a list of the published collectors and collections shown in the command output for your users. See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

3. Use the `$OPSA_HOME/bin/opsa-tag-manager.sh` script from the Operations Analytics Server Appliance to view and identify the tags in which your users are interested. Experiment with the options available with the `opsa-tag-manager.sh` script to identify the tags you must communicate to your users. . See the *opsa-tag-manager* reference page (or the Linux manpage) for more information. Make a list of these tags.

4. Combine the information from these steps and distribute this information to your OpsA users.

Chapter 6: Configuring Secure Access to Operations Analytics

There are several security methods you can configure for user access and authentication for Operations Analytics (OpsA).

Configuring SSL for the Operations Analytics Server Appliance

One-way SSL provides secure communication between the client and the Operations Analytics (OpsA) server. During an SSL session creation, the server sends a digital certificate (self-signed or CA signed) containing information about the server. This information, such as domain, organization, and location, helps the client verify the server's identity. SSL is disabled by default.

It is recommended that customers enable SSL communication for those environments where security is a concern. If customers are using a self-signed certificate, do one of the following:

- Export the public key of the Collector Appliance server certificate.
- Export the public root CA certificate if you are using CA signed server certificate.

Import the exported certificate into the trust store on the OpsA Server Appliance using the `$OPSA_HOME/bin/opsa-server-manager.sh` script. See "Configuring SSL for the Operations Analytics Collector Appliance" on page 127 for more information about configuring SSL for the OpsA Collector Appliance.

Use the information in this section to manage SSL on the OpsA Server Appliance.

Configuring SSL with a Certificate Authority (CA) Signed Certificate for the Operations Analytics Server Appliance

Complete the following steps to enable SSL communication to the Operations Analytics (OpsA) Server Appliance using a CA signed certificate:

1. Before enabling SSL to the OpsA Server Appliance, complete this step on the OpsA Server to create a user in JBoss **Management Realm**.

   Note: If you are configuring the OpsA Collector, do the following instead of completing this

step:

a. Run `/opt/HP/opsa/bin/opsa-collector-manager.sh`

b. Choose SSL (Option 1)

Do the following:

a. Run the `$OPSA_HOME/jboss/bin/add-user.sh` script.

b. For the first question, answer **"a" - Management User (mgmt-users.properties)**, then follow the instructions.

Note: You will need to provide the JBoss management realm credentials when enabling SSL later in these instructions.

2. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script on the OpsA server, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage), for more information.

Note: Configuration begins on the OpsA server side.

3. Select the **Configure SSL** option.

4. Select option 1 to generate a selfsign key pair.

5. Select option 2 to generate a certificate request for a signed CA certificate:

   a. Choose the `opsa_server` alias.

   b. Save the certificate to `/tmp/opsa_server_crf.src`.

6. After creating the request file, open this file in a text editor and do the following to copy the content into the CA form:

   a. Browse to **https://<your CA Request Server>/certsrv/**

   b. Select **Request a certificate** > **Advanced certificate request** > **Submit a certificate request by using a base-64-encoded CMC or PKCS #10 file, or submit a renewal request by using a base-64-encoded PKCS #7 file**

   c. Paste the content of the `opsa_server_crf.src` file into the CA form .

   d. Paste the resulting information into the **Saved Request** form.

e. Download the certificate chain on based 64 encoded format (`p7b` extension).

f. Copy this file to the OpsA server in the `/tmp` folder.

g. Go back to the first screen.

7. Select option 3 to import the CA signed certificate into the OpsA keystore.

> **Note:** The path to the signed opsa_server/collection certificate file is `/tmp/certnew.p7b`.

8. Complete the following steps to import the trusted certificate into the OpsA truststore:

a. Download the CA root certificate from **https://<your CA Request Server>/certsrv/.**

b. Select **Download a CA certificate, certificate chain, or CRL**

c. Select the **Base 64** radio button.

d. Select **Download CA certificate**

e. Copy this file to the OpsA server in the `/tmp` folder.

f. Go back to the first screen.

9. Select option 4 to import the trusted certificate to the OpsA truststore.

> **Note:** Enter the exact path to the downloaded CA certificate (.cer) file.

10. Select option 8 to enable SSL. You will need to enter the JBoss **Management Realm** user and password you created in the first step.The `opsa-server-manager.sh` script prompts you for the certificate alias to be used for SSL communication. Enter one of the aliases from the listed aliases.

11. Select the **Go back to main menu** option; then select **Option 6** to restart the Operations Analytics Server Appliance:

> **Note:** Your configuration changes will not occur unless the server is restarted.

> **Note:** Repeat all of the above steps for the OpsA Collector.

12. If the OpsA is already configured with collections (the collector is already registered), run the following command to unregister the collector from the server:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector IP address> -port 9443 -username <tenant admin username> -password
<password for tenant admin username>
```

Note: There is a need to unregister all of the collections. For example, to remove SiteScope Apache there is a need to run the following command: `opsa-collection-config.sh -unregister -collectorhost <IP address of Collector> -source SiteScope -domain Apache -group metrics -username opsatenantadmin`

Note: If the OpsA collector is not registered to the OpsA server, complete step 14 to do that registration.

13. If the OpsA is already configured with collections, run the following command to check that the collector was unregistered successfully:

    ```
    /opt/HP/opsa/bin/opsa-collection-config.sh -list -allversions -collectorhosts -
    port 9443 -username <tenant admin username> -password <password for tenant
    admin username>
    ```

    Note: Use a version of the following command to verify that the entire collection is unregistered.:
    `/opt/HP/opsa/bin/opsa-collection-config.sh -list -allversions -collectorhosts -port <port number> -username <user> -password <password>`

14. Run the following command to register the collector with the SSL command:

    ```
    /opt/HP/opsa/bin/opsa-collection-config.sh -register -collectorhost <collector
    IP address> -port 9443 -username <tenant admin username> -password <password
    for tenant admin username> -ssl
    ```

15. Operations Analytics users can access the Operations Analytics console using HTTP or HTTPS.

    Note: For example, use `https://<OpsA ServerIP Address>/opsa`

    Note: If a user attempts to use HTTP when HTTPS is configured, the user will automatically be redirected using HTTPS.

Configuring SSL with a Self-Signed Certificate for the Operations Analytics Server Appliance

Complete the following steps to enable SSL communication to the Operations Analytics (OpsA) Server Appliance using a self-signed certificate:

1. Before enabling SSL to the OpsA Server Appliance, complete this step to create a user in JBoss **Management Realm**. Do the following:

   a. Run the `$OPSA_HOME/jboss/bin/add-user.sh` script.

   b. For the first question, answer **"a" - Management User (mgmt-users.properties)**, then follow the instructions.

   > **Note:** You will need to provide the JBoss management realm credentials when enabling SSL later in these instructions.

2. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage), for more information.

3. Select the **Configure SSL** option.

4. Select the **Generate self-signed certificate for OPSA server** option to generate a self-signed certificate and add the certificate to the OpsA server keystore.

   > **Note:** The `opsa-server-manager.sh` script stores the self-signed certificate in the `keystore` file with the `opsa_server` alias name.

   > **Note:** Set the self-signed certificate attributes, such as `common name`, `country`, and `validity` by editing the `/opt/HP/opsa/conf/ssl/cert/opsa-self-signed-cert.template` file.

5. Optional: Select the **Import trusted certificate to OPSA server truststore** option to import trusted certificates (if any). For example, you can add HP ArcSight Logger's server certificate to the OPSA truststore file.

6. Select the **Enable/Disable SSL** option to enable SSL. You will need to enter the JBoss **Management Realm** user and password you created in the first step.The `opsa-server-manager.sh` script prompts you for the certificate alias to be used for SSL communication. Enter one of the aliases from the listed aliases.

7. Select the **Enable/Disable SSL** option to enable SSL. The `opsa-server-manager.sh` script

prompts you for the certificate alias to be used for SSL communication. Enter `opsa_server`.

Note: `opsa_server` is one of the aliases shown by the script.

8. Select the **Go back to main menu** option; then select the **Restart OPSA server** option to restart theOpsA Server Appliance.

Note: Your configuration changes will not occur unless the server is restarted.

9. OpsA users can access the Operations Analytics console using HTTP or HTTPS.

Note: If a user attempts to use HTTP when HTTPS is configured, the user will automatically be redirected using HTTPS.

Editing the SSL Configuration for the Operations Analytics Server Appliance

To change the certificate alias used for SSL communication, do the following:

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure SSL** option.
3. Select the **Change key alias to be used for SSL communication** option.
4. The `$OPSA_HOME/bin/opsa-server-manager.sh` script prompts you for the alias name, and lists the existing set of aliases from the OPSA keystore. Enter the desired alias name from the list.
5. Select the **Go back to main menu** option, then select the **Restart OPSA server** option to restart the Operations Analytics Server Appliance.

Note: Your configuration changes will not occur unless the server is restarted.

Disabling the SSL Configuration for the Operations Analytics Server Appliance

To disable the SSL communication, do the following:

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure SSL** option.
3. Select the **Enable/Disable SSL** option.
4. The `$OPSA_HOME/bin/opsa-server-manager.sh` script prompts you for a confirmation. Enter `yes` to disable the SSL communication.
5. Select the **Go back to main menu** option, then select the **Restart OPSA server** option to restart the Operations Analytics Server Appliance.

Managing the Operations Analytics Keystore and Truststore for the Operations Analytics Server Appliance

To modify the Operations Analytics (OpsA) keystore and truststore password, do the following:

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage) for more information.
2. Select the **Configure SSL** option.
3. Select the **Modify OPSA keystore/truststore password** option.
4. The `$OPSA_HOME/bin/opsa-server-manager.sh` script prompts you for the new password for the keystore and truststore. Enter the new passwords.
5. Select the **Go back to main menu** option, then select the **Restart OPSA server** option to restart the Operations Analytics Server Appliance.

To delete a certificate from the OpsA keystore and truststore, do the following:

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage) for more information.
2. Select the **Configure SSL** option.
3. Select the **Delete certificate from OPSA server keystore** or **Delete certificate from OPSA server truststore** option.
4. The `$OPSA_HOME/bin/opsa-server-manager.sh` script prompts you for the alias name, listing the existing set of aliases from the OPSA keystore/truststore. Enter the alias name to be deleted from the list.

5. Select the **Go back to main menu** option, then select the **Restart OPSA server** option to restart the OpsA Server Appliance.

Note: The certificate delete will fail if the certificate is in use.

To export a certificate from the OpsA keystore, do the following:

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage) for more information.
2. Select the **Configure SSL** option.
3. Select the **Export certificate from OPSA server keystore** option.
4. The `$OPSA_HOME/bin/opsa-server-manager.sh` script prompts you for the alias name, listing the existing set of aliases from the OpsA keystore. Enter the alias name to be deleted from the list.
5. The `$OPSA_HOME/bin/opsa-server-manager.sh` script prompts you for the file path to which the certificate should be exported. Enter the path to export the certificate.

To change an OpsA keystore file, do the following:

Note: The keystore password and the truststore password must be identical.

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage) for more information.
2. Select the **Configure SSL** option.
3. Select the **Change OPSA keystore file** option.
4. The `$OPSA_HOME/bin/opsa-server-manager.sh` script prompts you with a set of prerequisites actions to take before proceeding. Enter yes after you complete these prerequisites.
5. Enter the absolute path of the keystore file.

To change an OpsA truststore file, do the following:

Note: The keystore password and the truststore password must be identical.

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage) for more information.

2. Select the **Configure SSL** option.
3. Select the **Change OPSA truststore file** option.
4. The `$OPSA_HOME/bin/opsa-server-manager.sh` script prompts you with a set of prerequisites actions to take before proceeding. Enter yes after you complete these prerequisites.
5. Enter the absolute path of the truststore file.

Configuring SSL for the Operations Analytics Collector Appliance

One-way SSL provides secure communication between the client and the Operations Analytics (OpsA) server. During an SSL session creation, the server sends a digital certificate (self-signed or CA signed) containing information about the server. This information, such as domain, organization, and location, helps the client verify the server's identity. SSL is disabled by default.

It is recommended that customers enable SSL communication for those environments where security is a concern.

1. If customers are using a self-signed certificate, do one of the following:
   - Export the public key of the Collector Appliance server certificate.
   - Export the public root CA certificate if you are using CA signed server certificate.
2. Import the exported certificate into the trust store on the OpsA Server Appliance using the `$OPSA_HOME/bin/opsa-server-manager.sh` script. See "Configuring SSL for the Operations Analytics Collector Appliance" above for more information about configuring SSL for the OpsA Collector Appliance.

Use the information in this section to set up SSL and other communication changes on the OpsA Collector Appliance before registering the OpsA Collector Appliance with the OpsA server appliance. See "Registering Each Collector Appliance" on page 24 for more information.

Use the information in this section to manage SSL on the OpsA Collector Appliance.

Configuring SSL with a Certificate Authority (CA) Signed Certificate for the Operations Analytics Collector Appliance

SSL provides secure communication between the client and the Operations Analytics (OpsA) Collector Appliance. During an SSL session creation, the server sends a digital certificate (self-signed or CA signed) containing information about the server. This information, such as domain, organization, and location, helps the client verify the server's identity. SSL is disabled by default.

Complete the following steps to enable SSL communication to the OpsA Collector Appliance using a CA signed certificate:

1. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script on the OpsA collector. See the *opsa-collector-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure SSL** option.
3. Select option 1 to generate a selfsign key pair.
4. Select option 2 to generate a certificate request for a signed CA certificate:
   a. Choose the `opsa_server` alias.
   b. Save the certificate to `/tmp/opsa_server_crf.src`.
5. After creating the request file, open this file in a text editor and do the following to copy the content into the CA form:
   a. Browse to **https://<your CA Request Server>/certsrv/**
   b. Select **Request a certificate** > **Advanced certificate request** > **Submit a certificate request by using a base-64-encoded CMC or PKCS #10 file, or submit a renewal request by using a base-64-encoded PKCS #7 file**
   c. Paste the resulting information into the **Saved Request** form.
   d. Download the certificate chain on based 64 encoded format (`p7b` extension).
   e. Copy this file to the `opsa_server/collection` folder (What is the exact path to the folder?)
   f. Go back to the first screen.
6. Select option 3 to import the CA signed certificate into the OpsA keystore.

   Note: The path to the signed opsa_server/collection certificate file is `/tmp/certnew.p7b`.

7. Complete the following steps to import the trusted certificate into the OpsA truststore:
   a. Download the CA root certificate from **https://<your CA Request Server>/certsrv/.**
   b. Select **Download a CA certificate, certificate chain, or CRL**
   c. Select the **Base 64** radio button.
   d. Select **Download CA certificate**
   e. Copy the `<file>.cer` file to `opsa_server/collection`. (What is the exact path to the

folder?)

f. Go back to the first screen.

8. Select option 4 (why do I want to select option 4?)

Note: Enter the path to the downloaded CA certificate (.cer) file. (What is the exact path to this file?)

9. Select option 8 to enable SSL. You will need to enter the JBoss **Management Realm** user and password you created in the first step.The `opsa-server-manager.sh` script prompts you for the certificate alias to be used for SSL communication. Enter one of the aliases from the listed aliases.

10. Select the **Go back to main menu** option; then select option 6 to restart the Operations Analytics Server Appliance.

Note: Your configuration changes will not occur unless the server is restarted.

11. Run the following command to unregister the collector from the server:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector IP address> -port 9443 -username <tenant admin username> -password
<password for tenant admin username>
```

12. Run the following command to check that the collector was unregistered successfully:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -list -allversions -collectorhosts -
port 9443 -username <tenant admin username> -password <password for tenant
admin username>
```

13. Run the following command to register the collector with the SSL command:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -register -collectorhost <collector
IP address> -port 9443 -username <tenant admin username> -password <password
for tenant admin username> -ssl
```

14. **Optional Step**: Select the **Import trusted certificate to OPSA server truststore** option to import trusted certificates (if any). For example, you can add HP ArcSight Logger's server certificate to the OpsA truststore file.

Note: Although SSL does not need to be enabled for rawlog and structured log queries to work, this step is mandatory for rawlog and structured log queries to work properly. You must complete this certificate import on both the OpsA Server Appliance (for the rawlog query) and the OpsA Collector Appliance (for the structured log query). Follow these steps:

a. Log on to the Logger console, then click **System Admin**.

b. Select the **SSL Server Certificate** option under **security** on the left side of the screen.

c. Click the View Certificate button at the bottom of the screen.

d. After the dialog box opens, copy the certificate text and save it to a file on both the OpsA Collector Appliance and on the OpsA Server Appliance.

e. Complete this step on both the OpsA Collector Appliance and on the OpsA Server Appliance to import the certificate.

Configuring SSL with a Self-Signed Certificate for the Operations Analytics Collector Appliance

Complete the following steps to enable SSL communication to the Operations Analytics (OpsA) Collector Appliance using a self-signed certificate:

1. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script. See the *opsa-collector-manager.sh* reference page (or the Linux manpage), for more information.

2. Select the **Configure SSL** option.

3. Select the **Generate self-signed certificate for OPSA server** option to generate a self-signed certificate and add the certificate to the OpsA Collector Appliance keystore.

   Note: The `$OPSA_HOME/bin/opsa-collector-manager.sh` script stores the self-signed certificate in the `keystore` file with the `opsa_server` alias name.

   Note: Set the self-signed certificate attributes, such as `common name`, `country`, and `validity` by editing the `/opt/HP/opsa/conf/ssl/cert/opsa_self_signed_cert.template` file.

4. **Optional Step**: Select the **Import trusted certificate to OPSA truststore** option to import trusted certificates (if any). For example, you can add HP ArcSight Logger's server certificate to the OPSA truststore file.

   Note: Although SSL does not need to be enabled for rawlog and structured log queries to work, this step is mandatory for rawlog and structured log queries to work properly. You must complete this certificate import on both the OpsA Server Appliance (for the rawlog query) and

the OpsA Collector Appliance (for the structured log query). Follow these steps:

a. Log on to the Logger console, then click **System Admin**.

b. Select the **SSL Server Certificate** option under **security** on the left side of the screen.

c. Click the View Certificate button at the bottom of the screen.

d. After the dialog box opens, copy the certificate text and save it to a file on both the OpsA Collector Appliance and on the OpsA Server Appliance.

e. Complete this step on both the OpsA Collector Appliance and on the OpsA Server Appliance to import the certificate.

5. Select the **Enable/Disable SSL** option to enable SSL. The `opsa-collector-manager.sh` script prompts you for the certificate alias to be used for SSL communication. Enter `opsa_server`.

Note: `opsa_server` is one of the aliases shown by the script.

6. Select the **Go back to main menu** option; then select the **Restart OPSA Collector** option to restart the OpsA Collector Appliance.

Note: Your configuration changes will not occur unless the OpsA Collector Appliance is restarted.

7. If you have already registered the OpsA Collector Appliance with the OpsA Server Appliance, you will need to re-register this OpsA Collector Appliance for the new configuration changes to be used . Use the following command:
`$OPSA_HOME/bin/opsa-collection-config.sh –register -collectorhost <fully-qualified domain name of the collector host> -port <port> -username opsatenantadmin [-ssl] -coluser <collector_username> (the default collector username is opsa) -colpass <collector web service password> (the default password is opsa)`
See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

Editing the SSL Configuration for the Operations Analytics Collector Appliance

To change the server certificate used for SSL communication, do the following:

1. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script. See the *opsa-collector-manager.sh* reference page (or the Linux manpage), for more information.

2. Select the **Configure SSL** option.
3. Select the **Change key alias to be used for SSL communication** option.
4. The `$OPSA_HOME/bin/opsa-collector-manager.sh` script prompts you for the alias name, and lists the existing set of certificate aliases from the OPSA keystore. Enter the desired alias name from the list.
5. Select the **Go back to main menu** option, then select the **Restart OPSA Collector** option to restart the Operations Analytics Collector Appliance.

   Note: Your configuration changes will not occur unless the Operations Analytics (OpsA) Collector Appliance is restarted.

6. If you have already registered the OpsA Collector Appliance with the OpsA Server Appliance, you will need to re-register this OpsA Collector Appliance for the new configuration changes to be used . Use the following command:
   `$OPSA_HOME/bin/opsa-collection-config.sh –register -collectorhost <fully-qualified domain name of the collector host> -port <port> -username opsatenantadmin [-ssl] -coluser <collector_username> (the default collector username is opsa) -colpass <collector web service password> (the default password is opsa)`
   See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

Disabling the SSL Configuration for the Operations Analytics Collector Appliance

To disable the SSL communication, do the following:

1. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script. See the *opsa-collector-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure SSL** option.
3. Select the **Enable/Disable SSL** option.
4. The `$OPSA_HOME/bin/opsa-collector-manager.sh` script prompts you for a confirmation. Enter `yes` to disable the SSL communication.
5. Select the **Go back to main menu** option, then select the **Restart OPSA Collector** option to restart the Operations Analytics Collector Appliance.
6. If you have already registered the Operations Analytics (OpsA) Collector Appliance with the OpsA Server Appliance, you will need to re-register this OpsA Collector Appliance for the new configuration changes to be used . Use the following command:
   `$OPSA_HOME/bin/opsa-collection-config.sh –register -collectorhost`

`<fully-qualified domain name of the collector host> -port <port> -username opsatenantadmin [-ssl] -coluser <collector_username> (the default collector username is opsa) -colpass <collector web service password> (the default password is opsa)`
See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

Managing the OPSA Keystore and Truststore for the Operations Analytics Collector Appliance

To modify the Operations Analytics (OpsA) keystore and truststore password, do the following:

1. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script. See the *opsa-collector-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure SSL** option.
3. Select the **Modify OPSA keystore/truststore password** option.
4. The `$OPSA_HOME/bin/opsa-collector-manager.sh` script prompts you for the new password for the keystore and truststore. Enter the new passwords.
5. Select the **Go back to main menu** option, then select the **Restart OPSA Collector** option to restart the Operations Analytics Collector Appliance.
6. If you have already registered the OpsA Collector Appliance with the OpsA Server Appliance, you will need to re-register this OpsA Collector Appliance for the new configuration changes to be used. Use the following command:
`$OPSA_HOME/bin/opsa-collection-config.sh –register -collectorhost <fully-qualified domain name of the collector host> -port <port> -username opsatenantadmin [-ssl] -coluser <collector_username> (the default collector username is opsa) -colpass <collector web service password> (the default password is opsa)`
See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

To delete a certificate from the OPSA keystore and truststore, do the following:

1. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script. See the *opsa-collector-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure SSL** option.
3. Select the **Delete certificate from OPSA keystore** or **Delete certificate from OPSA truststore** option.
4. The `$OPSA_HOME/bin/opsa-collector-manager.sh` script prompts you for the alias name, listing the existing set of aliases from the OPSA keystore/truststore. Enter the alias name to be deleted from the list.

5. Select the **Go back to main menu** option, then select the **Restart OPSA Collector** option to restart the Operations Analytics Collector Appliance.

6. If you have already registered the OpsA Collector Appliance with the OpsA Server Appliance, you will need to re-register this OpsA Collector Appliance for the new configuration changes to be used . Use the following command:
   `$OPSA_HOME/bin/opsa-collection-config.sh –register -collectorhost <fully-qualified domain name of the collector host> -port <port> -username opsatenantadmin [-ssl] -coluser <collector_username> (the default collector username is opsa) -colpass <collector web service password> (the default password is opsa)`
   See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

To export a certificate from the OpsA keystore, do the following:

1. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script. See the *opsa-collector-manager.sh* reference page (or the Linux manpage), for more information.

2. Select the **Configure SSL** option.

3. Select the **Export certificate from OPSA server keystore** option.

4. The `$OPSA_HOME/bin/opsa-collector-manager.sh` script prompts you for the alias name, listing the existing set of aliases from the OPSA keystore. Enter the alias name to be deleted from the list.

5. The `$OPSA_HOME/bin/opsa-collector-manager.sh` script prompts you for the file path to which the certificate should be exported. Enter the path to export the certificate.

To change an OPSA keystore file, do the following:

> **Note:** The keystore password and the truststore password must be identical.

1. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-collector-manager.sh* reference page (or the Linux manpage), for more information.

2. Select the **Configure SSL** option.

3. Select the **Change OPSA keystore file** option.

4. The `$OPSA_HOME/bin/opsa-collector-manager.sh` script prompts you with a set of prerequisites actions to take before proceeding. Enter yes after you complete these prerequisites.

5. Enter the absolute path of the keystore file.

To change an OPSA truststore file, do the following:

Note: The keystore password and the truststore password must be identical.

1. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-collector-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure SSL** option.
3. Select the **Change OPSA truststore file** option.
4. The `$OPSA_HOME/bin/opsa-collector-manager.sh` script prompts you with a set of prerequisites actions to take before proceeding. Enter yes after you complete these prerequisites.
5. Enter the absolute path of the truststore file.

Configuring the HTTP and HTTPS Port for the Operations Analytics Collector Appliance

The Operations Analytics (OpsA) Collector Appliance comes with a pre-configured HTTP and HTTPS port of 9443. If you run into any port conflicts with this port, you might need to change it.

To change the HTTPS port to which the OpsA Collector Appliance listens, do the following:

1. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script. See the *opsa-collector-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure HTTP(S) port** option.
3. When prompted, change the port to a value greater than 1024.
4. Select the **Restart OPSA Collector** option.
5. After the HTTPS port is changed, you must register the OpsA Collector Appliance on the OpsA Server Appliance using the following command:
   `$OPSA_HOME/bin/opsa-collection-config.sh –register -collectorhost <collectorhost> -port <port> -username opsatenantadmin [-ssl] -coluser <collector_username> (the default collector username is opsa) -colpass <collector web service password> (the default password is opsa)`
   See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

   After you complete this step, future communication to this OpsA Collector Appliance uses the new HTTPS port.

Configuring the HTTP and HTTPS User Name and Password for the Operations Analytics Collector Appliance

The Operations Analytics (OpsA) Collector Appliance comes with a pre-configured HTTPS user name, **opsa**, having an identical password, **opsa**. It is recommended that customers change the user name and password for those environments where security is a concern.

1. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script. See the *opsa-collector-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure username/password** option.
3. When prompted, change the username and password values.

   Note: The `opsa-collector-manager.sh` script prompts you for the user name and password, then prompts you for the password again and validates that the passwords you entered are identical.
4. Select the **Restart OPSA Collector** option.
5. After the HTTP and HTTPS port is changed, you must register the OpsA Collector Appliance on the OpsA Server Appliance using the following command:
   `$OPSA_HOME/bin/opsa-collection-config.sh –register -collectorhost <collectorhost> -port <port> -username opsatenantadmin [-ssl] -coluser <collector_username> (the default collector username is opsa) -colpass <collector web service password> (the default password is opsa)`
   See the *opsa-collection-config.sh* reference page (or the Linux manpage) for more information.

   After you complete this step, future access to this OpsA Collector Appliance uses the new username and password values.

Configuring and Enabling Single Sign-on to Access Operations Analytics

These instructions show a practical example of configuring and enabling LWSSO between Operations Analytics (OpsA) and OMi. Use this practical example to help you configure LWSSO between OpsA and other applications you plan to use.

Enabling Single Sign-on (LWSSO) in OpsA permits users to launch the Operations Analytics console from an OMi event browser without needing to log on again. LWSSO is not enabled by default.

Note: For this example, the user accounts for the BSM server and the OpsA Server Appliance must match for these instructions to work correctly.

1. Before enabling LWSSO to the OpsA Server Appliance, complete this step to create a user in JBoss **Management Realm**. Do the following:
   a. Run the `$OPSA_HOME/jboss/bin/add-user.sh` script.

   b. For the first question, answer **"a" - Management User (mgmt-users.properties)**, then follow the instructions.

      Note: You will need to provide the JBoss management realm credentials when enabling LWSSO later in these instructions.

2. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage), for more information.

3. Select the **Configure LWSSO** option.

4. Select the **Configure LWSSOparameters** option.

5. When prompted with **Enter the Token Creation Key (initString) [xxxxxxx]**, enter the `initString` key. For example, if you are configuring LWSSO for OpsA and OMi, the value must match the `initString` configured in OMi.

   Note: To view the `initString` configured in OMi, log on to BSM and navigate to **BSM** > **Admin** > **Platform** > **Users and Permissions** > **Authentication Management**. It is important to use the exact `initString` configured in OMi for this example. It is also important to use the exact `initString` with other applications you plan to use with OpsA.

6. When prompted with **Enter the expiration period in minutes [60]**, enter the duration, in minutes, you want an LWSSO session to last before expiring.

7. When prompted with **Enter OPSA server domain**, enter the fully-qualified domain name of the Operations Analytics virtual appliance.

8. When prompted with **Enter trusted domains separated by comma**, the trusted domain names separated by a comma. Use the following form: `mytrusteddomain1.com, mytrusteddomain2.com`
   When finished, look for a **Configured LWSSO Successfully** message.

   Note: You must include the domain for the BSM server, considering the OMi example being show in these steps. This is even more important if the domain is not in the same domain in which the OpsA Server Appliance resides.

9. This step is important to complete if, considering the example being shown in these steps, the OMi domain is not in the same domain in which the OpsA Server Appliance resides.

   Note: If you already enabled LWSSO, and need to make LWSSO configuration changes, skip the instructions in this step.

   If this step is similar to the LWSSO configuration for your environment, complete the following:

   a. Select the **Configure LWSSO** option.

   b. From a browser, open the JMX console on the BSM server using the following syntax: http://<*FQDN of the BSM Server*> :8080/jmx-console/HtmlAdaptor?action=inspectMBean&name=Topaz%3Aservice%3DLW-SSO+Configuration

   c. Invoke the `addDNSDomainToTrustedHosts()` method and add the domain in which your Operations Analytics Server Appliance resides to the list.

10. After the `opsa-server-manager.sh` script finishes configuring LWSSO, it displays a **Configured LWSSO successfully** message, and gives you three options, one of which is to **Enable/Disable LWSSO**. Select the **Enable/Disable LWSSO** option to enable LWSSO. You will need to enter the JBoss **Management Realm** user and password you created in the first step.The `opsa-server-manager.sh` script prompts you for the certificate alias to be used for LWSSO communication. Enter one of the aliases from the listed aliases.

11. Select the **Go back to main menu** option; then select the **Restart OPSA server** option to restart the Operations Analytics Server Appliance.

    Note: Your configuration changes will not occur unless the server is restarted.

After completing the steps in this section, and configuring the correct URL on OMi, you can launch the Operations Analytics console from an OMi event browser without providing access credentials.

Note: If you already enabled LWSSO, and need to make LWSSO configuration changes, complete the above instructions, skipping step 8.

Disabling Single Sign-on to Access Operations Analytics

To disable LWSSO, do the following:

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure LWSSO** option.
3. Select the option **Enable/Disable LWSSO** to disable LWSSO.
4. Select the **Go back to main menu** option; then select the **Restart OPSA server** option to restart the Operations Analytics (OpsA) Server Appliance.

Note: Your configuration changes will not occur unless the server is restarted.

Configure Two-Way SSL Authentication for Accessing HP ArcSight Logger

Complete the following steps to configure two-way SSL authentication with ArcSight Logger:

1. Create an SSL truststore on Operations Analytics (OpsA) Server Appliance with HP ArcSight Logger's server certificate:
   a. Copy the self-signed or CA certificate from HP ArcSight Logger. You will find the self-signed certificate in the following location: `<Install_Dir>/current/local/apache/conf/ssl.crt/server.crt`
   b. Create a trust store on the OpsA Server Appliance with ArcSight Logger's self-signed certificate using the following command:
   ```
   keytool -import -alias logger -file <server_cert_path>/server.crt -storetype JKS -keystore /opt/HP/conf/opsa_truststore.jks
   ```
   Note: The keytool command prompts you for a password for the trust store. Provide a strong password and retain a copy of the password, as you will need it later. The keytool command also prompts you to trust the certificate. Type yes to trust the certificate
   .
2. Create a self-signed certificate and a keystore using OpenSSL for the OpsA Server Appliance:
   a. Create a private key using the following command:
   ```
   openssl genrsa -out /opt/HP/opsa/conf/opsa.key 1024
   ```
   b. Generate a certificate request using the following command:
   ```
   openssl req -new -key /opt/HP/opsa/conf/opsa.key -out /opt/HP/opsa/conf/opsa.csr
   ```

c. Create a self-signed certificate using the following command:
```
openssl x509 -req -days 365 -in /opt/HP/opsa/conf/opsa.csr -signkey
/opt/HP/opsa/conf/opsa.key -out /opt/HP/opsa/conf/opsa.crt
```

d. Export the self-signed certificate to PKCS#12 format using the following command:
```
openssl pkcs12 -export -out /opt/HP/opsa/conf/opsa.p12 -inkey
/opt/HP/opsa/conf/opsa.key -in /opt/HP/opsa/conf/opsa.crt
```

> **Note:** Retain a copy of the export password.

e. Use the following command to create a keystore and import the generated PKCS#12 format certificate:
```
keytool -importkeystore -srckeystore /opt/HP/opsa/conf/opsa.p12 -
destkeystore /opt/HP/opsa/conf/opsa_keystore.jks -srcstoretype
pkcs12 -deststoretype JKS -deststorepass <keystore_password> -
srcstorepass <export_password_entered_in_above_step>
```

3. Configure HP ArcSight Logger to enable client authentication:

a. Copy the OpsA Server Appliance's self-signed certificate from the following location:
`$OPSA_HOME/conf/opsa.crt`
to the following location on the HP ArcSight Logger server:
`<Install_Dir>/current/local/apache/conf/ssl.crt`

b. Edit HP ArcSight Logger's web server configuration file:
`<Install_Dir>/current/local/apache/conf/httpd.conf`

c. Modify the following lines and save your work:
```
SSLVerifyClient require SSLVerifyDepth 0 SSLCACertificateFile
<Install_Dir>/current/local/apache/conf/ssl.crt/opsa.crt
```

d. Run the following command to restart HP ArcSight Logger's web server: `<Install_Dir>/current/arcsight/service/apache restart`

4. Configure the OpsA Server Appliance's configuration file:

a. Edit the following file: `$OPSA_HOME/conf/opsa_config.prp`

b. Add the following line and save your changes. `logger.ssl.enabled=true`

5. Configure the JBoss Application server:

a. Edit the JBoss application server configuration file:
`$JBOSS_HOME/bin/standalone.conf`

b. Add the following lines and save your work:
```
JAVA_OPTS="$JAVA_OPTS -
Djavax.net.ssl.trustStore=/opt/HP/conf/opsa_truststore.jks" JAVA_
```

```
OPTS="$JAVA_OPTS -Djavax.net.ssl.trustStorePassword=<password of
trust store>" JAVA_OPTS="$JAVA_OPTS -
Djavax.net.ssl.keyStore==/opt/HP/conf/opsa_keystore.jks" JAVA_
OPTS="$JAVA_OPTS -Djavax.net.ssl.keyStorePassword=<password of key
store>"
```

6. Use the following commands to restart the JBoss server:

   a. Run the following command to stop JBoss:
   `$OPSA_HOME/jboss/bin/ jboss-cli.sh --connect controller=<ip_address>:19999 command=:shutdown`

   b. Run the following command to start JBoss:
   `$OPSA_HOME/jboss/bin/standalone.sh`

1. Create an SSL truststore on the OpsA Server Appliance with HP ArcSight Logger's server certificate:

   a. Copy the self-signed or CA certificate from HP ArcSight Logger. You will find the self-signed certificate in the following location: `<Install_Dir>/current/local/apache/conf/ssl.crt/server.crt`

   b. Create a trust store on the OpsA Server Appliance with HP ArcSight Logger's self-signed certificate using the following command:

```
keytool -import -alias logger -file <server_cert_path>/server.crt -
storetype JKS -keystore /opt/HP/conf/opsa_truststore.jks
```

> **Note**: The keytool command prompts you for a password for the trust store. Provide a strong password and retain a copy of the password, as you will need it later. The keytool command also prompts you to trust the certificate. Type yes to trust the certificate

2. Create a self-signed certificate and a keystore using OpenSSL for the OpsA Server Appliance:

   a. Create a private key using the following command:

```
openssl genrsa -out /opt/HP/opsa/conf/opsa.key 1024
```

   b. Generate a certificate request using the following command:

```
openssl req -new -key /opt/HP/opsa/conf/opsa.key -out
/opt/HP/opsa/conf/opsa.csr
```

   c. Create a self-signed certificate using the following command:

```
openssl x509 -req -days 365 -in /opt/HP/opsa/conf/opsa.csr -
signkey /opt/HP/opsa/conf/opsa.key -out /opt/HP/opsa/conf/opsa.crt
```

   d. Export the self-signed certificate to PKCS#12 format using the following command:

```
openssl pkcs12 -export -out /opt/HP/opsa/conf/opsa.p12 -inkey
/opt/HP/opsa/conf/opsa.key -in /opt/HP/opsa/conf/opsa.crt
```

Note: Retain a copy of the export password.

e. Use the following command to create a keystore and import the generated PKCS#12 format certificate:
```
keytool -importkeystore -srckeystore /opt/HP/opsa/conf/opsa.p12 -destkeystore /opt/HP/opsa/conf/opsa_keystore.jks -srcstoretype pkcs12 -deststoretype JKS -deststorepass <keystore_password> -srcstorepass <export_password_entered_in_above_step>
```

3. Configure HP ArcSight Logger to enable client authentication:

   a. Copy OpsA Server Appliance's self-signed certificate from the following location:
   `$OPSA_HOME/conf/opsa.crt`
   to the following location on the ArcSight Logger server:
   `<Install_Dir>/current/local/apache/conf/ssl.crt`

   b. Edit HP ArcSight Logger's web server configuration file:
   `<Install_Dir>/current/local/apache/conf/httpd.conf`

   c. Modify the following lines and save your work:
   `SSLVerifyClient require SSLVerifyDepth 0 SSLCACertificateFile <Install_Dir>/current/local/apache/conf/ssl.crt/opsa.crt`

   d. Run the following command to restart HP ArcSight Logger's web server: `<Install_Dir>/current/arcsight/service/apache restart`

4. Configure the OpsA Server Appliance's configuration file:

   a. Edit the following file: `$OPSA_HOME/conf/opsa_config.prp`

   b. Add the following line and save your changes. `logger.ssl.enabled=true`

5. Configure the JBoss Application server:

   a. Edit the JBoss application server configuration file:
   `$JBOSS_HOME/bin/standalone.conf`

   b. Add the following lines and save your work:
   ```
   JAVA_OPTS="$JAVA_OPTS -Djavax.net.ssl.trustStore=/opt/HP/conf/opsa_truststore.jks" JAVA_OPTS="$JAVA_OPTS -Djavax.net.ssl.trustStorePassword=<password of trust store>" JAVA_OPTS="$JAVA_OPTS -Djavax.net.ssl.keyStore==/opt/HP/conf/opsa_keystore.jks" JAVA_OPTS="$JAVA_OPTS -Djavax.net.ssl.keyStorePassword=<password of key store>"
   ```

6. Use the following commands to restart the JBoss server:

a. Run the following command to stop JBoss:
```
$OPSA_HOME/jboss/bin/ jboss-cli.sh --connect controller=<ip_
address>:19999 command=:shutdown
```

b. Run the following command to start JBoss:
```
$OPSA_HOME/jboss/bin/standalone.sh
```

1. Create an SSL truststore on OpsA Server Appliance with HP ArcSight Logger's server certificate:

   a. Copy the self-signed or CA certificate fromHP ArcSight Logger. You will find the self-signed certificate in the following location: `<Install_ Dir>/current/local/apache/conf/ssl.crt/server.crt`

   b. Create a trust store on the OpsA Server Appliance with HP ArcSight Logger's self-signed certificate using the following command:
   ```
   keytool -import -alias logger -file <server_cert_path>/server.crt
   -storetype JKS -keystore /opt/HP/conf/opsa_truststore.jks
   ```

   Note: The keytool command prompts you for a password for the trust store. Provide a strong password and retain a copy of the password, as you will need it later. The keytool command also prompts you to trust the certificate. Type yes to trust the certificate

   .

2. Create a self-signed certificate and a keystore using OpenSSL for the OpsA Server Appliance:

   a. Create a private key using the following command:
   ```
   openssl genrsa -out /opt/HP/opsa/conf/opsa.key 1024
   ```

   b. Generate a certificate request using the following command:
   ```
   openssl req -new -key /opt/HP/opsa/conf/opsa.key -out
   /opt/HP/opsa/conf/opsa.csr
   ```

   c. Create a self-signed certificate using the following command:
   ```
   openssl x509 -req -days 365 -in /opt/HP/opsa/conf/opsa.csr -
   signkey /opt/HP/opsa/conf/opsa.key -out /opt/HP/opsa/conf/opsa.crt
   ```

   d. Export the self-signed certificate to PKCS#12 format using the following command:
   ```
   openssl pkcs12 -export -out /opt/HP/opsa/conf/opsa.p12 -inkey
   /opt/HP/opsa/conf/opsa.key -in /opt/HP/opsa/conf/opsa.crt
   ```

   Note: Retain a copy of the export password.

   e. Use the following command to create a keystore and import the generated PKCS#12 format certificate:
   ```
   keytool -importkeystore -srckeystore /opt/HP/opsa/conf/opsa.p12 -
   destkeystore /opt/HP/opsa/conf/opsa_keystore.jks -srcstoretype
   ```

```
pkcs12 -deststoretype JKS -deststorepass <keystore_password> -
srcstorepass <export_password_entered_in_above_step>
```

3. Configure HP ArcSight Logger to enable client authentication:

   a. Copy OpsA Server Appliance's self-signed certificate from the following location:
   `$OPSA_HOME/conf/opsa.crt`
   to the following location on the ArcSight Logger server:
   `<Install_Dir>/current/local/apache/conf/ssl.crt`

   b. Edit HP ArcSight Logger's web server configuration file:
   `<Install_Dir>/current/local/apache/conf/httpd.conf`

   c. Modify the following lines and save your work:
   `SSLVerifyClient require SSLVerifyDepth 0 SSLCACertificateFile <Install_Dir>/current/local/apache/conf/ssl.crt/opsa.crt`

   d. Run the following command to restart ArcSight Logger's web server: `<Install_Dir>/current/arcsight/service/apache restart`

4. Configure the OpsA Server Appliance's configuration file:

   a. Edit the following file: `$OPSA_HOME/conf/opsa_config.prp`

   b. Add the following line and save your changes. `logger.ssl.enabled=true`

5. Configure the JBoss Application server:

   a. Edit the JBoss application server configuration file:
   `$JBOSS_HOME/bin/standalone.conf`

   b. Add the following lines and save your work:

```
JAVA_OPTS="$JAVA_OPTS -
Djavax.net.ssl.trustStore=/opt/HP/conf/opsa_truststore.jks" JAVA_
OPTS="$JAVA_OPTS -Djavax.net.ssl.trustStorePassword=<password of
trust store>" JAVA_OPTS="$JAVA_OPTS -
Djavax.net.ssl.keyStore==/opt/HP/conf/opsa_keystore.jks" JAVA_
OPTS="$JAVA_OPTS -Djavax.net.ssl.keyStorePassword=<password of key
store>"
```

6. Use the following commands to restart the JBoss server:

   a. Run the following command to stop JBoss:

```
$OPSA_HOME/jboss/bin/ jboss-cli.sh --connect controller=<ip_
address>:19999 command=:shutdown
```

   b. Run the following command to start JBoss:
   `$OPSA_HOME/jboss/bin/standalone.sh`

Configuring User Authentication using Public Key Infrastructure (PKI) to Access Operations Analytics

SSL Client Certificate authentication using PKI enables users to log on to the Operations Analytics console with a client-side X.509 certificate.

As part of user authentication, you can configure the Operations Analytics (OpsA) Server Appliance to check the certificate to make sure it has not been revoked. You can configure the revocation check to do one of the following:

- Validate the certificate using a Certificate Revocation List (CRL) .
- Validate the certificate using the Online Certificate Status Protocol (OCSP) to run a direct query to the PKI.

PKI authentication is disabled by default. To enable PKI authentication, do the following:

1. Before enabling SSL to the OpsA Server Appliance, complete this step to create a user in JBoss **Management Realm**. do the following:

   a. Run the `$OPSA_HOME/jboss/bin/add-user.sh` script.

   b. For the first question, answer **"a" - Management User (mgmt-users.properties)**, then follow the instructions.

   > **Note:** You will need to provide the JBoss management realm credentials when enabling SSL later in these instructions.

2. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage), for more information.

3. Select the **Configure PKI Authentication** option.

4. Use one of the following approaches:

   - **Self-signed Certificate**: Select the **Generate self-signed certificate for OPSA server** option to generate a self-signed certificate and add the certificate to the OpsA Server Appliance keystore.

   - **CA Signed Certificate**: Select the **Import CA certificate to OPSA server keystore** option to import a CA signed certificate to the OpsA Server Appliance keystore.

5. **Mandatory Step**: Select the **Import trusted certificate to OPSA server truststore** option to

import the trusted root CA certificate that will be used for PKI authentication.

Note: The certificate should be in base 64, otherwise the import will not work.

6. Select the **Enable/Disable PKI authentication** option to enable PKI. You will need to enter the JBoss **Management Realm** user and password you created in the first step.The `opsa-server-manager.sh` script prompts you for the certificate alias to be used for SSL communication, enter one of the aliases from the list. For example, you might enter `opsa-server`.

7. When prompted with **Allow smart card logon only [yes/no**], enter `yes` if only a smart log on is permitted. Enter `no` if a smart log on is not mandatory.

8. When prompted to select the field to use for a user name, enter the option you want OpsA to use.

9. When prompted for **Check for certificate revocation [yes/no**], enter `yes` for OpsA to check if the certificate provided by the client is revoked or not. Enter `no` to disable the revocation check. If you enter `yes`, the `opsa-server-manager.sh` script prompts you to select between the following revocation test methods:

   - **Option 1**: Validate the certificate using a Certificate Revocation List (CRL) .
   - **Option 2**: Validate the certificate using the Online Certificate Status Protocol (OCSP) to run a direct query to the PKI .

   Note: If you select option 2 the `opsa-server-manager.sh` script prompts you to configure the OCSP responder URL. You can accept the default behavior and have OpsA use the value of the `authorityInfoAccess` field of the client certificate to obtain the responder URL,or you can directly configure the OCSP responder URL.

10. When prompted with **Do you want to configure proxy host [yes/no]**, enter `yes` if you want to configure the proxy host to check for certificate revocation status. Enter `no` if you do not want to configure the proxy host to check for certificate revocation status (a local OCSP responder is available).

    If you enter `yes`, the `opsa-server-manager.sh` script prompts you for the following information:

    - proxy http proxy host
    - http port number
    - https proxy host
    - https port number

11. After successfully completing the registration, the `opsa-server-manager.sh` scrip shows an

`authentication enabled successfully` message.

12. Select the **Go back to main menu** option; then select the **Restart OPSA server** option to restart the OpsA Server Appliance.

> **Note:** Your configuration changes will not occur unless the server is restarted.

After completing the above steps, OpsA users can access the Operations Analytics console using HTTP or HTTPS as follows:

See the *opsa-server-manager.sh* reference page (or the Linux manpage), for more information.

1. If an OpsA user enters an HTTP URL, OpsA automatically redirects the URL to HTTPS, and shows a **Login with digital certificate** button.
2. After clicking the **Login with digital certificate** button, OpsA presents its digital certificate, and the browser verifies it against its truststore.
3. After verifying the OpsA certificate, OpsA prompts the user to select the client certificate. On selecting the client certificate, OpsA verifies the client certificate and performs authentication.

> **Note:** The client certificate must be installed and imported to the browser, otherwise the user is not prompted for the client certificate.

4. If the authentication is successful, the browser opens the OpsA home page.

Disabling User Authentication using Public (PKI) to Access Operations Analytics

To disable PKI authentication, do the following:

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure Client Authentication** option.
3. Select the **Enable/Disable client authentication** button.
4. The `$OPSA_HOME/bin/opsa-server-manager.sh` script prompts you for confirmation. Enter `yes` to disable PKI authentication.
5. The `$OPSA_HOME/bin/opsa-server-manager.sh` script disables PKI, then prompts, **Do you want to disable SSL as well [yes/no]**. Enter `yes` to disable SSL communication or `no` to

keep the existing SSL configuration.

6. Select the **Go back to main menu** option; then select the **Restart OPSA server** option to restart the Operations Analytics (OpsA) Server Appliance.

> **Note:** Your configuration changes will not occur unless the server is restarted.

After completing the above steps, OpsA presents its users with a user name and password page to access the Operations Analytics console.

Editing User Authentication using Public (PKI) to Access Operations Analytics

To modify PKI authentication, do the following:

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script, and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux manpage), for more information.
2. Select the **Configure Client Authentication** option.
3. Select the **Edit client authentication settings** button.
4. The `$OPSA_HOME/bin/opsa-server-manager.sh` script prompts you for PKI configuration information, similar to the prompts shown in "Configuring User Authentication using Public Key Infrastructure (PKI) to Access Operations Analytics" on page 145
5. Select the **Go back to main menu** option; then select the **Restart OPSA server** option to restart the Operations Analytics (OpsA) Server Appliance.

> **Note:** Your configuration changes will not occur unless the server is restarted.

Configuring SSL for Communication between Vertica and Operations Analytics

The information in this section explains how to manage SSL communications between OpsA and the Vertica (OpsA) database.

Enabling SSL Communications between the Operations Analytics Server Appliance and Vertica

Complete the following steps from the server that contains the Vertica database to enable SSL communications between the Operations Analytics Server Appliance and the Vertica (OpsA) database:

1. Complete only one of the following options:
   - **Option 1**: Self-signed certificate:
     i. Run the following command to create the CA private key:
     `openssl genrsa -des3 -out rootkey.pem`
     ii. Run the following command to create the CA public certificate. When prompted, fill in the correct information:
     `openssl req -new -x509 -key rootkey.pem -out root.crt`
     iii. Run the following command to create the server private key:
     `openssl genrsa -out server.key`
     iv. Run the following command to create the server certificate request. When prompted, fill in the correct information:
     `openssl req -new -out reqout.txt -key server.key`
   - **Option 2**: Certificate Authority (CA) Signed Certificate:
     i. Run the following command to create the server private key:
     `openssl genrsa -out server.key`
     ii. Run the following command to create the server certificate request. When prompted, fill in the correct information:
     `openssl req -new -out reqout.txt -key server.key`
     iii. Submit the server certificate request to a public Certificate Authority.
2. Run the following command to sign the certificate for the server that contains Vertica. This command uses the CA private key:
   `openssl x509 -req -in reqout.txt -days 3650 -sha1 -CAcreateserial -CA root.crt -CAkey rootkey.pem -out server.crt`

   Note: Following the completion of this step you have the server private key (the `server.key` file) and the signed server certificate (the `server.crt` file).
3. Run the following command to convert the signed certificate into a format understood by Java:
   `openssl x509 -in server.crt -out server.crt.der -outform der`
   Look for the `server.crt.der` file in the directory from which you ran the command shown in

this step.

4. Move the newly created `server.crt.der` file to a directory on the OpsA Server Appliance.

5. Although you run the other commands in this section from the server that contains the Vertica database, you must run the following command from the OpsA Server Appliance to import the signed certificate from Vertica (the file generated from the previous step) into the OpsA truststore:
`keytool -keystore $OPSA_HOME/conf/ssl/opsa-truststore.jks -alias verticasql -import –file server.crt.der`

   Note: If you have not updated the default password of the truststore, it is `keystore_neutron_analytics_bigdata_opsa_2013`. Check with the Operations Analytics administrator to obtain the correct password.

   Note: You can also complete this step can using the `opsa-server-manager.sh` script. See the *opsa-server-manager.sh* reference page (or the Linux manpage) for more information.

6. Assuming the username for the database user is `dbadmin`: As `dbadmin`, run the commands in the following steps to modify the Vertica configuration.

   Note: You can also complete these steps using the following Vertica tool:
   `/opt/vertica/bin/adminTools`

   a. `cp server.crt /home/dbadmin/opsadb/v_opsadb_node0001_catalog`

   b. `cp server.key /home/dbadmin/opsadb/v_opsadb_node0001_catalog`

   c. `chmod 700 /home/dbadmin/opsadb/v_opsadb_node0001_catalog/server.crt`

   d. `chmod 700 /home/dbadmin/opsadb/v_opsadb_node0001_catalog/server.key`

7. Assuming the username for the database user is `dbadmin`: As `dbadmin`, edit the following file:
`/home/dbadmin/opsadb/v_opsadb_node0001_catalog/vertica.conf`

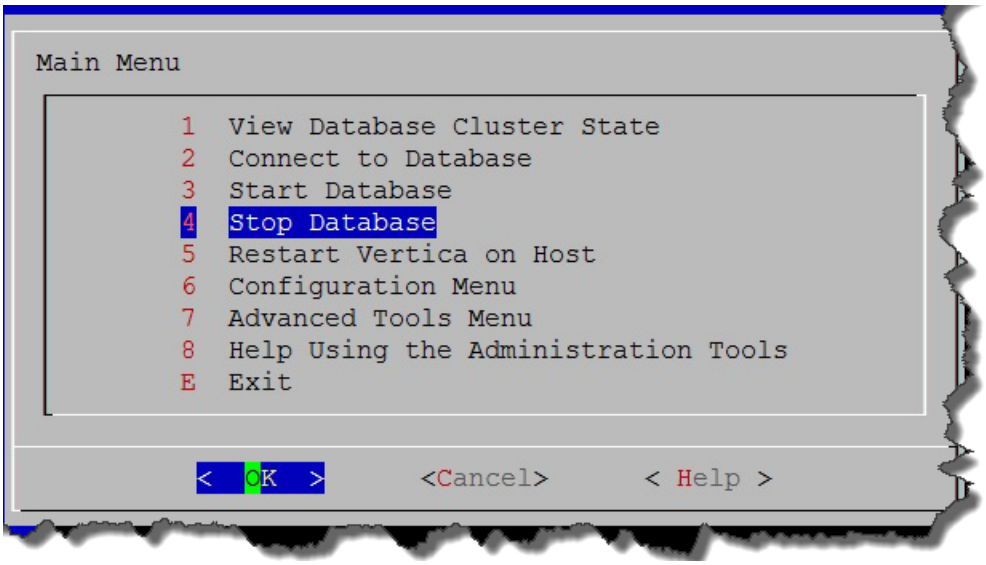
   Add the following lines; then save your work:

   ```
   EnableSSL=1
   ClientAuthentication = local all password
   ClientAuthentication = hostssl all 0.0.0.0/0 password
   ```

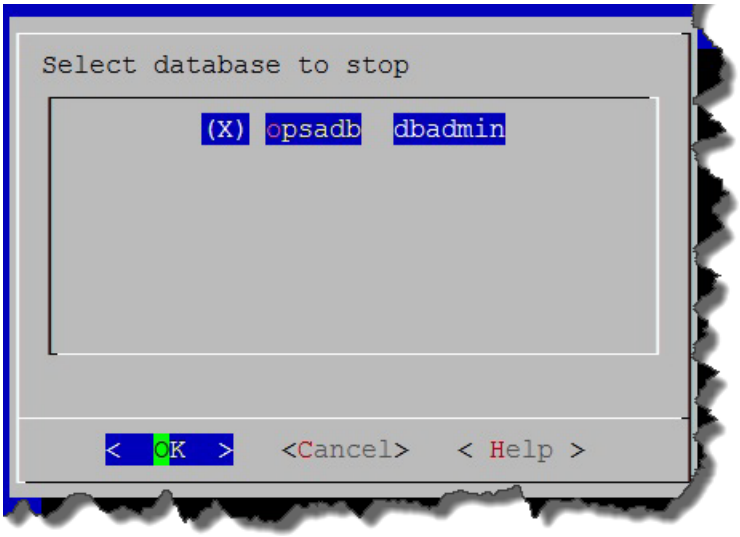
8. Complete the following steps to restart Vertica:

a. Assuming the username for the database user is `dbadmin`: As `dbadmin` run `/opt/vertica/bin/adminTools`.

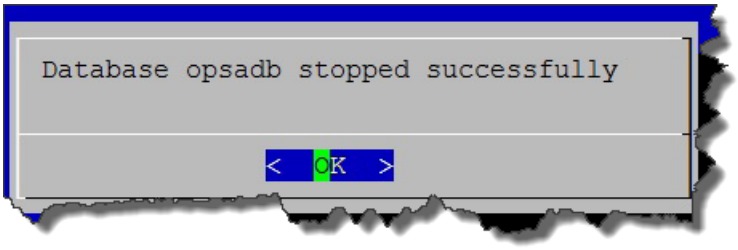
b. Select **Stop Database**; then click **OK**.



c. Select the database you want to stop (opsadb); then click **OK**.



d. Look for the following message to make sure the database stopped; then click **OK** to go back to the main menu.



Note: If users are still connected to the database, the **Stop Database** command might not work, as Vertica prevents it from shutting down. To stop the database anyway, do the following:

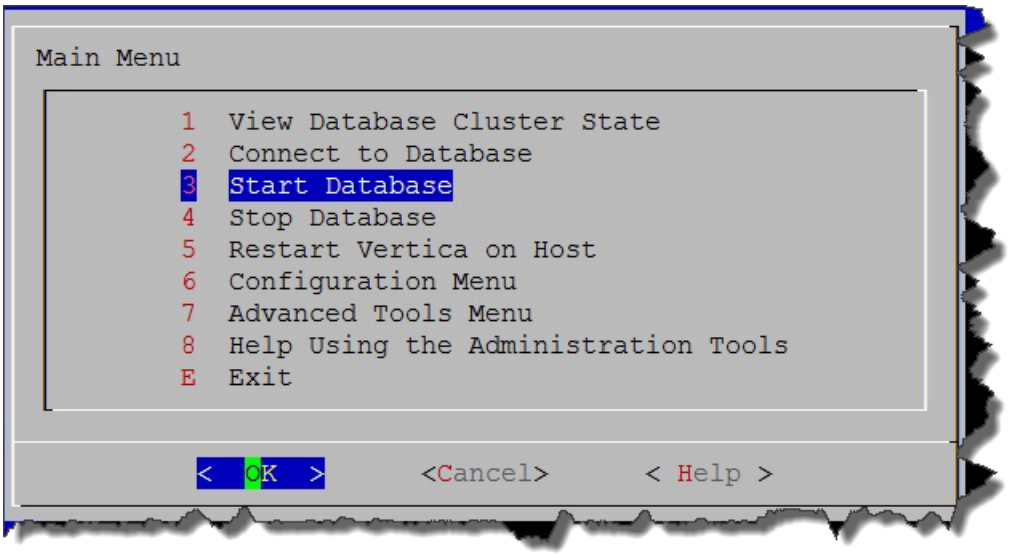
i. Select the **Advanced Tools Menu**

ii. Select **Stop Vertica on Host**

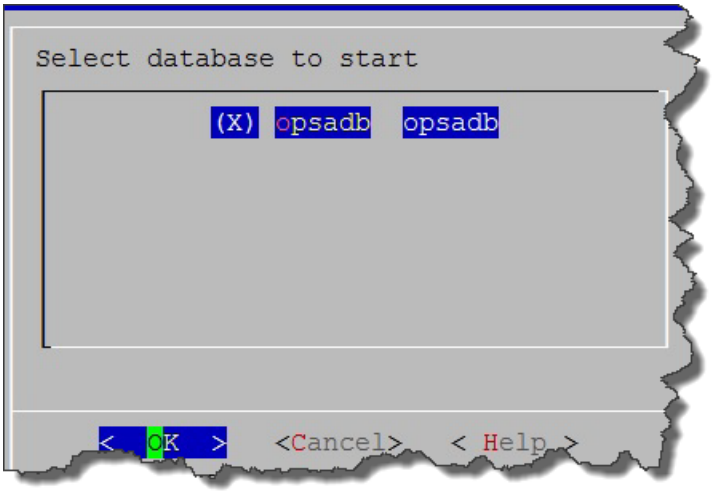
iii. Select the host.

Note: After completing these steps you can start the database again.

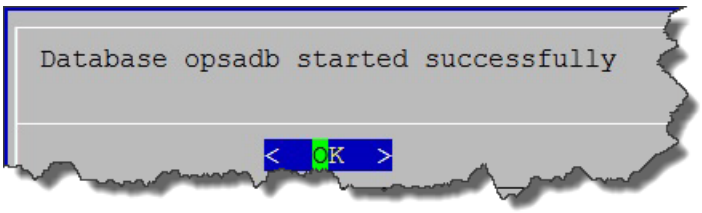
e. Select **Start Database**; then click **OK**.



f. Select the database you want to start (opsadb); then click **OK**.



g. Look for the following message to make sure the database started successfully; then click **OK** to go back to the main menu.



h. Exit the admin tool.

Note: If you set the file permissions incorrectly, it could result in the following error messages:

```
Unsafe permissions on private key file "/home/dbadmin/opsadb/v_
opsadb_node0001_catalog/server.key"

Could not load server certificate file "/home/dbadmin/opsadb/v_
opsadb_node0001_catalog/server.crt": error:0200100D:system
library:fopen:Permission denied
```

If you see error messages like this, see "Assuming the username for the database user is dbadmin: As dbadmin, run the commands in the following steps to modify the Vertica configuration. " on page 150 to correct any file permissions issues and continue.

9. Complete the following steps on the OpsA Server Appliance to enable SSL:

a. Edit the `$OPSA_HOME/conf/opsa-config.properties` file.

b. Search for the following string: `vertica.ssl.enabled=false`

> **Note:** If the string in this step does not exist, add the string shown in the next step to the bottom of the text.

c. Change the string as follows: `vertica.ssl.enabled=`**`true`**; then save your work.

d. Edit the `$OPSA_HOME/jboss/standalone/configuration/standalone.xml` file. You should see text that resembles the following:

```
datasource jndi-name="java:jboss/datasources/VerticaDS" pool-
name="VerticaDS" enabled="true" use-java-context="true">
<connection-url>jdbc:vertica://fully-qualified domain name of
Vertica Server:5433/opsadb</connection-url>
<driver>vertica</driver>
<pool>
<min-pool-size>20</min-pool-size>
<max-pool-size>100</max-pool-size>
</pool>
<security>>
<security-domain>opsa-ds</security-domain>
</security>
<validation>
<validate-on-match>false</validate-on-match>
<background-validation>false</background-validation>
</validation>
<statement>
<share-prepared-statements>false</share-prepared-statements>
</statement>
</datasource>
```

e. Add the line shown in bold font; then save your work.

```
datasource jndi-name="java:jboss/datasources/VerticaDS" pool-
name="VerticaDS" enabled="true" use-java-context="true">
<connection-url>jdbc:vertica://fully-qualified domain name of
Vertica Server:5433/opsadb</connection-url>
<connection-property name="ssl">true</connection-property>
<driver>vertica</driver>
<pool>
<min-pool-size>20</min-pool-size>
<max-pool-size>100</max-pool-size>
</pool>
<security>>
<security-domain>opsa-ds</security-domain>
```

```
</security>
<validation>
<validate-on-match>false</validate-on-match>
<background-validation>false</background-validation>
</validation>
<statement>
<share-prepared-statements>false</share-prepared-statements>
</statement>
</datasource>
```

10. Do the following to add the truststore location and password so that Jboss can find them and initialize the SSL handshake when communicating with Vertica.

    a. Edit the `$OPSA_HOME/jboss/standalone/configuration/standalone.xml` file.

    b. Locate the first bold phrase shown in following section in the `standalone.xml` file:

```
<system-properties>
<property name="org.apache.coyote.http11.Http11Protocol.MAX_HEADER_SIZE"
value="2097152"/>
<!--
To enable JDBC over SSL, uncomment this block.
<property name="javax.net.ssl.trustStorePassword" value="your_
truststore_password"/>
<property name="javax.net.ssl.trustStore"
value="/opt/HP/opsa/conf/ssl/opsa-truststore.jks"/>
-->
</system-properties>
```

    c. Remove the bold **<!--**, **To enable JDBC over SSL, uncomment this block.**, and **-->** strings shown in the previous step (doing so uncomments the lines), then add the correct truststore password to *your_truststore_password*. See the example shown below in bold font:

```
<system-properties>
<property name="org.apache.coyote.http11.Http11Protocol.MAX_
HEADER_SIZE" value="2097152"/>
<property name="javax.net.ssl.trustStorePassword" value="your_truststore_
password"/>
<property name="javax.net.ssl.trustStore" value="/opt/HP/opsa/conf/ssl/opsa-
truststore.jks"/>
</system-properties>
```

    > **Note:** If you have not updated the default password of the truststore, it is `keystore_neutron_analytics_bigdata_opsa_2013`. Check with the Operations Analytics administrator to obtain the correct password.

    d. Save your work.

11. You must restart Jboss any time you change the setting in the `opsa-config.properties` or `standalone.xml` files. Use the following command to restart the JBoss server: `$OPSA_`

```
HOME/bin/opsa-server restart
```

Disabling SSL Communications between the Operations Analytics Server Appliance and Vertica

Complete the following steps from the server that contains the Vertica database to disable SSL communications between the Operations Analytics Server Appliance and the Vertica (OpsA) database:

1. Edit the following file:
   `/opt/vertica/opsa_data/opsadb/v_opsadb_node0001_catalog/vertica.conf`

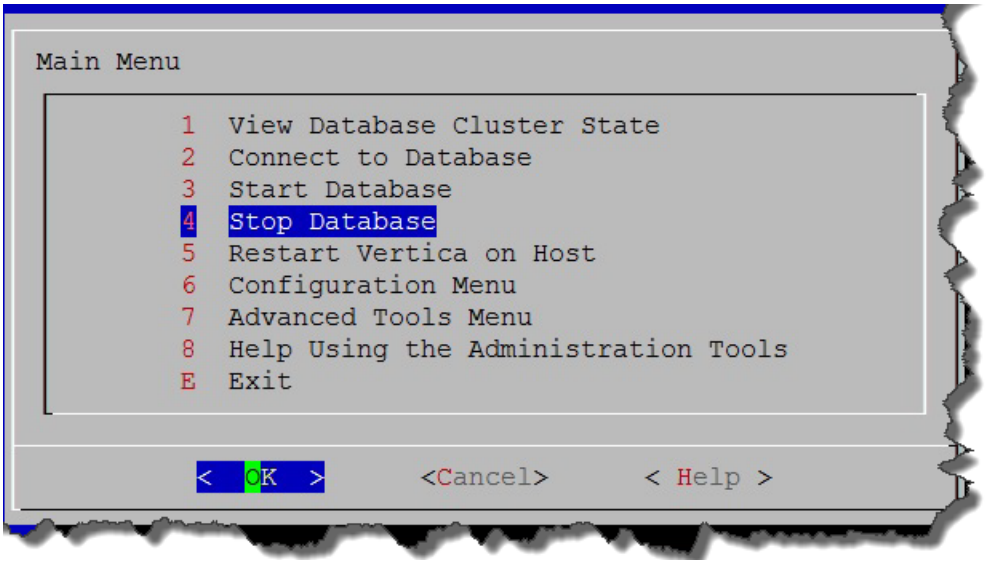
   Search for text that resembles the following lines.
   ```
   EnableSSL=1
   ClientAuthentication = local all password
   ClientAuthentication = hostssl all 0.0.0.0/0 password
   ```

   Comment the lines with the **#** character (shown in bold font below); then save your work:
   ```
   #EnableSSL=1
   #ClientAuthentication = local all password
   #ClientAuthentication = hostssl all 0.0.0.0/0 password
   ```

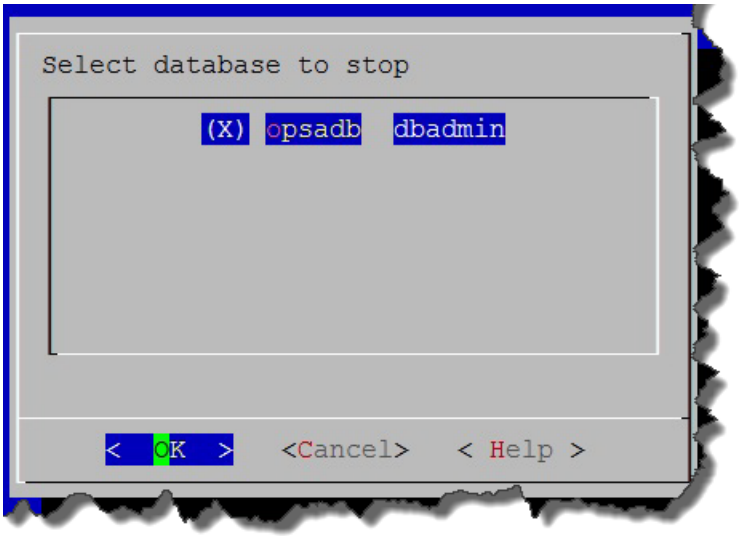
2. Complete the following steps to restart Vertica:

   a. Run `/opt/vertica/bin/adminTools`.

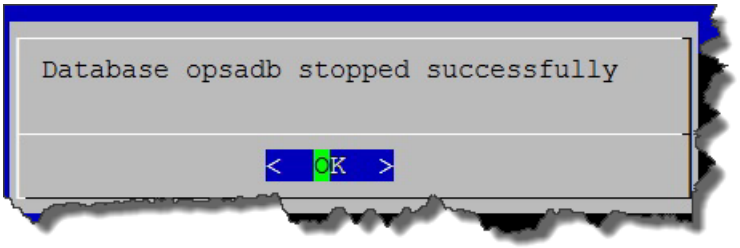
   b. Select **Stop Database**; then click **OK**.



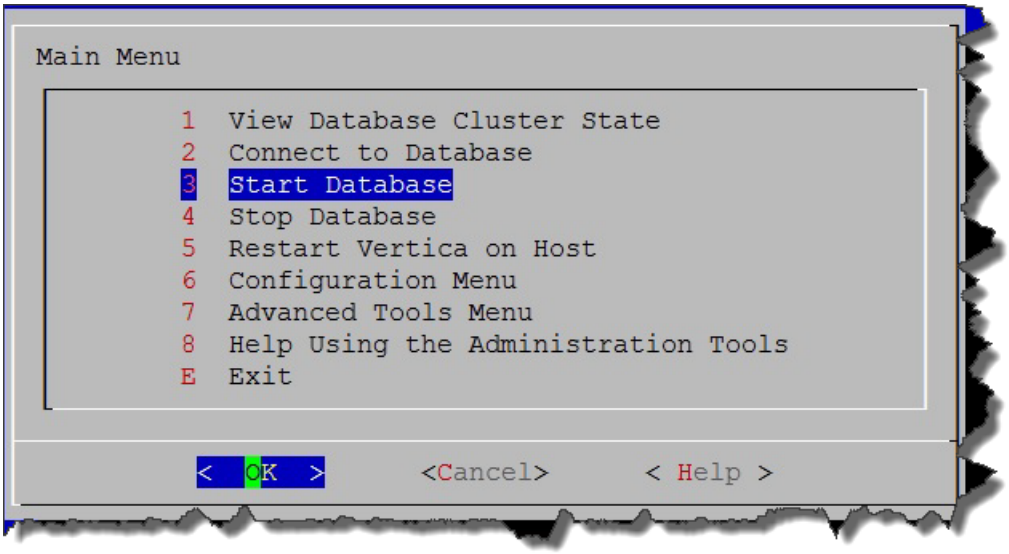
c. Select the database you want to stop (opsadb); then click **OK**.



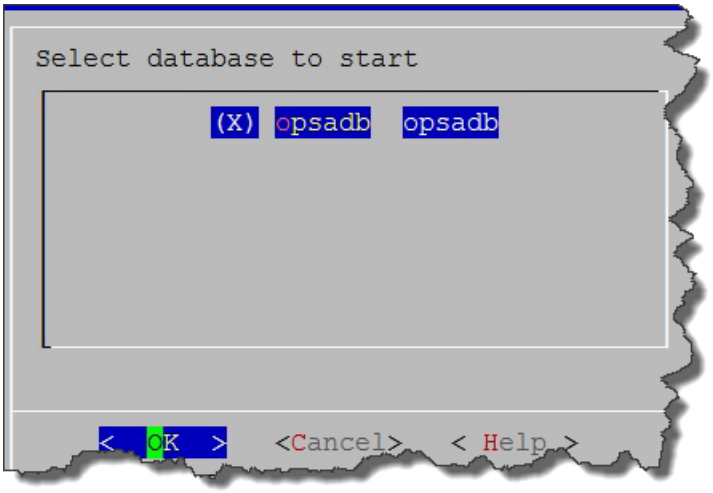
d. Look for the following message to make sure the database stopped; then click **OK** to go back to the main menu.



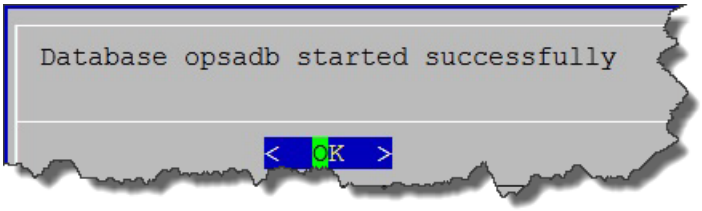
e. Select **Start Database**; then click **OK**.



f. Select the database you want to start (opsadb); then click **OK**.



g. Look for the following message to make sure the database started successfully; then click **OK** to go back to the main menu.



h. Exit the admin tool.

SSL communications between between the Operations Analytics Server Appliance and the Vertica (OpsA) database is now disabled.

Enabling SSL Communications between the Operations Analytics Collector Appliance and Vertica

Complete the following steps on the OpsA Collector Appliance to enable SSL between the OpsA Collector Appliance and the Vertica (OpsA) database:

1. Complete steps 1-8 in "Enabling SSL Communications between the Operations Analytics Server Appliance and Vertica" on page 149 to enable SSL on Vertica.
2. Edit the `$OPSA_HOME/conf/opsa-config.properties` file.
3. Search for the following string: `vertica.ssl.enabled=false`

   Note: If the string in this step does not exist, add the string shown in the next step to the bottom of the text.

4. Change the string as follows: `vertica.ssl.enabled=`**`true`**; then save your work.
5. Run the following command for the changes to take effect: `$OPSA_HOME/bin/opsa-collector restart`

SSL communications between the Operations Analytics Collector Appliance and the Vertica (OpsA) database is now enabled.

Disabling SSL Communications between the Operations Analytics Collector Appliance and Vertica

Complete the following steps on the OpsA Collector Appliance to disable SSL between the OpsA Collector Appliance and the Vertica (OpsA) database:

1. Complete the steps shown in "Disabling SSL Communications between the Operations Analytics Server Appliance and Vertica" on page 156 to disable SSL on Vertica.
2. Edit the `$OPSA_HOME/conf/opsa-config.properties` file.
3. Search for the following string: `vertica.ssl.enabled=true`

> **Note:** If the string in this step does not exist, add the string shown in the next step to the bottom of the text.

4. Change the string as follows: `vertica.ssl.enabled=`**`false`**; then save your work.
5. Run the following command for the changes to take effect: `$OPSA_HOME/bin/opsa-collector restart`

SSL communications between the Operations Analytics Collector Appliance and the Vertica (OpsA) database is now disabled.

Configuring SSL for the SMTP Server Used for OpsA Alerts

The information in this section explains how to manage SSL communications to your SNMP server.

1. Verify that OpsA servers are already communicating using SSL.
2. Copy the SMTP's root server certificate to the OpsA Servers and give the file full permissions.
3. Run the following command

   keytool -importcert -alias <opsa user name> -file <certificate file> -keystore /opt/HP/opsa/jdk/jre/lib/security/cacerts

Chapter 7: Maintaining Operations Analytics Collections

The information in this section helps you check the status of your collections and explains some common troubleshooting tools and techniques for collections.

Managing Collected Data File Usage with Existing Delete Policies

Operations Analytics Collector Appliances store collected data on their file systems. Each Operations Analytics Collector Appliance periodically runs a process controlled by **delete policies** to reduce the amount of stored data. You can adjust the parameters associated with these delete policies to better manage the data retained by each Operations Analytics Collector Appliance.

To configure the parameters associated with these delete policies, do the following from the Operations Analytics Collector Appliance or Appliances you want to control:

1. Edit the `/opt/HP/opsa/conf/opsa-collector.properties` file.
2. Using the helpful comments that reside in the `opsa-collector.properties` file, remove the `#` characters and set the desired parameters in the following lines:
   ```
   #com.hp.opsa.collector.file.garbage.schedule.interval_min = 30
   #com.hp.opsa.collector.file.garbage.diskfreepct.start = 10
   #com.hp.opsa.collector.file.garbage.diskfreepct.stop = 20
   #com.hp.opsa.collector.file.garbage.max.daysold = 30
   #com.hp.opsa.collector.file.garbage.enabled = true
   ```
3. Save your work.
4. Run the following command from the Operations Analytics Collector Appliance:

   ```
   $OPSA_HOME/bin/opsa-collector restart
   ```

Now the collected data on the Operations Analytics Collector Appliance or Appliances on which you made these changes are being managed by the newly adjusted parameters for these delete policies.

Although you can adjust parameters for the existing delete policies, you cannot add new delete policies or modify the functionality of the existing delete policies. The remainder of this section explains the static behavior of the existing delete policies.

Each Operations Analytics Collector Appliance contains the following delete policies.

- DELETE_ALWAYS : Delete the files if it exists.
- DELETE_LOW_FREE : Delete the files if the free disk space is low.
- DELETE_WHEN_OLD: Delete the file if it is old.

Each Operations Analytics Collector Appliance is configured as shown in "Delete Policies by Folder" below.

Delete Policies by Folder

Folder	Delete Policy
/opt/HP/opsa/data/archive	DELETE_WHEN_OLD & DELETE_LOW_FREE
/opt/HP/opsa/data/failed_to_load	DELETE_LOW_FREE
/opt/HP/opsa/data/load	DELETE_WHEN_OLD
/opt/HP/BSM/PMDB/extract	DELETE_ALWAYS

Troubleshooting Operations Analytics Collections

This section includes some troubleshooting tips and techniques for resolving Operations Analytics (OpsA) Collection issues.

Checking a Collector's Status

To check a collector's status, do the following:

- Run the following command from an Operations Analytics (OpsA) Server Appliance to list the collections deployed to that OpsA Collector Appliance:
  `$OPSA_HOME/bin/opsa-collection-config.sh -list -collectorhost <collector hostname> -username opsatenantadmin`
- Run the following commands from an OpsA Collector Appliance to check the status of collector sources and processes:
  - `$OPSA_HOME/bin/opsa-collector status`
  - `$OPSA_HOME/bin/opsa-loader status`
- Run the following command from an OpsA Server Appliance to check the status of the collector sources and processes configured on a OpsA Collector Appliance:
  `$OPSA_HOME/bin/opsa-collection-config.sh -status -collectorhost <collector hostname> -username opsatenantadmin`

Troubleshooting Configurations from the Operations Analytics Server Appliance

To troubleshoot collector and collection configuration, do the following from the Operations Analytics (OpsA) Server Appliance:

- If you completed the instructions to set up Operations Analytics System Health in "Checking Operations Analytics System Health" and installed and configured the Operations Analytics Log File Connector for HP ArcSight Logger on the OpsA Collector Appliances (to collect OpsA log files), you can check the **OpsA Health** dashboard and look for `ERROR` and `WARN` severity log messages.
- Look in the `/opt/HP/opsa/log/collection_config.log` file for any errors and warnings.
- If you want to adjust the logging level, edit the `/opt/HP/opsa/bin/log4j.properties` file and set the following properties. Then reconfigure the collection and view the results.
  - `log4j.logger.com.hp.opsa.collection.config=DEBUG,coll_cfg`
  - `log4j.logger.com.hp.opsa.collector=DEBUG,coll_cfg`

Troubleshooting the Absence of Collection Data

The information in this sections helps you troubleshoot collections that have been published to a Operations Analytics (OpsA) Collector Appliance, but are not collecting data. This troubleshooting takes place on the OpsA Collector Appliance.

After configuring a collection, open the **OpsA Health** Dashboard and check **MOVING_TOTAL (collector_rows)** to see if it contains a row for each newly configured collection. If any of your newly configured collections are not present, you might need to restart the collector using the following command: `$OPSA_HOME/bin/opsa-collector start`

- Always run the following commands to check that the collector and data loader are functioning correctly.
  - `$OPSA_HOME/bin/opsa-collector status`
  - `$OPSA_HOME/bin/opsa-loader status`
- Look for any error messages in the `/opt/HP/opsa/log/opsa-collector.log` and `/opt/HP/opsa/log/loader.log` files.
- If you want to adjust the logging levels, edit the `/opt/HP/opsa/conf/opsa-collector-log.properties` file and set the following properties:

- `log4j.logger.com.hp.opsa.collector = DEBUG`
- `log4j.logger.com.hp.opsa.collector.common = DEBUG`
- `log4j.logger.com.hp.opsa.collector.agent = DEBUG`
- `log4j.logger.com.hp.opsa.collector.server = DEBUG`

- If your collection problem is with the following collections, look in the associated log files shown for the collection:
  - HP Operations Agent or HP Operations Smart Plug-in for Oracle Collections
    `/opt/HP/BSM/PMDB/log/collections.log`
    `/opt/HP/BSM/PMDB/log/hpacollector.log`
  - HP Operations Manager (HPOM or OMi) Collections
    `/opt/HP/BSM/PMDB/log/collections.log`
    `/opt/HP/BSM/PMDB/log/dbcollector.log`
  - Any of associated HP BSM RTSM Collections
    `/opt/HP/BSM/PMDB/log/collections.log`
    `/opt/HP/BSM/PMDB/log/topologycollector.log`
- Custom SiteScope Collection: If your collection problem is with the Custom SiteScope Collection, do the following:
  - Check the `%SITESCOPE_HOME%/log/error.log` and `%SITESCOPE_HOME%/log/data_integration.log` files on the SiteScope server for any error messages about not being able to push SiteScope data to an OpsA Collector Appliance.
  - Check the OpsA Collector Appliance data integration to make it is configured correctly on the SiteScope server. See "Configuring a Custom SiteScope Collection (Detailed Method)" on page 281 for more information.
- Structured Log Collection: If your collection problem involves no structured log data being collected, check to make sure the configured query is correct. The easiest way to do this is to use the query in the ArcSight Logger console to see that the query works.
- For the following collections, look in the specific processed folder to check that the collector is collecting data for that collection:
  - HP Operations Agent Collection: `/opt/HP/opsa/data/pa_processed/<tenant>`
  - HP Operations Smart Plug-in for Oracle Collection: `/opt/HP/opsa/data/ora_pa_processed/<tenant>`
  - NNMi Custom Poller Collection: `/opt/HP/opsa/data/nnm_processed/<tenant>`

Note: The NNMi Custom Poller collector needs to have read/write access to the NNMi CSV files to move them to the processed directory. If the collector cannot move them, the NNMi Custom Poller CSV files will be reprocessed the next time the collector starts up. See "Configuring an NNMi Custom Poller Collection" on page 70 for more information.

- NNM ISPi Performance for Metrics Interface Health
  Collection: `/opt/HP/opsa/data/netinterface_processed/<tenant>`

Note: The NNM ISPi Performance for Metrics Interface Health collector needs to have read/write access to the NNM ISPi Performance for Metrics Interface Health CSV files to move them to the processed directory. If the collector cannot move them, the NNM ISPi Performance for Metrics Interface Health CSV files will be reprocessed the next time the collector starts up. See "Configuring an NNM ISPi Performance for Metrics Interface Health Collection (Detailed Method)" on page 230 for more information.

- NNM ISPi Performance for Metrics Component Health
  Collection: `/opt/HP/opsa/data/netcomponent_processed/<tenant>`

Note: The NNM ISPi Performance for Metrics Component Health collector needs to have read/write access to the NNM ISPi Performance for Metrics Component Health CSV files to move them to the processed directory. If the collector cannot move them, the NNM ISPi Performance for Metrics Component Health CSV files will be reprocessed the next time the collector starts up. See "Configuring an NNM ISPi Performance for Metrics Component Health Collection (Detailed Method)" on page 226 for more information.

- HP Operations Manager (HPOM) Collections: `/opt/HP/opsa/data/om_events_processed/<tenant>`
- HP Operations Manager (OMi) Collections: `/opt/HP/opsa/data/omi_events_processed/<tenant>`
- Custom SiteScope Collection:
  `/opt/HP/opsa/data/sis_processed/<tenant>`
  Look for collected Customer SiteScope Collection data in `/opt/HP/opsa/data/SIS_GDIAPI_DATA/<tenant>`.
- Custom CSV Collection: `<custom CSV source parent directory>_processed/<tenant>`

Note: The Custom CSV collector needs to have read/write access to the Custom CSV files to move them to the processed directory. If the collector cannot move them, the Custom CSV files will be reprocessed the next time the collector starts up. See "Configuring a Custom CSV Collection" on page 88 for more information.

- To check if data is being loaded into the OpsA database, a user can look at the files in the `/opt/HP/opsa/data/load/<tenant>` directory. Files with a `.working` extension are files to which the collector is actively writing. Working files should never be older than 10 minutes. So any files with a `.csv` extension are ready to be loaded into the OpsA database. If the system is functioning correctly, there should not be any `*.csv` files older than 10 minutes as well.

  - If you do not find any files with a `.csv` extension in the `/opt/HP/opsa/data/load/<tenant>` directory, do the following:

    i. Check for any CSV files that were successfully loaded into the OpsA database by looking in the `/opt/HP/opsa/data/archive/<tenant>` directory.

    ii. If you do not see files for the collection in question, check to see if the data loader has rejected the load files by looking in the `/opt/HP/opsa/data/failed_to_load` directory.

Chapter 8: Maintaining Operations Analytics

The information in this section explains how to complete maintenance tasks to protect your investment in Operations Analytics (OpsA).

Adding Operations Analytics Server Appliances

As your Operations Analytics (OpsA) environment expands, you might need to add more OpsA Server Appliances. To add another OpsA Server Appliance, do the following:

1. Install a new OpsA Server Appliance as shown in the *Operations Analytics Installation Guide*.
2. Run the `$OPSA_HOME/bin/opsa-server-postinstall.sh -scaleout` command to add the new OpsA Server Appliance. See the *opsa-server-postinstall.sh* reference page (or the Linux manpage) for more information.

   Note: After the `opsa-server-postinstall.sh` command completes, the passwords for the `opsa`, `opsatenantadmin`, and `opsaadmin` users (on the added servers) match the passwords you set when you installed the original OpsA Server Appliance.
3. Reboot all of the OpsA Server Appliances.
4. After all of the OpsA Server Appliances finish rebooting, you must reboot all of the OpsA Collector Appliances so they can identify the newly added OpsA Server Appliance.

After completing the above steps, the newly added OpsA Server Appliance should be ready to use.

Checking Operations Analytics System Health

You can configure Operations Analytics (OpsA) to display the metrics, topology, event, and log information available for the following OpsA servers and applications:

- OpsA Collector Appliances
- OpsA Server Appliances
- OpsA Database Servers
- HP ArcSight Logger

Configuring Operations Analytics Health

To configure OpsA to monitor its own active components, do the following:

1. Make sure the following software is installed and configured:

   a. Vertica: See *Task 2: Installing and Configuring the Vertica Software* in the *Operations Analytics Installation Guide*.

   b. HP ArcSight Logger: See *Task 3: Installing and Configuring HP ArcSight Logger* in the *Operations Analytics Installation Guide*.

   c. OpsA Server Appliance: See *Task 4: Installing and Licensing the Operations Analytics Server Appliance using the VMware vSphere Client* in the *Operations Analytics Installation Guide*.

   d. OpsA Collector Appliance: See *Task 5: Installing and Configuring the Operations Analytics Collector Appliance using the VMware vSphere Client* in the *Operations Analytics Installation Guide*.

2. Edit the `/etc/yum.conf` file and add the proxy information for your network.
   You entry should look similar to the following:

   ```
   # The proxy server - proxy server:port number
   proxy=http://mycache.mydomain.com:3128
   # The account details for yum connections
   proxy_username=yum-user
   proxy_password=qwerty
   ```

   Save your work.

3. Install the libstdc++ package on the Vertica database server, the OpsA Collector Appliance, and the OpsA Server Appliance.

4. To install the HP Operations Agent libraries, run the following command on the Vertica database server, the OpsA Collector Appliance, and the OpsA Server Appliance:
   `yum install compat-libstdc++-33-3.2.3-69.el6.i686`

5. Install the latest HP Operations Agent patches on the Vertica database server, the HP ArcSight Logger server, the OpsA Collector Appliance, and the OpsA Server Appliance.

6. Configure the syslogs from the Vertica database server, the OpsA Collector Appliance, and the OpsA Server Appliance to forward to the HP ArcSight Logger server by appending "`*.* @@<logger_hostname>:515`" to the `/etc/rsyslog.conf` file.)

7. Run the following command to restart the `rsyslog` service:
   `service rsyslog restart`

8. Configure the Operations Analytics Log File Connector for HP ArcSight Logger. See "Configuring the Operations Analytics Log File Connector for HP ArcSight Logger" on page 172 for more information.

Using Operations Analytics Health

After completing the steps in the previous section to configure OpsA to monitor its own active components, you can investigate the health of the OpsA servers using the **OpsA Health** dashboard. This dashboard shows you the following information:

- **Host System Metrics over Time**: Shows metrics such as CPU utilization and system up time that is being collected for each host.
- **Service Topology**: Shows all nodes that are configured as part of an OpsAservice and their health. This include OpsA servers, collector servers, Vertica database servers, and HP ArcSight Logger servers.
- **Collected Metric - Row Counts**: Shows the number of data rows being collected by each configured collection.
- **Configured Collections Dictionary**: Shows a table of information that includes collection property information for each collector host.
- **Log Messages(100+)**: Shows a list of the recent logs from OpsA service components.

See *Check the Health of Operations Analytics* in the *OpsA Help* for more information.

Deleting a Tenant

To delete a tenant from Operations Analytics (OpsA) , you must delete the tenant, then remove files from the OpsA Collector Appliance being used by the tenant you delete.

1. Remove all of the collection registrations for a tenant before deleting the tenant. See "Removing a Collection Registration for a Tenant" on the next page for more information.
2. There are two methods to use to delete a tenant from OpsA. To delete a tenant from OpsA, **use only one of the following methods**:

   Note: There are additional steps you must complete to remove files from your configured collectors after deleting a tenant.

   - **Method 1**: Run the `$OPSA_HOME/bin/opsa-tenant-manager.sh` command as a user assigned to the Super Admin User Group. opsaadmin is a Super Admin user created during

installation. You reset the password for this user during installation. Then follow the interactive commands to remove the tenant.

- **Method 2**: Run the `$OPSA_HOME/bin/opsa-tenant-manager.sh -delete -loginUser opsaadmin -loginPassword opsaadmin -tenant <tenant name>`

See the *opsa-tenant-manager.sh* reference page (or the Linux manpage) for information about creating and managing tenants.

3. To remove files from your configured collectors, do the following:

   a. From each Collector Appliance that contains collectors for the tenant being removed, run only one of the following commands to remove the tenant collection configuration:
      - If the OpsA Collector Appliance is only collecting data for the tenant being removed:
        `rm –rf /opt/HP/opsa/conf/collection/config.files/<collector host>`
      - If the OpsA Collector Appliance is collecting data for multiple tenants:
        `rm –rf /opt/HP/opsa/conf/collection/config.files/<collector host>/<tenant>`

   b. *Only complete this step if a collector appliance currently collects data for tenants other than the one being deleted*. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance. Use a Tenant Admin user for one of the other active tenants for which that this OpsA Collector Appliance is collecting.

      ```
      $OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost
      <fully-qualified domain name of the collector host> –username
      <tenant admin user>
      ```

      The `-publish` option uploads the collection configuration you created to the OpsA Collector Appliance. To verify that the collector configuration published successfully, look for a message stating that the publish was successful, that a table was successfully created, and that the collection was restarted.

   c. From the OpsA Collector Appliance, run the following commands to remove specific files from the OpsA Collector Appliance associated with the tenant being removed :

      - `rm -rf /opt/HP/opsa/data/load/<tenant name>`
      - `rm -rf /opt/HP/opsa/data/failed_to_load/<tenant name>`

Removing a Collection Registration for a Tenant

If you no longer want to analyze data for a collection, you must remove the collection registration and the stored data for that collection.

Note: Important: Just unregistering a collection does not drop the tables from the OpsA database.

If you do not complete all of the steps below, then try to register the collection again using the original name, OpsA will not create the database table. By completing all of the following steps, you will run the `opsa-collection-config.sh` script with the `-purgecollection` option. Doing so drops the database table and removes any references to the table.

To remove the collection registration and the stored data for that collection, do the following:

1. Run the following command to list all of the collectors for the tenant:
   `$OPSA_HOME/bin/opsa-collection-config.sh -list –collectorhosts -allversions -username opsatenantadmin`
   See the *opsa-collection-config.sh* reference page (or the Linux manpage), for more information.

2. Unregister the collections you no longer want to analyze for a tenant using the following command:

   `$OPSA_HOME/bin/opsa-collection-config.sh -unregister -source <collection source> -domain <collection domain> -group <collection group> -collectorhost <collector host> -username opsatenantadmin`

   Note: The `unregister` option is the opposite of the `create` option.The `unregister` option removes a collection from being collected on an Operations Analytics (OpsA) Collector Appliance where the `create` option was used to create that collection.

   Note: The command in this step also removes any Custom CSV Collection entries for the specified tenant.

3. Repeat the previous two steps until there are no collectors listed when running the command shown in step 1. If the command in step 1 lists no collectors, that means none of the original collectors for the tenant are collecting data for the collection you plan to remove.

4. After unregistering all of the collections you no longer want to analyze for a tenant (from all the tenant's collectors), remove the collection from the database using the following command:

   `$OPSA_HOME/bin/opsa-collection-config.sh -purgecollection -source <collection source> -domain <collection domain> -group <collection group> -collectorhost <collector host> -username <Tenant Admin User>`

   See the *opsa-collection-config.sh* reference page (or the Linux manpage), for more information.

   Note: The command in this step also removes all Custom CSV Collection data for the specified tenant from the Operations Analytics database.

   Note: After unregistering a Custom CSV Collection, the data remains intact. This means that

you can register a Custom CSV Collection that you removed and resume that Custom CSV Collection.

5. View the dashboards associated with the collections you just purged. The data from these purged collections should no longer be present in their associated dashboards.

Configuring the Operations Analytics Log File Connector for HP ArcSight Logger

The Operations Analytics Log File Connector for HP ArcSight Logger collects raw log files and tags them with some important information, such as hostname and process name. Operations Analytics (OpsA uses the Operations Analytics Log File Connector for HP ArcSight Logger to create a generic application log file connector that collects all of the log information needed by Operations Analytics.

Note: The Operations Analytics Log File Connector for HP ArcSight Logger is supported on the Windows and Linux platforms operating system.

Purpose : Use the installation instructions in this section if you must configure OpsA to collect raw log files and tag them with some important information. You also might need to install the Operations Analytics Log File Connector for HP ArcSight Logger to collect application logs.

Note: If there is an ArcSight connector available to collect the type of log file you must collect, use that connector. Only use the Operations Analytics Log File Connector for HP ArcSight Logger if an existing ArcSight connector does not meet your needs.

All other ArcSight connectors, such as the **SmartConnector for Apache HTTP Server Access File**, should be installed and configured using the ArcSight installation packages and documentation for any specific ArcSight connector.

Note: For the best results, use existing ArcSight connectors to collect log data, since they will do extensive parsing of the log message. The Operations Analytics Log File Connector for HP ArcSight Logger does not do any parsing of the log message.

If there is not a specific ArcSight connector available for the log collection you need, then use the Operations Analytics Log File Connector for HP ArcSight Logger. To install and configure the Operations Analytics Log File Connector for HP ArcSight Logger, follow the instructions in "Installing the Operations Analytics Log File Connector for HP ArcSight Logger" on the next page.

There are several out-of-the-box connectors (SmartConnectors) available which let you connect to standard Windows, Linux, or Apache logs. For details, see *Out of the Box Log Content* in the *Operations Analytics Installation Guide* and "Configuring ArcSight Logger Out of the Box Smart Connector Collections" on page 67.

For Operations Analytics to better utilize the raw log data from HP ArcSight Logger, use the fields shown in the following table.

Mandatory Fields to use for Arcsight Logger Collections

Field	Field Definition
Timestamp	The timestamp of the log message. For HP ArcSight Logger this is the receipt time that shows when the connector read the log message from the log file.
Hostname	The hostname of the system on which the log file resides. The Operations Analytics Log File Connector for HP ArcSight Logger sets the `Common Event Format (CEF)` field, `sourceHostName`, with the configured hostname for a log folder.

It is helpful to know the name of the process (for example: Apache Web Server) that created the log file along with the name of the application (such as Acme Order Application). The Operations Analytics Log File Connector for HP ArcSight Logger sets the `sourceProcessName` CEF field with the configured process name and the `sourceServiceName` CEF field with the configured application name for a log folder.

Installing the Operations Analytics Log File Connector for HP ArcSight Logger

During installation, the Operations Analytics Log File Connector for HP ArcSight Logger can be installed automatically (you are prompted about this automatic installation during installation). Use these instructions to manually install the Operations Analytics Log File Connector for HP ArcSight Logger on other non-OpsA servers that need to send log events to HP ArcSight Logger.

Look for the following installation package on any OpsA Collector Appliance: `$OPSA_HOME/logfile/opsa-arcsight-connector-dist-linux.zip`

Do not install the Operations Analytics Log File Connector for HP ArcSight Logger before deciding how you want to collect application logs in your environment. Select from the following deployment methods:

1. **Central Log File Management**: To use this method, install the Operations Analytics Log File Connector for HP ArcSight Logger on a central server. Using this approach, all of your application logs are stored using one of the following methods:

   - The application log files are copied to the central log server in their own directory.
   - The log file directories are shared, then mounted on the central server.

   This method enables you to use one central log server to administer the Operations Analytics Log File Connector for HP ArcSight Logger, but introduces extra work to get the application log files located on the central log server.

   When using this method, the Operations Analytics Log File Connector for HP ArcSight Logger is already installed in the `/opt/HP/opsa/arcsight` folder and configured for collecting log files

from the Collector Appliance. That means you can use the Operations Analytics Collector Appliance as a central log file server.

2. **Local Log File Management**: To use this method, install the Operations Analytics Log File Connector for HP ArcSight Logger on the same system that is running the application, and on which the log files are being created. This type of deployment eliminates the need to export or copy log files to a central server, but requires more effort to manage and maintain a larger quantity of Operations Analytics Log File Connector for HP ArcSight Loggers.

After selecting the deployment method you plan to use, complete the following steps:

1. Extract the install package in the desired installation directory (for example, you might run the following command: `unzip opsa-arcsight-connector-dist-linux.zip`).
2. Open the directory containing the extracted the zip files.
3. Run the following command to install the Operations Analytics Log File Connector for HP ArcSight Logger:
   - **Windows**: `opsa-logfile-flexconnector-config.bat`
   - **Linux**: `sh opsa-logfile-flexconnector-config.sh`
4. During installation, the script prompts you for the information shown in the following table:

Parameters for the opsa_logfile_flexconnector_config Script

Parameter	Parameter Description
ArcSight Logger Hostname	Hostname or IP address of the HP ArcSight Logger server to which you want this connector to send log messages.
ArcSight Logger Port (443)	The Smart Message Receiver port to which you want the Operations Analytics Log File Connector for HP ArcSight Logger to connect. By default HP ArcSight Logger uses port 443.
Name of Smart Message Receiver	The name of the Smart Message Receiver that receives messages from this Operations Analytics Log File Connector for HP ArcSight Logger. By default, HP ArcSight Logger defines a Smart Message Receiver called **SmartMessage Receiver**. You must enable this Smart Message Receiver name on the HP ArcSight Logger server.
Connector Name	The unique name for this installation of the Operations Analytics Log File Connector for HP ArcSight Logger.
Connector Location	This is an optional configuration for specifying the location of the Operations Analytics Log File Connector for HP ArcSight Logger.

5. After the installation completes, you will be prompted to configure the Operations Analytics Log File Connector for HP ArcSight Logger. To configure the Operations Analytics Log File Connector

for HP ArcSight Logger, complete the steps shown in "Configuring the Operations Analytics Log File Connector for HP ArcSight Logger" below.

Configuring the Operations Analytics Log File Connector for HP ArcSight Logger

During installation, the Operations Analytics Log File Connector for HP ArcSight Logger can be installed automatically. It can also be automatically configured for the Operations Analytics Collector and Server Appliances to collect Operations Analytics log events. If you chose to have this connector installed during the OpsA installation, use the following instructions to make additional configuration changes for the Operations Analytics Log File Connector for HP ArcSight Logger.

Unless specifically noted in these instructions, always use the $OPSA_HOME/bin/opsa-logfile-flexconnector-config.sh or $OPSA_HOME/bin/opsa-logfile-flexconnector-config.bat script to configure the Operations Analytics Log File Connector for HP ArcSight Logger.

Note: Configuring the Operations Analytics Log File Connector for HP ArcSight Logger using different methods than described in this documentation is not supported.

1. Stop the Operations Analytics Log File Connector for HP ArcSight Logger before configuring it. See "Manually Starting and Stopping the Operations Analytics Log File Connector for HP ArcSight Logger " on page 180 for more information.

   Note: You must restart the Operations Analytics Log File Connector for HP ArcSight Logger after configuring it.

2. Navigate to the root installation directory.

   Note: This is the folder in which you copied, then unzipped, the `opsa-arcsight-connector-dist-windows.zip` or `opsa-arcsight-connector-dist-linux.zip` file.

3. Run the following command to configure the Operations Analytics Log File Connector for HP ArcSight Logger:

   - **Windows**: `$OPSA_HOME/bin/opsa-logfile-flexconnector-config.bat`
   - **Linux**: `sh $OPSA_HOME/bin/opsa-logfile-flexconnector-config.sh`

4. The configuration menu appears, showing the following options:

   - "Option 1) Change Logger Server"
   - "Option 2) List Log Folders"

- "Option 3) Add Log Folder"
- "Option 4): Edit Log Folder"
- "Option 5): Delete Log Folder"
- "Option 6): Test Log Folders"
- "Option 7): Exit"

Select the option for the configuration task you want to complete.

Option 1) Change Logger Server

Use the **Change Logger Server** option to change the configuration associated with the connection between the Operations Analytics Log File Connector for HP ArcSight Logger and the HP ArcSight Logger server.

After selecting this option, the configuration script prompts you for the parameters shown in the next table.

HP ArcSight Logger Server Parameters

Parameter	Description
HP ArcSight Logger Hostname	The hostname or IP address of the HP ArcSight Logger server to which you want the Operations Analytics Log File Connector for HP ArcSight Logger to send log messages.
HP ArcSight Logger Port (443)	The Smart Message Receiver port to which you want HP ArcSight Logger to connect. HP ArcSight Logger uses port 443 by default.
Name of Smart Message Receiver	The name of the Smart Message Receiver that you want to receive the log messages from the Operations Analytics Log File Connector for HP ArcSight Logger. By default, HP ArcSight Logger defines a Smart Message Receiver called **SmartMessage Receiver**. The Smart Message Receiver name you provide must be enabled on the HP ArcSight Logger server.

Option 2) List Log Folders

Use the **List Log Folder** option to display the current list of configured log folders and associated configuration parameters within the Operations Analytics Log File Connector for HP ArcSight Logger.

Option 3) Add Log Folder

Use the **Add Log Folder** option to add a log folder to the Operations Analytics Log File Connector for HP ArcSight Logger.

After selecting this option, the configuration script prompts you for the following configuration parameters associated with adding a log folder.

Log Folder Parameters

Parameter	Description
Log Folder Path	The full directory path in which the log files reside.
Log File Name Wildcard	The wildcard filter used to select which files to process in the log folder path.
Process Name	The name of the process that creates the log files in the log folder path.
Application Name	The name of the application to which the log files are associated.
Hostname	The hostname of the server from which the log files originated. If the log files originate from the server on which the Operations Analytics Log File Connector for HP ArcSight Logger is installed, this would be the hostname of the local server. If the log files originate from a remote server, specify the hostname of the server from which the log files originated.
Multiline Regular Expression	If the log files being collected in a log folder span more than one line, you must provide a regular expression that is used to match the beginning of a line. A frequently used example of this would be to create a regular expression to match a time stamp that is located at the beginning of a line.

Option 4): Edit Log Folder

Use the **Edit log Folder** option to list the current configured log folders. To edit a specific log folder, enter the number assigned to that log folder. After selecting this option, the configuration script prompts you for the following configuration parameters associated with editing that log folder. The configuration script shows the current configured values, as shown in the following table (the values between the parentheses).

After selecting this option, the configuration script prompts you for the following configuration parameters associated with editing the specified log folder.

Configured Log Folder Parameters

Parameter	Description
Log Folder Path (<*current configure value*>)	The full directory path in which the log files reside.
Log File Name Wildcard (<*current configure value*>)	The wildcard filter used to select which files to process in the log folder path.
Process Name (<*current configure value*>)	The name of the process that creates the log files in the log folder path.
Application Name (<*current configure value*>)	The name of the application to which the log files are associated.
Hostname (<*current configure value*>):	The hostname of the server from which the log files originated. If the log files originate from the server on which the Operations Analytics Log File Connector for HP ArcSight Logger is installed, this would be the hostname of the local server. If the log files originate from a remote server, specify the hostname of the server from which the log files originated.
Multiline Regular Expression (<*current configure value*>)	If the log files being collected in a log folder span more than one line, you must provide a regular expression that is used to match the beginning of a line. A frequently used example of this would be to create a regular expression to match a time stamp that is located at the beginning of a line.

Option 5): Delete Log Folder

Use the **Delete Log Folder** option to list the configured log folders. To delete a specific log folder, enter the number assigned to that log folder.

Option 6): Test Log Folders

Use the **Test Log Folders** option to run a test against all of the configured log folders. This test checks to see that the file name pattern and multiline regular expression are working as configured. It is highly

recommended that you run this test to ensure that your configuration works before starting the Operations Analytics Log File Connector for HP ArcSight Logger.

The test does the following:

- For each configured log folder, the test script reads the first file that matches the file name pattern and parses that file using the multiline regular expression for that folder.
- The test script shows the first 3 messages of the file for each configured log folder.
- The test script only shows the first 40 characters of a line.

Option 7): Exit

Use the **Exit** option to exit the Operations Analytics Log File Connector for HP ArcSight Logger configuration script .

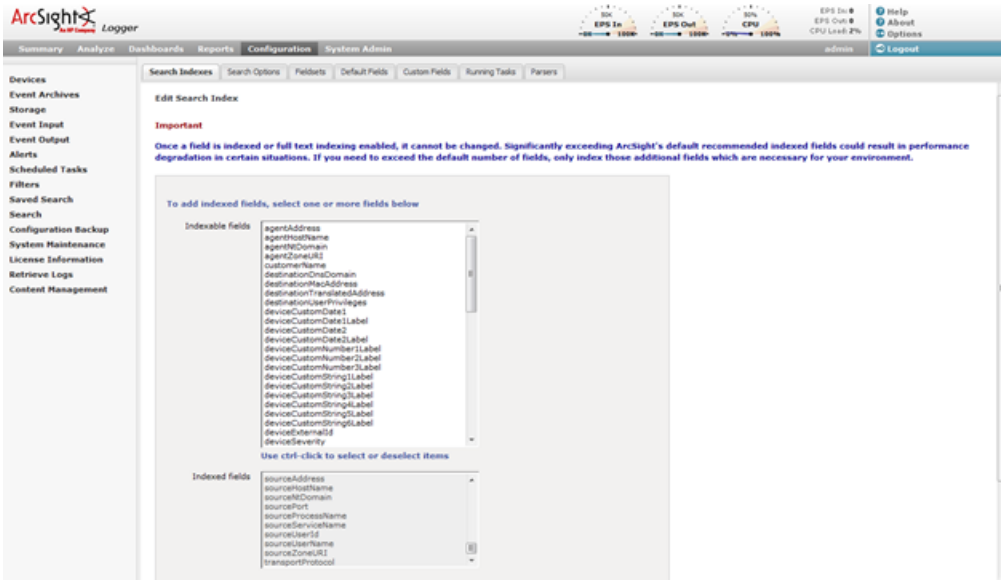
Filtering HP ArcSight Logger Queries

You can use the `sourceHostName`, `sourceProcessName`, and `sourceServiceName` CEF fields to filter log messages to just those for a particular application and process. For example, suppose you want to query only **Collector** related log files for Operations Analytics. To do this, you might run the following HP ArcSight Logger query: `sourceProcessName = "OPSA Collector" AND sourceServiceName = "OPSA"`

If any CEF fields you are trying to use as search filters are not working, do the following for each of those CEF fields to add them as search fields:

1. From the HP ArcSight Logger UI, Navigate to the **Configuration**->**Search**->**Search Indexes** TAB

2. Add the CEF fields to HP ArcSight Logger you want to use in your HP ArcSight Logger search queries.



Manually Starting and Stopping the Operations Analytics Log File Connector for HP ArcSight Logger

Starting the Operations Analytics Log File Connector for HP ArcSight Logger : Navigate to the root installation directory; then run the following command to start the Operations Analytics Log File Connector for HP ArcSight Logger:

- **Windows**: `".\bin\arcsight.bat agents"`
- **Linux**: `./bin/arcsight agents`

Stopping the Operations Analytics Log File Connector for HP ArcSight Logger: Type `control-C` to stop the Operations Analytics Log File Connector for HP ArcSight Logger.

To make it easier to start and stop the Operations Analytics Log File Connector for HP ArcSight Logger, follow the instructions shown in "Configuring the Operations Analytics Log File Connector for HP ArcSight Logger to Run as a Service" below.

Configuring the Operations Analytics Log File Connector for HP ArcSight Logger to Run as a Service

It is recommended that you configure the Operations Analytics Log File Connector for HP ArcSight Logger to run as a service so that it automatically starts when rebooting the server on which the

connector is running. Use one of the following methods to run the Operations Analytics Log File Connector for HP ArcSight Logger as a service.

Method 1 (Command Line):

Note: You must run the `arcsight.bat` (Windows) and `arcsight` (Linux) scripts shown in this section as a user that has permission to add a service to the server on which you run the command. It is recommended that the user the service is run as is the same user that installed the Operations Analytics Log File Connector for HP ArcSight Logger.

1. From the server on which you installed the Operations Analytics Log File Connector for HP ArcSight Logger, navigate to the root installation directory.

   Note: This is the folder in which you copied, then unzipped, the `opsa-arcsight-connector-dist-windows.zip` or `opsa-arcsight-connector-dist-linux.zip` file.

2. Run the following command to configure the Operations Analytics Log File Connector for HP ArcSight Logger to run as a service.

   - **Windows**: `current\bin\arcsight.bat agentsvc -i -u <user that installed the Operations Analytics Log File Connector for HP ArcSight Logger>`
   - **Linux**: `current/bin/arcsight agentsvc -i -u <user that installed the Operations Analytics Log File Connector for HP ArcSight Logger>`

3. Complete the following steps to adjust the amount of memory used by the Operations Analytics Log File Connector for HP ArcSight Logger service:

   a. Edit the `<InstallDir>/current/user/agent/agent.wrapper.conf` file.

   b. Set the `wrapper.java.initmemory` property value to a larger value. For example, if the value is set to 256 (MB), you might double the value to 512 (MB).

   c. Set the `wrapper.java.maxmemory` property value to a larger value. For example, if the value is set to 512 (MB), you might double the value to 1024 (MB).

      Note: The `wrapper.java.maxmemory` property value must be equal to or greater than the `wrapper.java.initmemory` property value.

   d. Save your work.

Method 2: User Interface

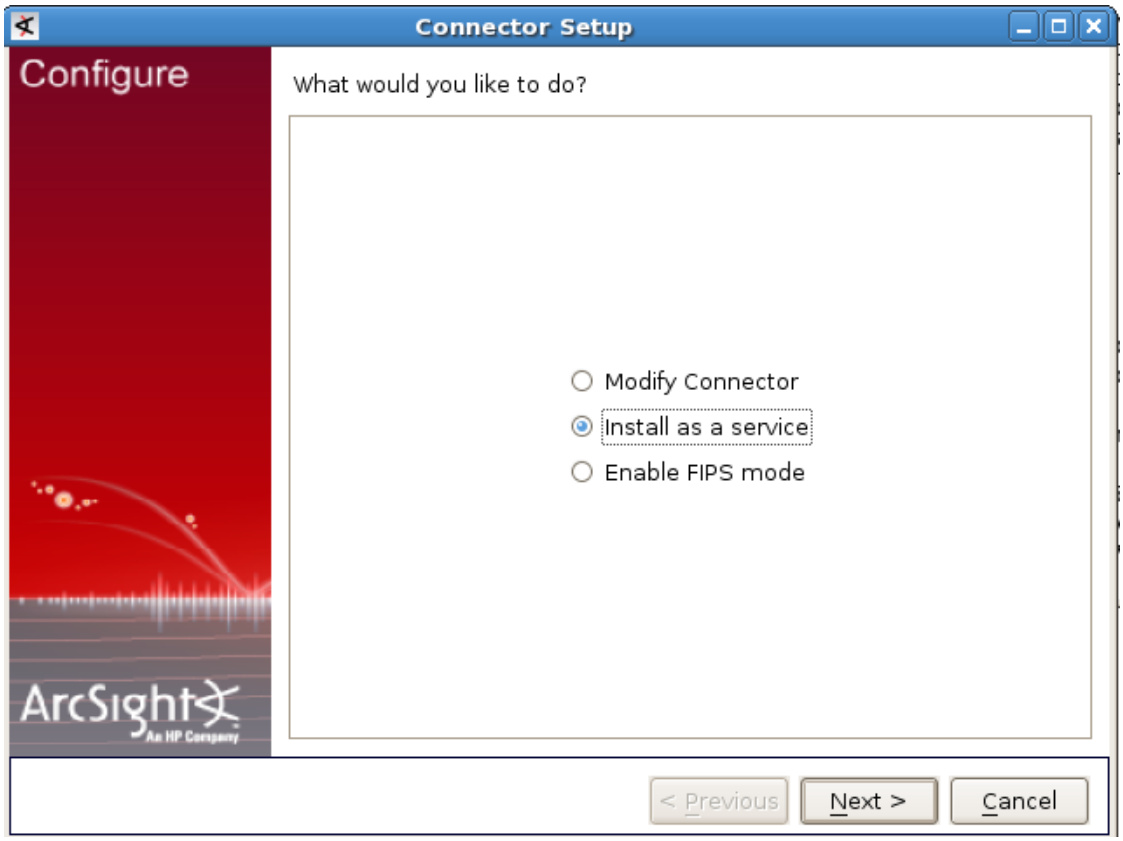
1. Navigate to the root installation directory.

   Note: This is the folder in which you copied, then unzipped, the `opsa-arcsight-connector-dist-windows.zip` or `opsa-arcsight-connector-dist-linux.zip` file.

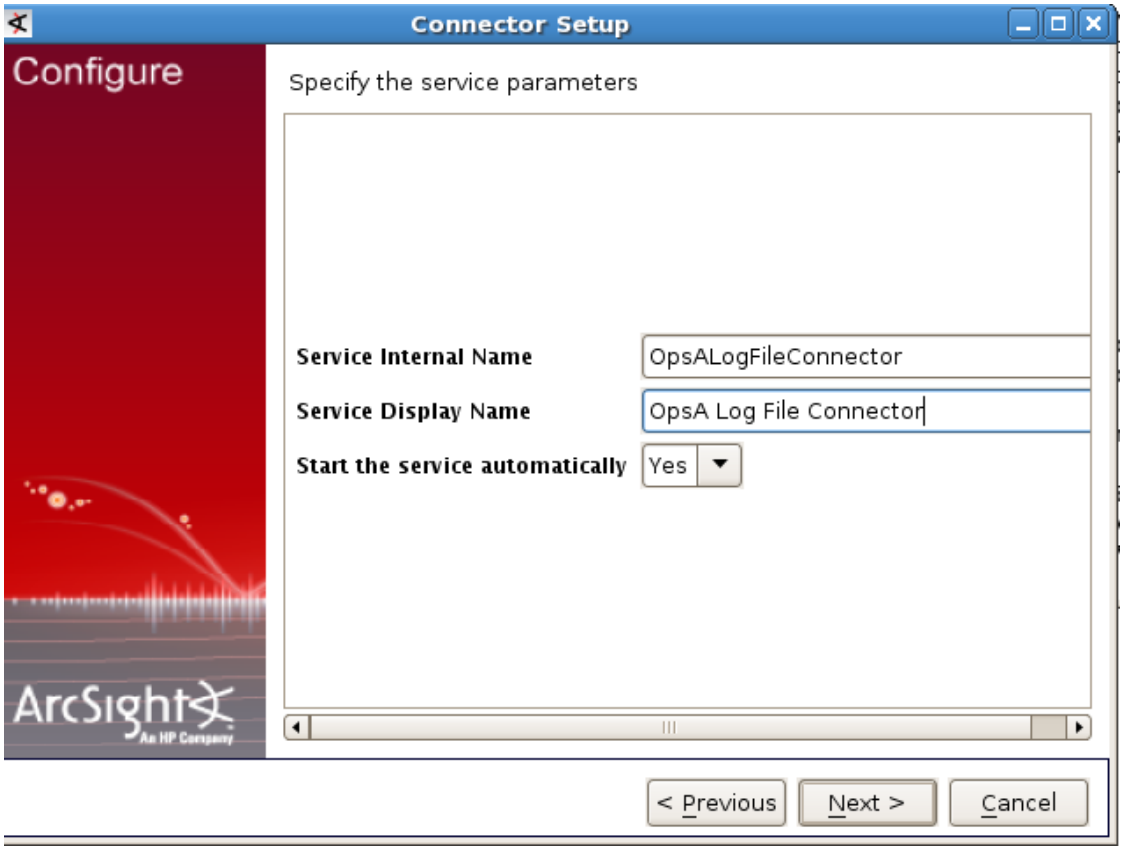
2. Run the following command to configure the Operations Analytics Log File Connector for HP ArcSight Logger to run as a service.

   - **Windows**: `".\bin\arcsight agentsetup.bat -c"`
   - **Linux**: `./bin/arcsight agentsetup -c`

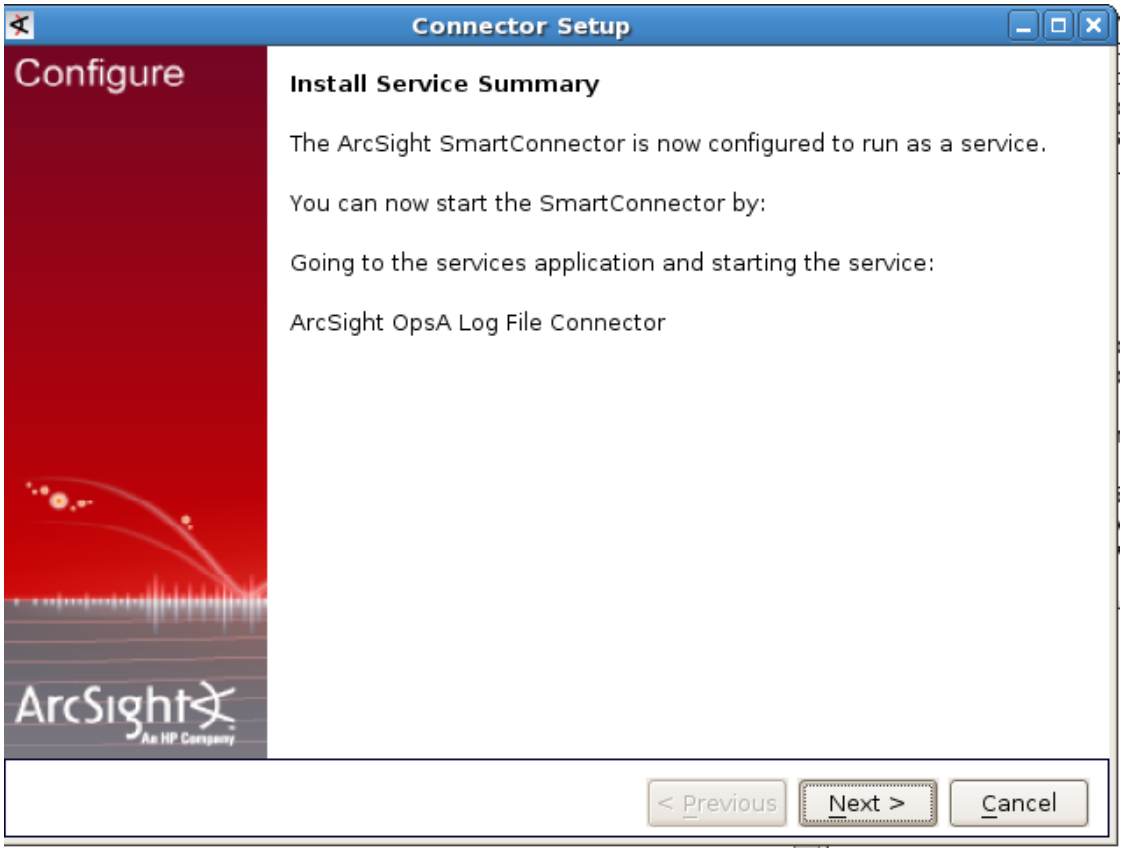
3. Select **Install as a service**; then click **Next**.



4. Enter the service name details. You must set **Start the service automatically** to **Yes**. Click **Next** after you finish.

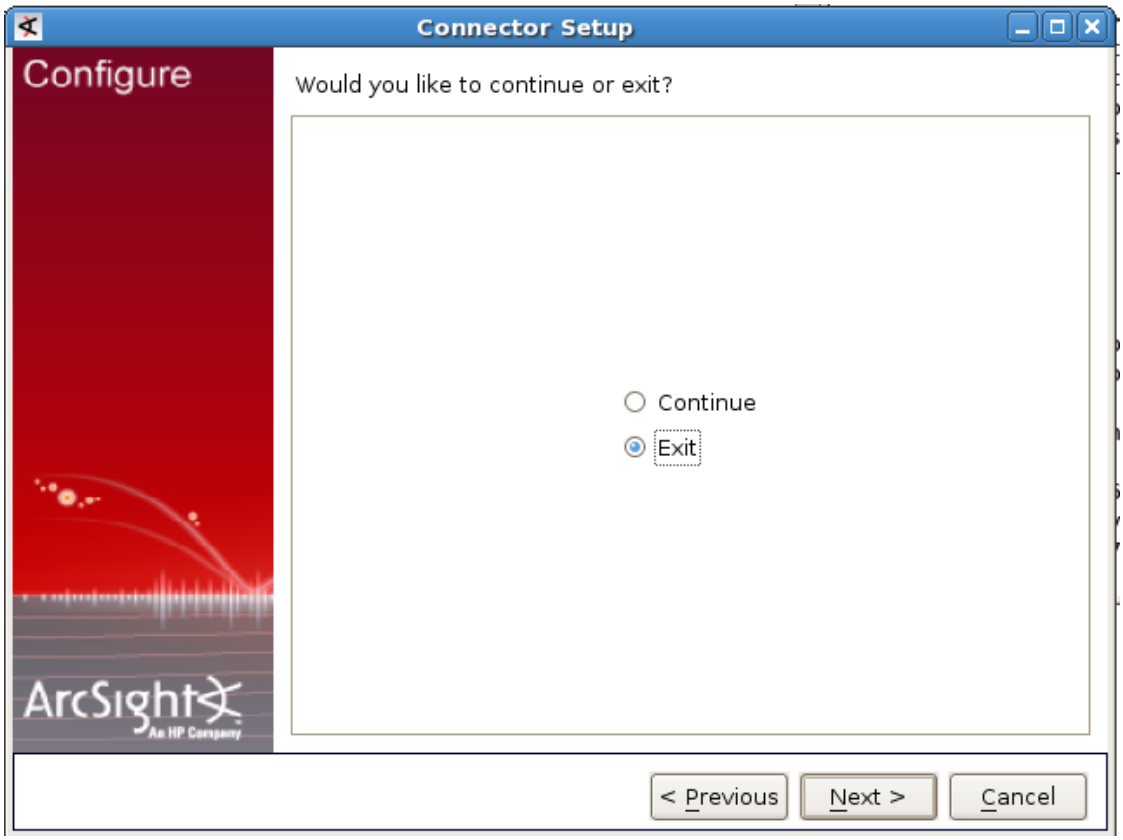


5. After the installation completes successfully, you should see the following message:



Click **Next** to continue.

6. Select **Exit** ; then click **Next** to complete the installation.



7. Complete the following steps to adjust the amount of memory used by the Operations Analytics Log File Connector for HP ArcSight Logger service:

   a. Edit the `<InstallDir>/current/user/agent/agent.wrapper.conf` file.

   b. Set the `wrapper.java.initmemory` property value to a larger value. For example, if the value is set to 256 (MB), you might double the value to 512 (MB).

   c. Set the `wrapper.java.maxmemory` property value to a larger value. For example, if the value is set to 512 (MB), you might double the value to 1024 (MB).

   Note: The `wrapper.java.maxmemory` property value must be equal to or greater than the `wrapper.java.initmemory` property value.

   d. Save your work.

Stopping the Operations Analytics Log File Connector for HP ArcSight Logger from Running as a Service

If you have a need to stop Operations Analytics Log File Connector for HP ArcSight Logger from running as a service, use one of the following methods to run the Operations Analytics Log File Connector for HP ArcSight Logger as a service.:

Method 1 (Command Line):

Note: You must run the `arcsight.bat` (Windows) and `arcsight` (Linux) scripts shown in this section as a user that has permission to remove a service from the server on which you run the command.

1. Navigate to the root installation directory.

   Note: This is the folder in which you copied, then unzipped, the `opsa-arcsight-connector-dist-windows.zip` or `opsa-arcsight-connector-dist-linux.zip` file.

2. Run the following command to stop Operations Analytics Log File Connector for HP ArcSight Logger from running a service.

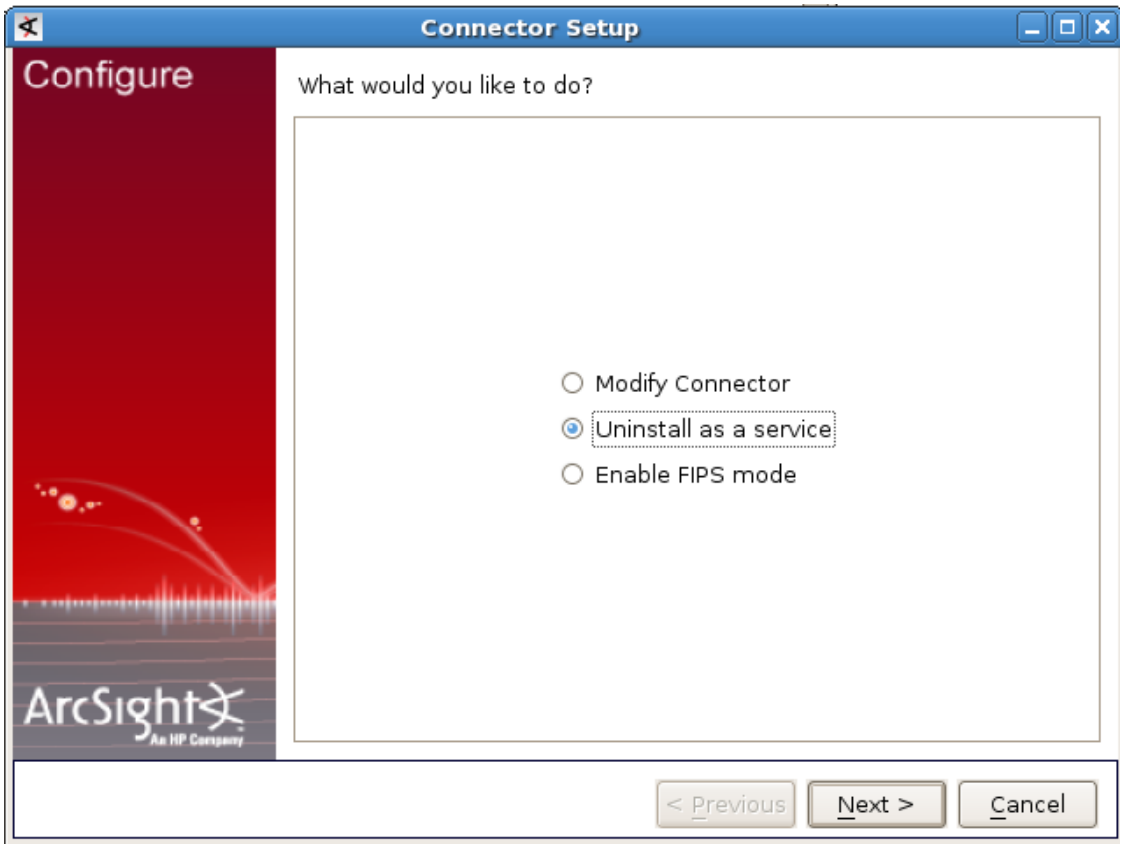
   - **Windows**: `current\bin\arcsight.bat agentsvc -r`
   - **Linux**: `current/bin/arcsight agentsvc -r`

Method 2 (User Interface):

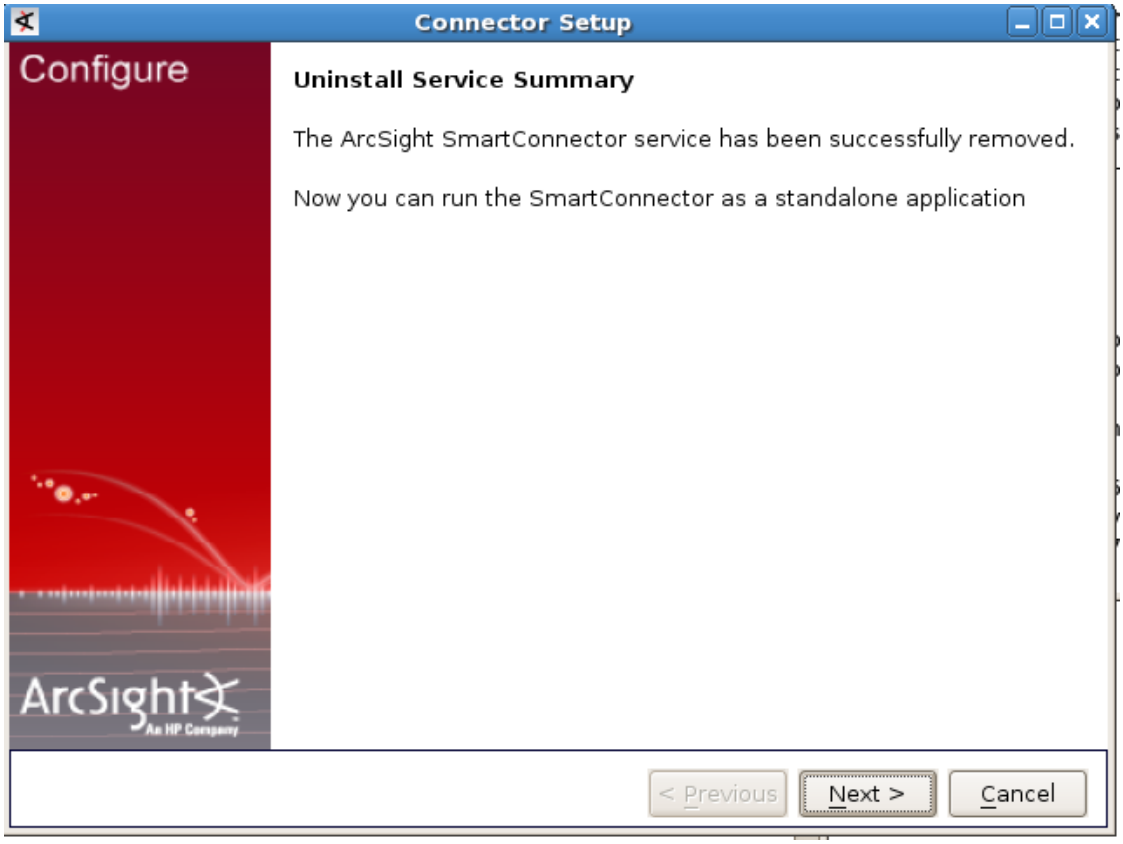
1. Navigate to the root installation directory. Run the following command to stop Operations Analytics Log File Connector for HP ArcSight Logger from running a service.

   - **Windows**: `".\bin\arcsight agentsetup.bat -c"`
   - **Linux**: `./bin/arcsight agentsetup -c`

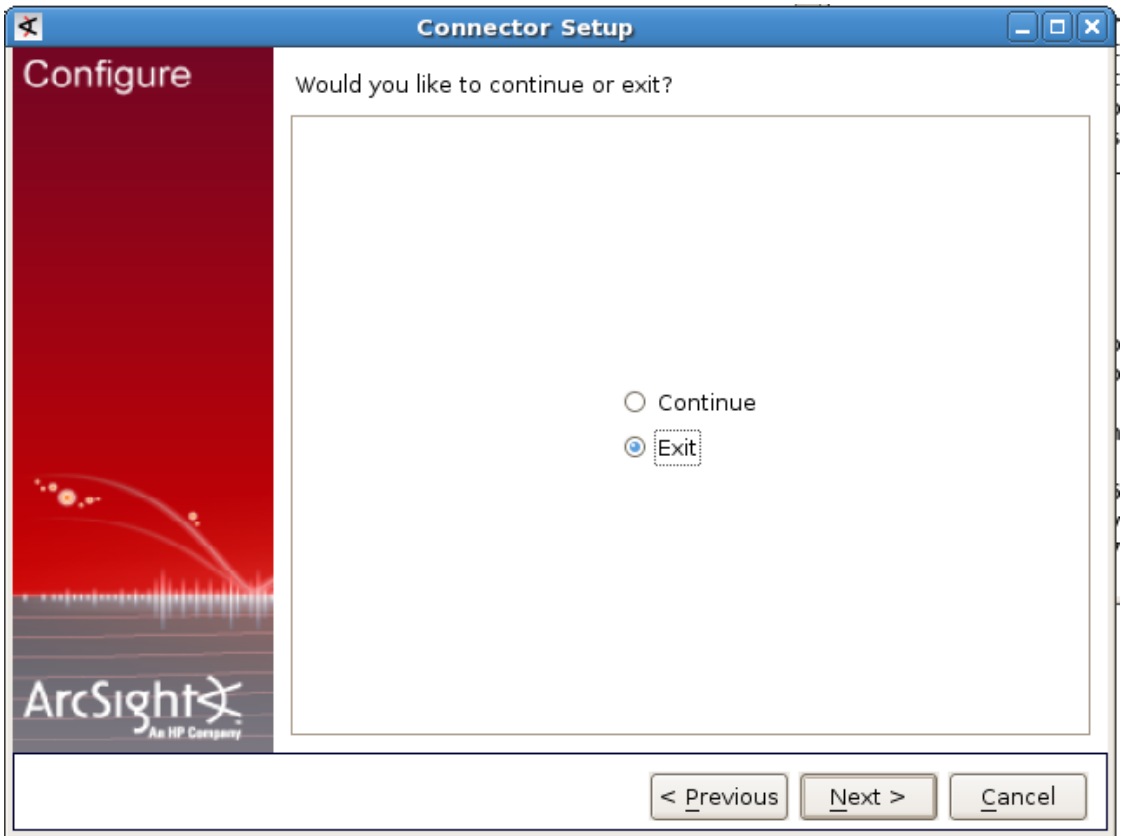
2. Select **Uninstall as a service**; then click **Next**.



3. After the service is successfully uninstalled, you should see the following message:



4. Select **Exit** ; then click **Next** to complete the installation.



Troubleshooting the Operations Analytics Log File Connector for HP ArcSight Logger

When troubleshooting the Operations Analytics Log File Connector for HP ArcSight Logger, look in the `[<InstallDir>]/current/logs` directory for log files associated with the connector's directory. Look for log entries that contain `WARN` or `ERROR`.

Problem: You do not see any log messages appearing in the HP ArcSight Logger UI from the Operations Analytics Log File Connector for HP ArcSight Logger.

Solution: This problem could be caused by a connection error between the Operations Analytics Log File Connector for HP ArcSight Logger and the server. Check for log entries containing `AgentLoggerSecureEventTransport` in the `[<InstallDir>]/current/logs/agent.log` file. If you see an entry similar to the following, there are connection issues between the connector and the HP ArcSight Logger server:

```
[2013-07-05 09:18:03,449][ERROR]
[default.com.arcsight.agent.loadable.transport.event._
AgentLoggerSecureEventTransport][openConnection] Connection to
```

```
[10.10.10.155] port 443 and receiver [My Smart Receiver] failed 0-event
message test
```

Problem: Log messages appear in the HP ArcSight Logger server, but some of the CEF fields are wrong or missing.

Solution: Make sure that the data you entered is valid by checking the `[<InstallDir>]/current/logs/agent.log` file for log entries containing `AgentSanityVerifier`. The following log entry shows an example involving `sourceHostName` not appearing in the HP ArcSight Logger server due to an invalid host name being entered when configuring the Operations Analytics Log File Connector for HP ArcSight Logger:

```
[2013-07-08 13:46:26,271][WARN ][default.com.arcsight.agent.loadable._
AgentSanityVerifier][checkHostNames] Invalid device host name encountered
[my Hostname]
```

As a best practice, always run the **Test Log Folders** option after configuring the Operations Analytics Log File Connector for HP ArcSight Logger.

Using Other ArcSight Connectors

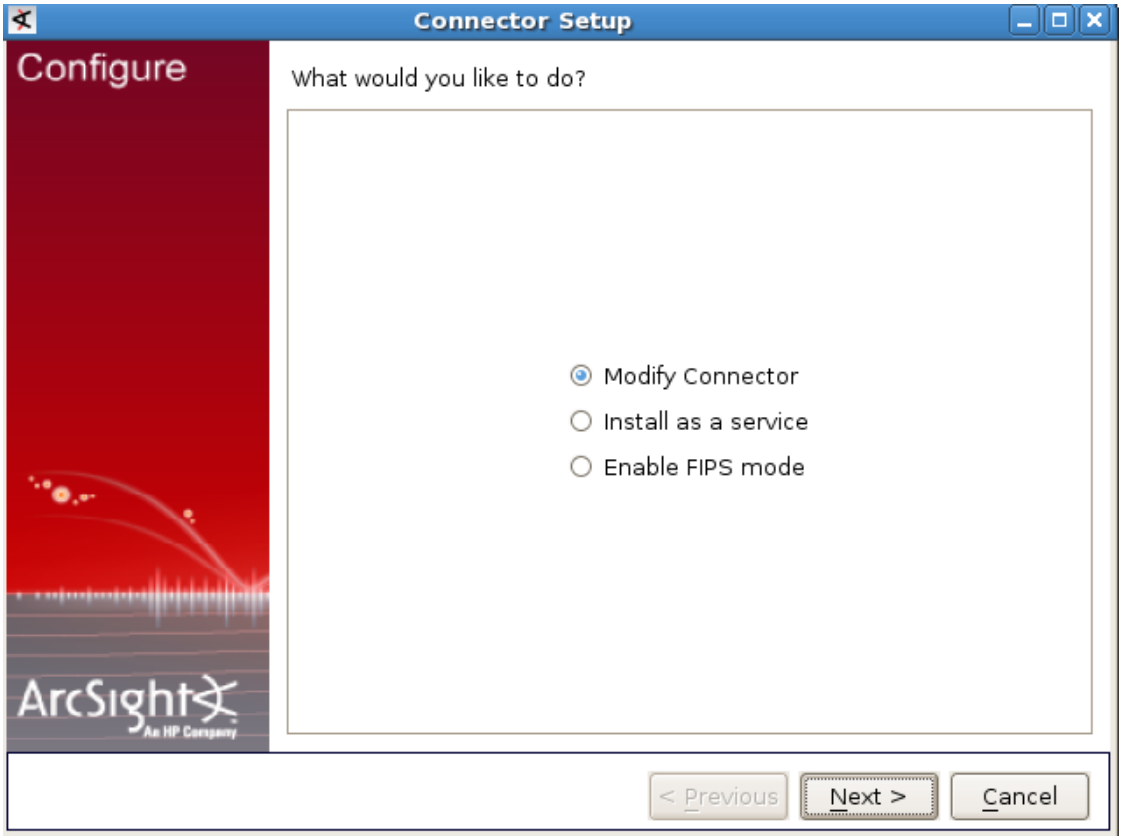
Raw Log Message

For OpsA to display a user friendly log message, it is recommended that you configure all ArcSight connectors to preserve the raw log message. If you prefer not to do this because of the extra cost of storing the raw message, then OpsA uses the Raw CEF string when it displays the log message.

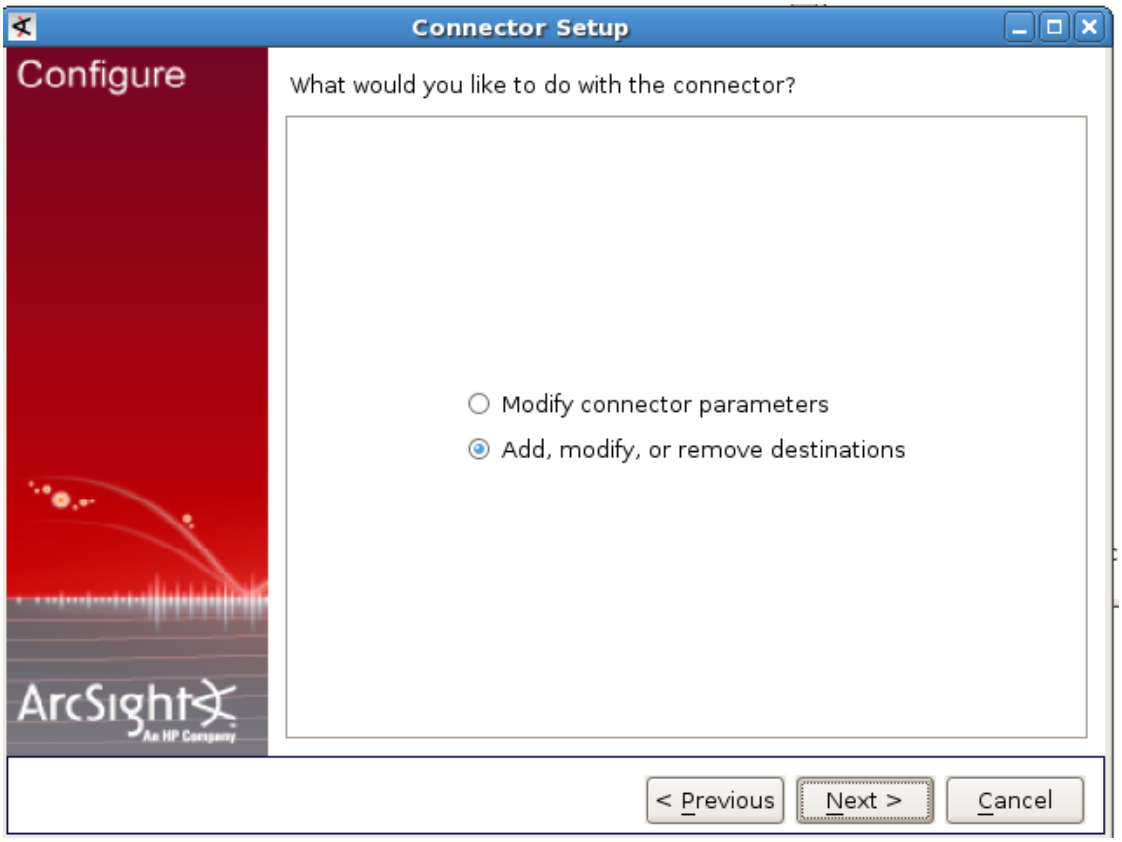
Do the following to set up the ArcSight connector to preserve the raw event:

1. Run the following command to start the setup script:
   - **Windows**: `<Connector Install Directory>/current/bin/runagentsetup.bat`
   - **Linux**: `<Connector Install Directory>/current/bin/runagentsetup.sh`

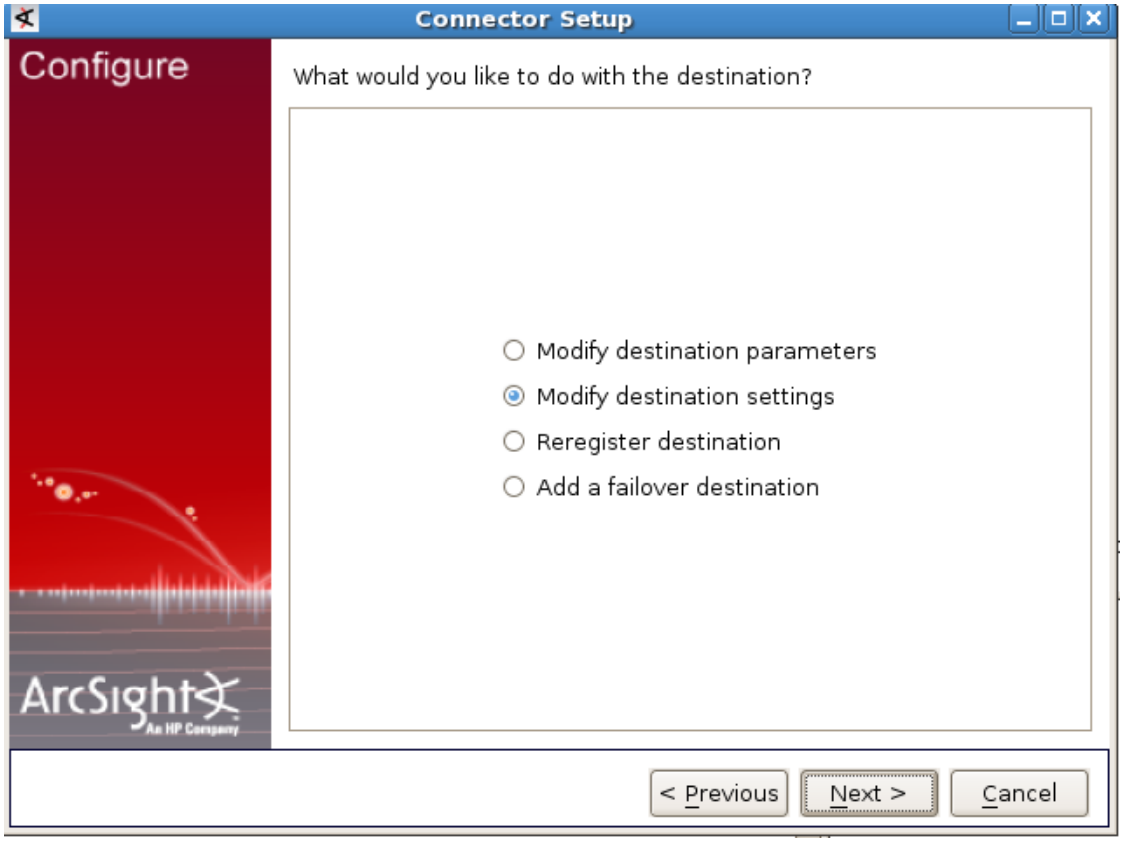
2. Select **Modify Connector**; then click **Next**.



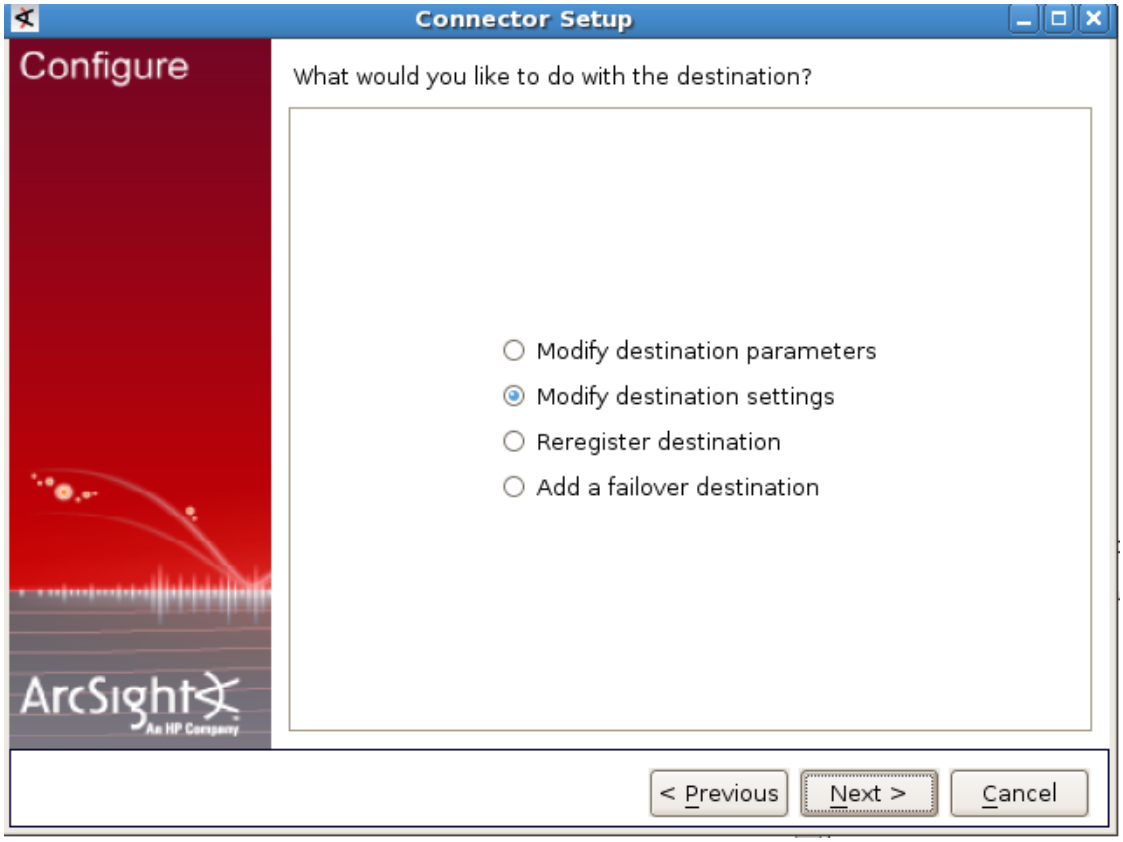
3. Select **Add, modify, or remove destinations**; then click **Next**.



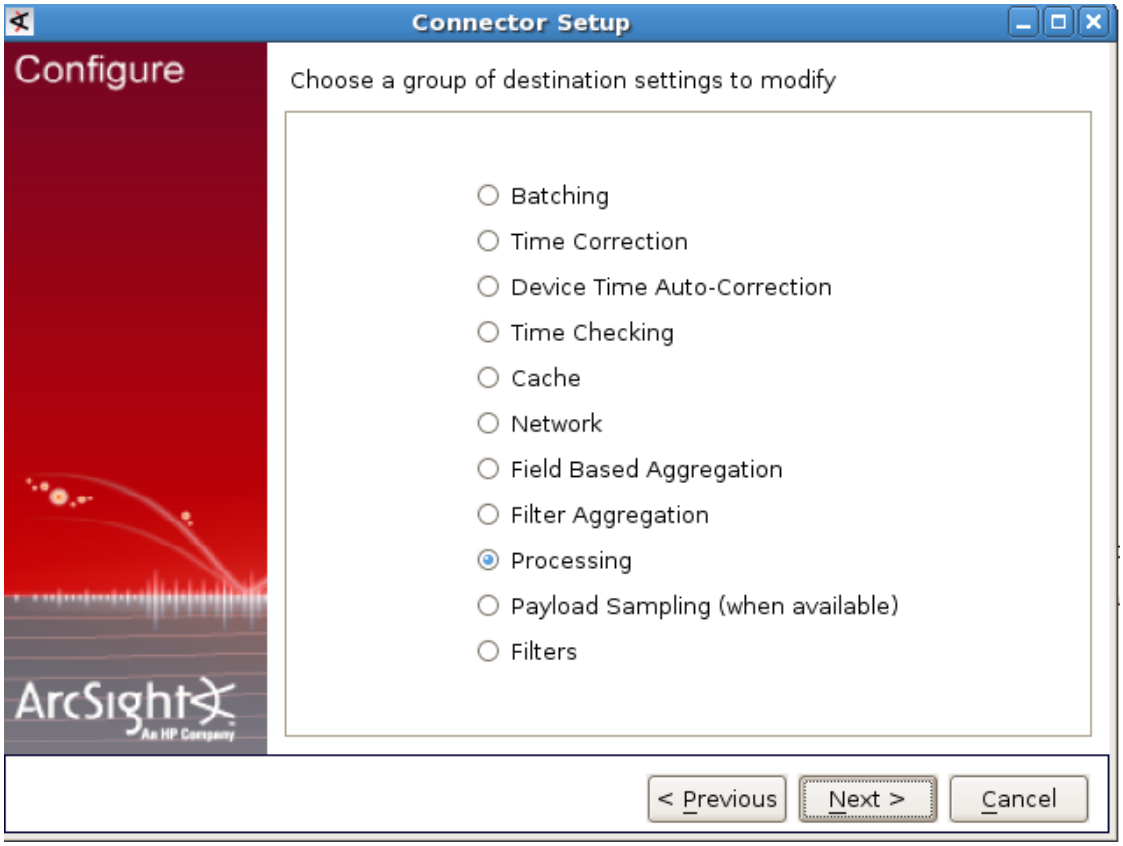
4. Select the destination to modify; then click **Next**.



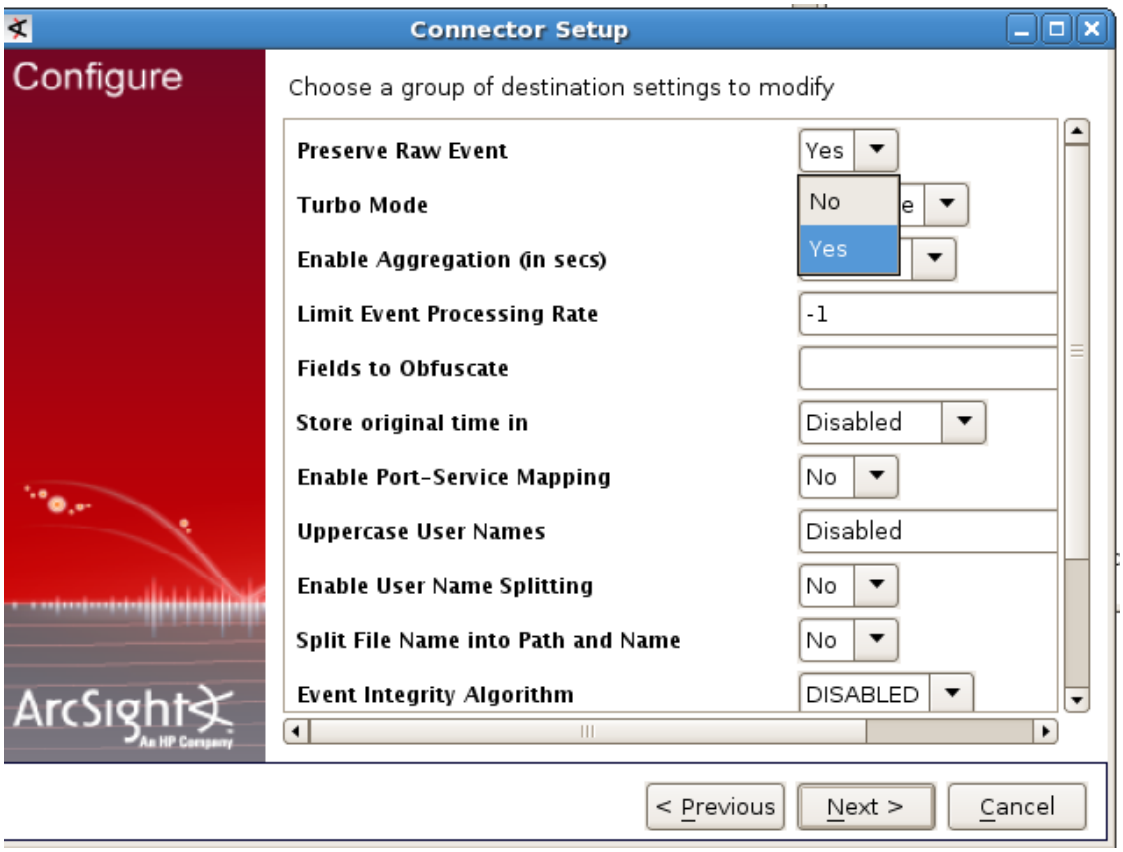
5. Select **Modify destination settings**; then click **Next**.



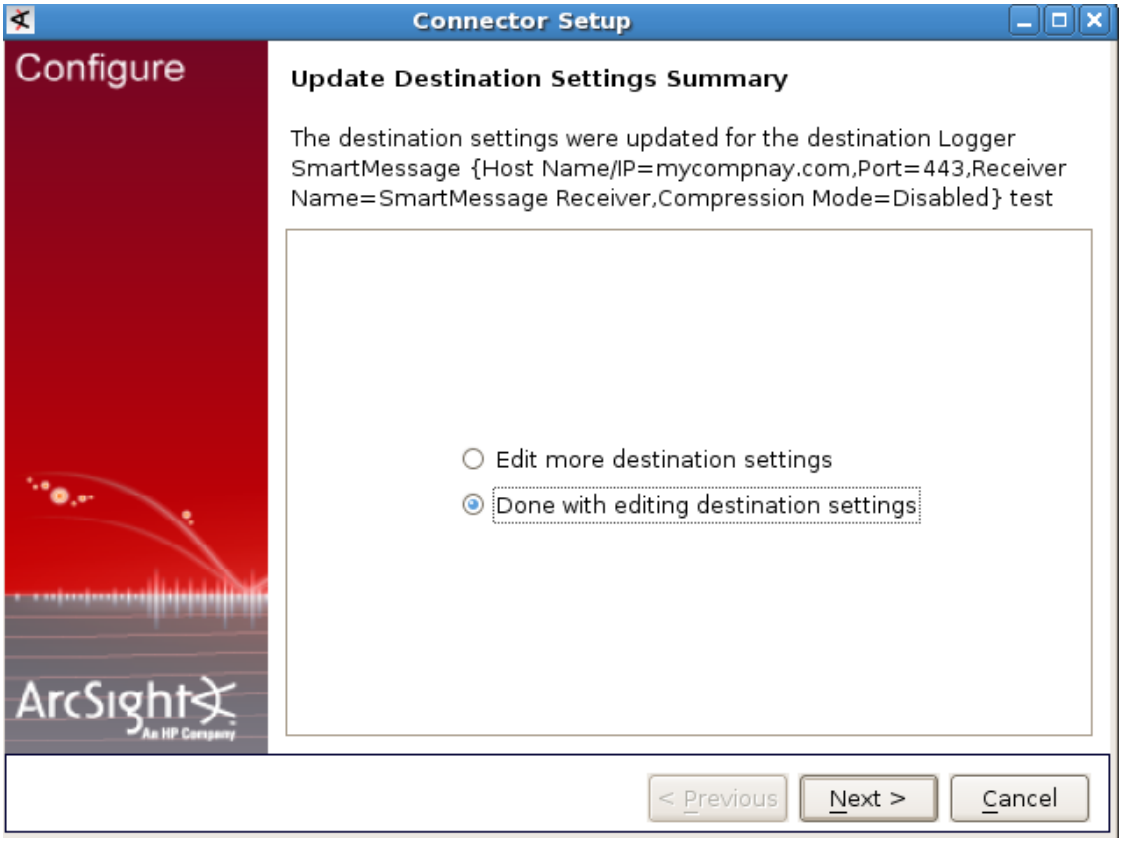
6. Select **Processing**; then click **Next**.



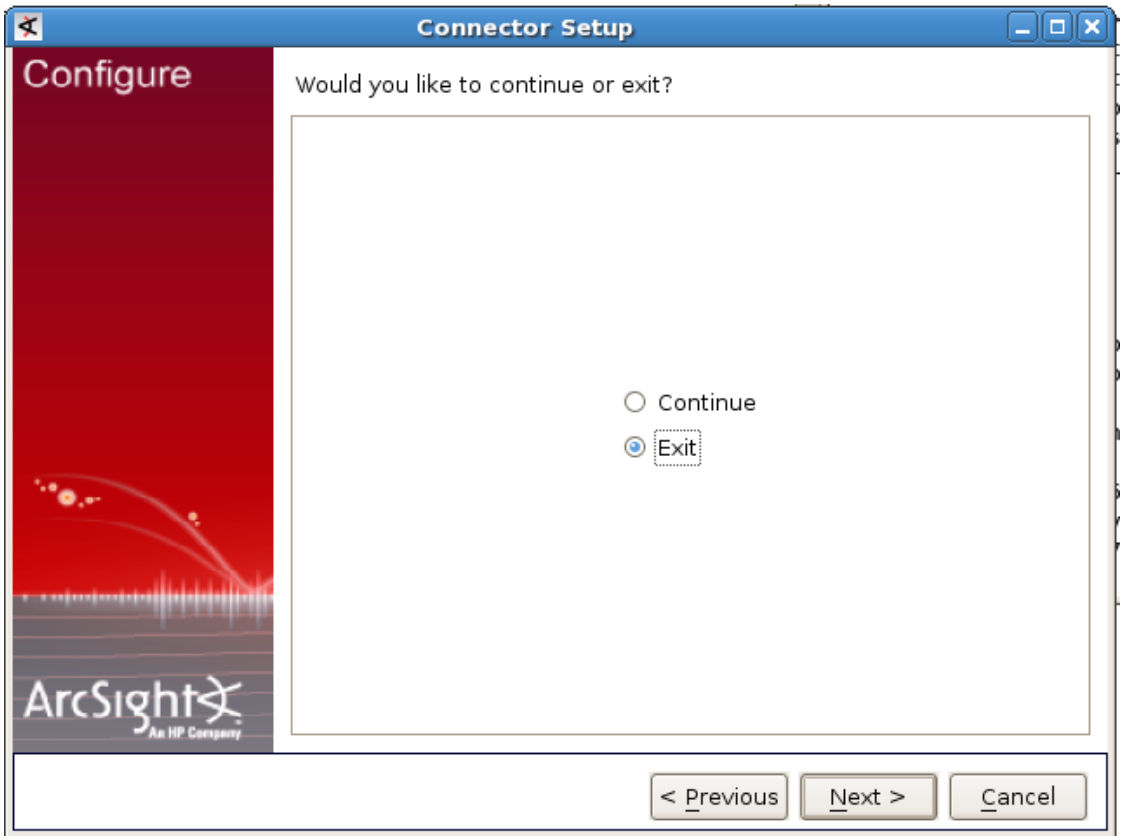
7. Set **Preserve Raw Event** to **Yes**; then click **Next**.



8. Click **Done with editing destination settings**; then click **Next**.



9. Select **Exit**; then click **Next**.



Setting Hostname, Application, and Process Names

If the ArcSight connector is not currently setting the `sourceHostName`, `sourceProcessName`, or `sourceServiceName` CEF fields, you can set these to the name of the process and application using the following steps:

If the ArcSight connector is not setting the CEF fields to store a process name, application name, or host name, you can correct this issue using the following steps:

1. Navigate to the `<Connector Install Director/current/user/agent/map` directory.
2. Identify all of the map property files in this directory (`map*.properties`), then identify the next available number that can be used. For example if the map property files listed were `map.0.properties map.1.properties` the next available number would be 2 (as in `map.2.properties`). This (next available) number is used to define the order in which the map files are processed by the ArcSight connector.
3. Create a file called `map.<next available number>.properties`.
4. Insert the following entries into the `map.<next available number>.properties` file:

a. Add the CEF fields you want to set as the first line in the file. For example, to set all three of the CEF fields (`set.event.sourceProcessName`, `set.event.sourceHostName`, and `set.event.sourceServiceName`), add the following line as the first line in the file: `set.event.sourceProcessName, set.event.sourceHostName, set.event.sourceServiceName`.

b. The second line contains the static values to be set for the CEF fields you just defined in the first line. For example, add the following line to the file to set the `sourceProcessName`, `sourceHostName`, and `sourceServiceName` CEF fields to `MyHostname`, `MyProcess`, and `MyApplication` respectively: `MyProcess, MyHostname.com, MyApplication`

Note: Example:
```
set.event.sourceProcessName,set.event.sourceHostName,set.event.sou
rceServiceName
MyProcess,MyHostname.com,MyApplication
```

Uninstalling the Operations Analytics Log File Connector for HP ArcSight Logger

To uninstall the Operations Analytics Log File Connector for HP ArcSight Logger, do the following:

1. Stop the Operations Analytics Log File Connector for HP ArcSight Logger before continuing. See "Manually Starting and Stopping the Operations Analytics Log File Connector for HP ArcSight Logger " on page 180 for more information.
2. If you configured the Operations Analytics Log File Connector for HP ArcSight Logger to run as a service then remove that service. See "Stopping the Operations Analytics Log File Connector for HP ArcSight Logger from Running as a Service " on page 186 for more information.
3. Remove the root installation directory.

Note: This is the folder in which you copied, then unzipped, the `opsa-arcsight-connector-dist-windows.zip` or `opsa-arcsight-connector-dist-linux.zip` file.

Exporting and Importing Operations Analytics Dashboards

An Operations Analytics (OpsA) dashboard is the graphical user interface for troubleshooting your IT operations problems. See *Dashboards and Query Panes* in the *Operations Analytics help* for more information.

After you create new dashboards or modify existing ones, you might want to export these dashboards to a file, then import them for use among tenants.

Caution: Do not edit the dashboard file (shown as `mydashboard.xml` file in the examples in this section) after you export it, then attempt to import the file. Manually editing an exported dashboard file is not supported.

Note: If you choose to add spaces to your dashboard names, such as using two or more words in your dashboard names, you must always use quotation marks when working with these dashboards.

Exporting and Importing Dashboards Among Operations Analytics Tenants

Suppose you created a new dashboard, dashboard1, and want to export this dashboard and share it with another OpsA tenant. You can use the instructions in this section to import these dashboards to another tenant.
To accomplish this, do the following:

1. From the Operations Analytics console, navigate to the **Dashboards** menu.
2. While viewing the dashboards, make a list of the dashboards that you want to export. For this example, you have dashboard1 and dashboard2 on your list.

   Note: Dashboards can be **shared** with other users in your user community. See *Share a dashboard with other users in your user community* in the *Operations Analytics help* for more information. The instructions in this section work for both shared and non-shared dashboards.

3. Run the following command to export dashboard1:
   `opsa-dashboard-manager.sh -u <the user that owns the dashboard> -e dashboard1 -f <mydashboardfile>`

   Note: To export more than one dashboard, use `-e dashboard1 dashboard2`.

   Note: The `opsa-dashboard-manager.sh` script exports the dashboard to the current directory. For example, if you run the `opsa-dashboard-manager.sh` script from the `$OPSA_HOME/bin` directory, look for the exported dashboards in the `$OPSA_HOME/bin` directory.

Note: You can use variations of the opsa-dashboard-manager.sh to export specific dashboards or all dashboards. See the opsa-dashboard-manager reference page (or the Linux manpage) for more information.

Note: If you create dashboard names that include spaces, you must wrap those dashboard names in double quotation marks. For example, wrap any dashboard names that include white space as shown in the bold font: `opsa-dashboard-manager.sh -u opsa -p opsa -e` **`"metrics dashboard"`** `-f mydashboardfile`

4. To import your exported dashboard or dashboards to another OpsA installation, run the following command from the OpsA Server Appliance on which you want to import your dashboards: `opsa-dashboard-manager.sh -u <`*`the user that will own the dashboard or dashboards`*`> -i -f <`*`mydashboardfile`*`>`

   Note: The `opsa-dashboard-manager.sh` script prompts you for the password for the opsatenantadmin password, which you set during installation.

   Note: Before you import dashboards, it is a good practice to create a backup copy of any dashboards you plan to import. See *Copy a dashboard* in the *Operations Analytics help* for more information.

5. To see the newly imported dashboards, you must run the following command from the OpsA Server Appliance on which you imported the dashboards: `$OPSA_HOME/bin/opsa-server restart`

   See the *opsa-server* reference page (or the Linux manpage) for more information.

After you successfully import a dashboard, users associated with the tenant you used for the import should be able to see data using the imported dashboard.

Log Files in Operations Analytics

This information in this section discusses the purpose and location of log files used in Operations Analytics (OpsA).

Using and Maintaining Audit Log Files

The information in this section discusses the log files Operations Analytics (OpsA) provides for auditing events associated with account and application activity. This audit activity does not include any information that might be considered sensitive in nature. OpsA logs information related to the following topics:

- REST (Representational state transfer) calls
- Log on requests
- User setting changes
- Administrator setting changes
- Users attempting to log on without OpsA roles
- Users attempting to use unauthorized resources
- Users accessing administrative consoles
- Create, delete, or disable user accounts
- Lock or release user accounts
- Password resets

Audit logs for the OpsA Server Appliance reside in the following location:
`$OPSA_HOME/log/audit/opsa-server-audit.log`

These audit logs are configured for read and write permissions for the opsa user, and cannot be edited by other users.

There are several logging levels supported by the OpsA audit logs. The following list is in order from the least severity to the most severity.

- INFO
- LOW
- MEDIUM
- HIGH
- CRITICAL

To change the level of logging of the OpsA Server Appliance, edit the following file and follow the instructions shown in the file: **OpsA Server Appliance**:
`$OSPA_HOME/jboss/standalone/configuration/standalone.xml`

Note: Back up the standalone.xml file before doing any editing. Carefully edit this file, keeping the xml well-formed and valid.

For example, to turn off logging, do the following.

1. Edit the following file on the OpsA Server Appliance:
   `$OSPA_HOME/jboss/standalone/configuration/standalone.xml`
2. Look for xml content that resembles the following:

```
<subsystem xmlns="urn:jboss:domain:logging:1.2">
<periodic-rotating-file-handler name="AUDIT_FILE">
<level name="INFO"/>
 <formatter>
 <pattern-formatter pattern="%d{yyyy-MM-dd HH:mm:ss,SSS} %s%E%n"/>
</formatter>
 <file relative-to="jboss.server.log.dir" path="../../../audit/opsa-
server-audit.log"/>
<suffix value=".yyyy-MM-dd"/>
 <append value="true"/>
</periodic-rotating-file-handler>
 <logger category="com.hp.opsa.common.audit" use-parent-
handlers="false">
<handlers>
<handler name="AUDIT_FILE"/>
 </handlers>
</logger>
</subsystem>
```

3. Change the **INFO** text to **OFF**; then save your changes.
4. Run the following command to apply your changes: `$OPSA_HOME/bin/opsa-server restart`

Maintaining the Operations Analytics Database

Use the instructions in this section to maintain the OpsA databases.

Backing up and Restoring Data

To back up or restore data for the Operations Analytics (OpsA) Server and Collector Appliances, see the referenced sections in the following documents:

- The *Configuration Backup and Restore* section of the *ArcSight Logger Administrator's Guide*
- The *Backing up and Restoring the Database* section of the *Vertica Administrator's Guide*

Managing Vertica Data

By default, the OpsA uses the Vertica Community Edition license, which is a non-expiring 1TB license. To avoid any disruptions in service, it is a good practice to monitor the size of the OpsA database.

To check or verify the size of the OpsA database, do the following:

1. Log on to the Vertica server as a `root` or `dbadmin` user.
2. Run the following command: `/opt/vertica/bin/vsql -U dbadmin -c 'select get_compliance_status();'`

   Note: Only use the `-U dbadmin` option if you log on as a root user.

3. Review the compliance status. The message you see resembles the following example, which shows a 70 percent utilization percentage (70 percent of the 1TB that is available is currently in use):

```
                              get_compliance_status
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
----------
Raw Data Size: 0.00TB +/- 0.00TB
License Size : 1.00TB
Utilization  : 70%
Audit Time   :      -12-31 17:00:00-07
Compliance Status : The database is in compliance with respect to raw data size.

No expiration date for a Perpetual license
(1 row)
```

   If you have exceeded your licensed database size, do one or more of the following:

   - **Shorten the data retention period:** See "Setting Collection Retention Periods" on the next page for more information.
   - **Set a Purge Policy for the Vertica database**: See *Purging Deleted Data* in the *Vertica Administrator's Guide*.
   - **Manually purge data from the Vertica database**: See *Purging Deleted Data* in the *Vertica Administrator's Guide*
   - **Increase the Vertica license size**: See *Managing Licenses* in the *Vertica Administrator's Guide*

See *Monitoring Database Size for License Compliance* in the *Vertica Administrator's Guide* for more information.

Setting Collection Retention Periods

By default, OpsA's distributed version includes a three month data retention period . After purchasing and applying a production license, you can modify the data retention period as follows:

You can set the amount of time that Operations Analytics (OpsA) retains the data it is collecting. You can set the retention period for a collection or for all of the collections belonging to a tenant or a data source.

To set the amount of time to retain the data for a collection, use the following command:
`$OPSA_HOME/bin/opsa-collection-config.sh -setretention <retention period in months> –source <source name> -domain <domain name> -group <group name> -username opsatenantadmin`
See the *opsa-collection-config.sh* reference page (or the Linux manpage), for more information.

The following shows several examples of setting collection retention periods:

- To set the retention period for a specific source, domain, and group, use the following command: `/opt/HP/opsa/bin/opsa-collection-config.sh -setretention <retention period in months> -source <source> -domain <domain> -group <group> -username <username> [-force]`
- To set the retention period for a specific source, use the following command: `/opt/HP/opsa/bin/opsa-collection-config.sh -setretention <retention period in months> -source <source> -username <username> [-force]`
- To set the overall retention period, use the following command: `/opt/HP/opsa/bin/opsa-collection-config.sh -setretention <retention period in months> -username <username> [-force]`

Note: When setting retention period for multiple collection policies, you can use the `-force` option to forcefully set the retention name and to avoid responding with `yes` for each collection.

After setting the retention period for specific collections belonging to a tenant, OpsA removes any data record with a time stamp older than the listed retention period for those collections.

Managing Data in Logger

Logger supports several storage groups, each of which can have a different retention policy. Retention policy is specified in terms of number of days that events are stored, or overall maximum size (in GB). See *Retention Policy* in the *ArcSight Logger Administrator's Guide* for more information.

For software Loggers, the storage volume is set to the maximum capacity specified in the license or the available disk space, whichever is smaller. See *Storage Volume* in the *ArcSight Logger Administrator's Guide* for more information.

To manage adherence to the Logger license, do the following from the ArcSight Console:

1. Click **System Admin**.
2. Click **License & Update**.
3. Review the license information. If you have exceeded your licensed database size, do one or more of the following:
   - Increase the Logger licensed capacity. See *License and Update* in the *ArcSight Logger Administrator's Guide* for more information
   - Add a storage group to Logger. See *Adding Storage Groups* in the *ArcSight Logger Administrator's Guide* for more information
   - Regularly manage your Logger storage volume. See *Storage Volume* in the *ArcSight Logger Administrator's Guide* for more information

Modifying Unit Scaling on Collected Data

Data can be displayed in different scales, for example 1000 bytes may be displayed as 1 kilobyte or 1000 bytes. This procedure shows you how to modify the way data is displayed in query panes.

1. Open the configuration file of the collection from which the data originates. The files are found in the **/opt/HP/opsa/conf/collection/server/config.templates** directory.

   Example:
   /opt/HP/opsa/conf/collection/server/config.templates/bpm/1.0/application/performance/bpm_collection.xml

2. Locate the name of the metric you want to modify. In the example below, this is **Transaction_ Response_Time**. Add a scaling_unit element using the following options:

   %,mbps,kbps,gbps,kb,mb,gb,hz,khz,mhz,ghz,bytes,BIT,PB,EB,W,V,A,secs,millisecs,ms,pages/sec,per second,switches/sec,bytes/sec,KB/sec,interrupts/sec,packets/sec,errors/sec,reads/sec,bps,per hour,per min

   then specify the factor that you want to multiply the incoming data by.

   Example: This example takes incoming milliseconds and displays them as seconds.

   <collection sourcegroup="performance" ……

   <column name=**"Transaction_Response_Time"** position="9" datatype="float" length="0" key="no" value="" mapsto="“ label="Transaction Response Time" columnname="" unit="ms“ **scaling_unit="secs" factor="0.001“** type="metric"/> </collection>

3. Run the create and publish commands on the collection.

Example:

opt/HP/opsa/bin/opsa-collection-config.sh -create -nodelist /opt/HP/opsa/conf/collection/sample/bpm_nodelist -collectorhost 1.2.3.4 -source bpm -domain application -group performance -username <admin username> -password <admin password>

/opt/HP/opsa/bin/opsa-collection-config.sh -publish -collectorhost 1.2.3.4 -username <admin username>-password <admin password>

Monitoring Operations Analytics Processes

OpsA provides the `opsa` script to check status or control OpsA services. See the `opsa` reference page (or the Linux manpage) for more information.

OpsA depends on several different services to be active on the deployed OpsA Collector and Server Appliances. These services start automatically when booting up your OpsA Collector, Server, or All-In-One appliances (you do not need to specifically configure these services).

You can use the `opsa` script to do several things:

- Run the `$OPSA_HOME/bin/opsa status` command script to check the status of all of these services at once.
- When necessary, you can control all of the OpsA-related services on any OpsA appliance (for example, in preparation for installing a software patch):
  - Start OpsA services: `$OPSA_HOME/bin/opsa start`
  - Stop OpsA services: `$OPSA_HOME/bin/opsa stop`

Note: A network disruption can cause OpsA services to stop functioning. If you suspect that the OpsA Server or Collector Appliances lost connectivity to the network, you might restart them using `$OPSA_HOME/bin/opsa restart` command.

As a less powerful alternative, OpsA) also provides the `opsa-process-manager` script to stop and start processes on a single OpsA Server or Collector Appliance. You can also use the `opsa-process-manager` script to monitor OpsA processes. See the `opsa-process-manager` reference page (or the Linux manpage) for more information.

Note: A network disruption can cause this process management feature to stop functioning. If you suspect that the OpsA Server or Collector Appliances lost connectivity to the network, restart them as detailed in "Restarting the Operations Analytics Server and Collector Appliance" on the next page after the network connectivity is restored.

Restarting the Operations Analytics Server and Collector Appliance

Restarting the OpsA Server and Collector Appliances: If you suspect that Vertica has stopped functioning (such as during a power outage, network outage, or other software disruption), you can restart the OpsA services on the OpsA Server and Collector Appliances. The symptom you might see is that new data is no longer being collected with old data still available for viewing.

To restart the OpsA Server and Collector Appliances, do the following on each server: `$OPSA_HOME/bin/opsa restart`

See the *opsa* reference page (or the Linux manpage) for more information.

Using Parametric Dashboards

OpsA provides you with the ability to use dashboards as launching points for parametric dashboards. For example, assume you have an existing dashboard, called *MyDashboard*. From *MyDashboard*, assume you want to navigate to launch a parametric dashboard, called *MyParametricDashboard*.

Note: This section discusses using an existing dashboard to implement a navigation (drill) to a parametric dashboard. The use of parametric dashboards is only one form of drill. There is a PQL drill as well, and that drill is not covered in this section.

To configure this drill point, do the following:

1. Using the *MyDashboard* dashboard, click **Parameters** in the panel and scroll down. To determine the parameters you want to enter, do the following:

    a. **Drill Destination**: Decide what name you want to use for the dashboard to which you want to drill (*MyParametricDashboard*).

    b. **Drill Label Field** and **Drill Value Field**: Identify the parameter names for these panels. To obtain these values, do the following:

        i. Navigate to the *MyParametricDashboard* dashboard.

        ii. Click **Edit Pane Settings**.

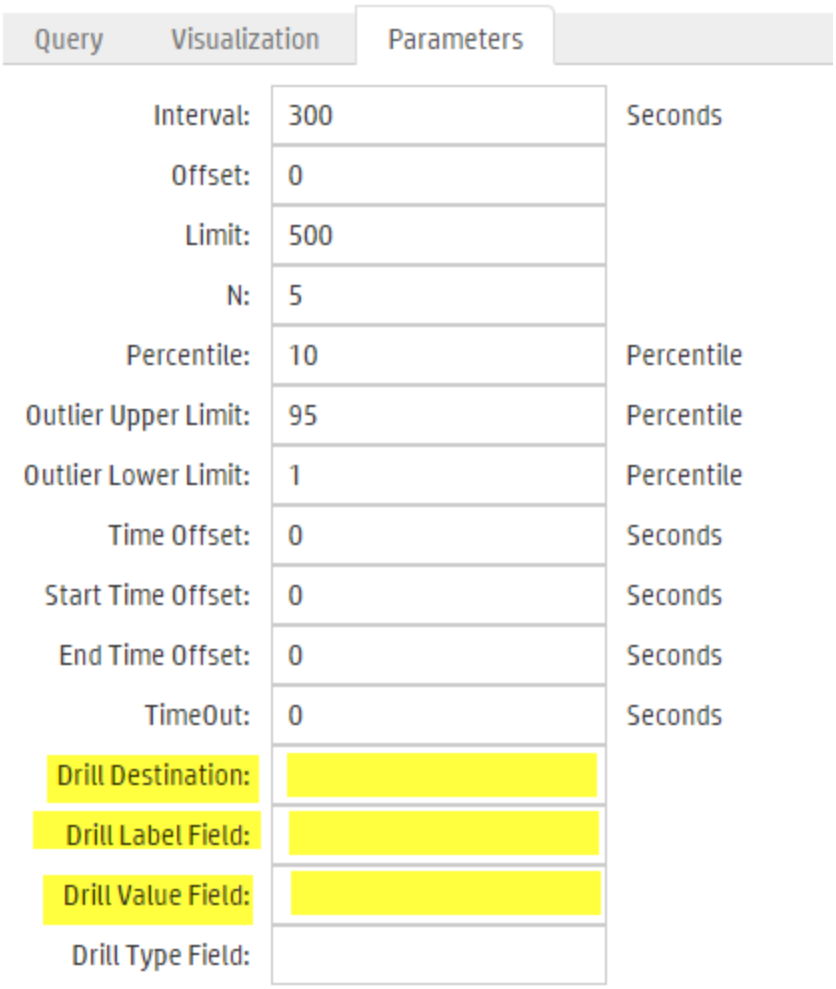
        iii. Scroll down; then click **Show Properties**.

        iv. If necessary, enter a filter to limit the number of entries you want to review.

        v. Write down the following values for the **Drill Label Field** and **Drill Value Field**:

- **Drill Label Field**: Use a combination of the `property group id` and `property uid` (concatenated by an underscore (_) for the value in this field).
- **Drill Value Field**: Use a combination of the `property group id` and `property uid` (concatenated by an underscore (_) for the value in this field).

2. Open the *MyDashboard* dashboard, click **Parameters** in the pane, then scroll down. Note the highlighted fields shown below.



3. Enter the values you identified for the **Drill Destination**, **Drill Label Field**, and **Drill Value Field**.
4. **Save** your work.
5. Open the *MyParametricDashboard* dashboard for editing.
6. In the AQL, use the following as an example to edit the AQL: Change *<the observed string or ID*

string> to read as `param1`.

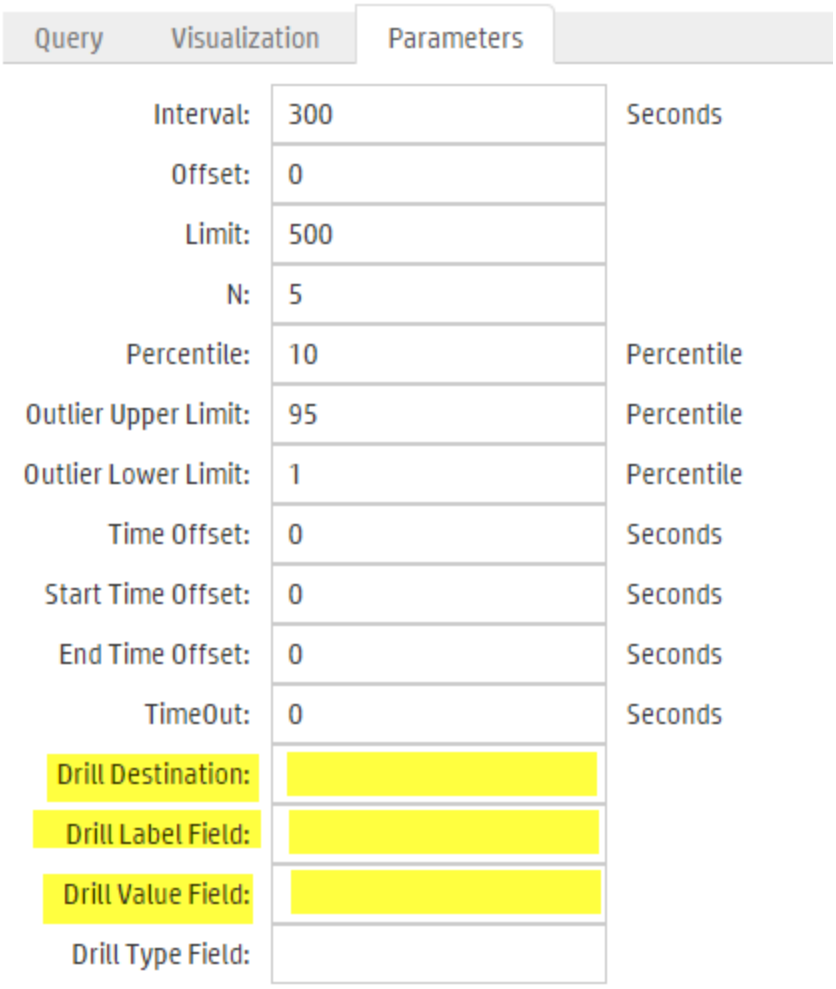
Note: The <*the observed string or ID string*> represents the parameter you are passing into the *MyParametricDashboard* dashboard. You are limited to passing one parameter.

Note: After you replace the ID string with `param1`, the dashboard is now a "template" dashboard. Any instantiation of this dashboard will not be editable.

After completing the above steps, you can drill to the *MyParametricDashboard* dashboard from a link within the *MyDashboard* dashboard.

Below is a simple example, using a dashboard you created, *MyAlertsDashboard*, and configuring it to drill to a parametric dashboard, *MyAlertsInstance*, that you created.

1. Using the *MyAlertsDashboard* dashboard, click **Parameters** in the panel, and scroll down. Note the highlighted fields shown below.



To determine the parameters you want to enter, do the following:

a. **Drill Destination**: Decide what name you want to use for the dashboard to which you want to drill. For this example, you are using *MyAlertsInstance*.

b. **Drill Label Field** and **Drill Value Field**: Identify the parameter names for these panels. To obtain these values, do the following:

   i. Navigate to the dashboard to which you want to drill. For this example, you are using *MyAlertsInstance*.

   ii. Click **Edit Pane Settings**.

   iii. Scroll down; then click **Show Properties**.

   iv. If necessary, enter a filter to limit the number of entries you want to review. For this

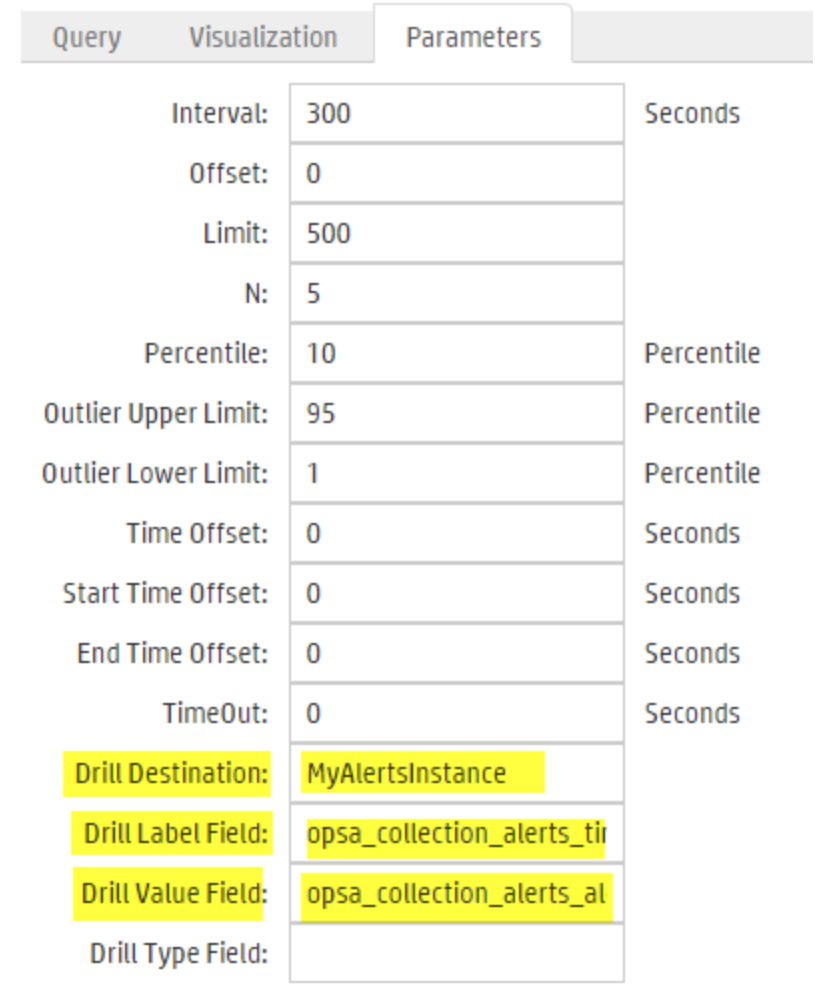
example you can enter `alerts` in the filter to limit the entries.

v. Write down the following values for the **Drill Label Field** and **Drill Value Field**:

- **Drill Label Field**: For this example, you want to use a time stamp for this field. Use a combination of the `property group id` and `property uid` (concatenated by an underscore (_) for the value in this field). For this example, you decide to use `opsa_collection_alerts_timestamp` as a value for this field.
- **Drill Value Field**: For this example, you want to use an identifier for the instance for this field. Use a combination of the `property group id` and `property uid` (concatenated by an underscore (_) for the value in this field). For this example, you decide to use `opsa_collection_alerts_alert_instance_id` as a value for this field.

2. Open the *MyAlertsDashboard* dashboard, click **Parameters** in the pane, and scroll down.

3. Enter the values you identified for the **Drill Destination** (*MyAlertsInstance* for this example), **Drill Label Field** (`opsa_collection_alerts_timestamp` for this example), and **Drill Value Field** (`opsa_collection_alerts_alert_instance_id` for this example). Note the highlighted fields shown below.



4. **Save** your work.

5. Open the *MyAlertsInstance* dashboard for editing.

6. In the AQL, use the following as an example to edit the AQL: Change <*the observed string or ID string*> to read as `param1`. For this example, change `{((i.alert_instance_id ilike "902496"))}` to `{((i.alert_instance_id ilike param1))}`

> **Note:** The <*the observed string or ID string*> represents the parameter you are passing into the *MyAlertsInstance* dashboard. You are limited to passing one parameter.

Note: After you replace the ID string with `param1`, the dashboard is now a "template" dashboard. Any instantiation of this dashboard will not be editable.

After completing the above steps, you can drill to the *MyAlertsInstance* dashboard from a link within the *MyAlertsDashboard* dashboard.

Additional Information

The appendices in this section provide some additional detailed information about Operations Analytics.

Much of the information in this section discusses how to configure collections using tools such as the `opsa-collection-config.sh` script that originated with Operations Analytics 2.0. Since the Operations Analytics 2.0 release, other configuration tools such as the Collections Manager and the `opsa-collection-setup.sh` script were developed that made collection configuration easier. If you must use these older tools, see the instructions in this section.

Configuring an HP Operations Smart Plug-in for Oracle Collection (Detailed Methods)

It is recommended that you use a more automated method of configuring this collection. See "Configuring an HP Operations Smart Plug-in for Oracle Collection" on page 27 for the recommended method. For a more detailed approach, you can complete the following steps for the HP Operations Smart Plug-in for Oracle Collection.

There are two methods of configuring Operations Analytics (OpsA) for HP Operations Smart Plug-in for Oracle, the **Automated Configuration Method** and the **Manual Configuration Method**. You must select one of these methods to configure OpsA for HP Operations Smart Plug-in for Oracle. Do not attempt to use both the Automated Configuration Method and the Manual Configuration Method.

After you complete the steps in this section, the HP Operations Smart Plug-in for Oracle Collection collects metrics every 15 minutes, with 5 minute data granularity.

> **Note**: An OpsA Collector Appliance can reliably collect data from approximately 5,000 nodes being monitored by the HP Operations Agent and the HP Operations Smart Plug-in for Oracle.

Automated Configuration Method: This approach is more automated than the Manual Configuration Method, making configuration easier. Do the following to use the Automated Configuration Method:

1. The Automated Configuration Method provides the sample collection template, `$OPSA_HOME/conf/collection/sample/sample_auto-config-node.properties`, located on the OpsA Server Appliance. Copy the `sample-auto-config-node.properties` file to a separate location, then edit the `sample-auto-config-node.properties` file and add, among other information, the following Operations Manager information.

   > **Note:** The sample file mentioned in this step does not contain a password property. When using the `opsa-collection-config.sh` script (in a later step) to encrypt the password, this script prompts you for the password, encrypts it, and inserts it into the sample file.

- For an OMW host (HP Operations Manager on a Windows server), the Automated Configuration Method includes the OMW host's fully-qualified domain name and its database details.
- For an OMU host (HP Operations Manager on a UNIX or Linux server), the Automated Configuration Method includes the OMU host's fully-qualified domain name and its database details.

2. Save your work.
3. Run the following command from the OpsA Server Appliance to encrypt the passwords in the `mynodelists.properties` file:
   `$OPSA_HOME/bin/opsa-collection-config.sh -encrypt <path>/sample-auto-config-node.properties`

   Note: You will be prompted to enter the password after running this command.

4. Run the following command from the OpsA Server Appliance to configure the collector hosts and publish the collection information:

   `$OPSA_HOME/bin/opsa-collector-auto-conf.sh -nodelist <path>/<sample-auto-config-node.properties> -collectevents -configuredomain [ORACLE|SYSTEM] -username opsatenantadmin`

   Note: The `opsa-collector-auto-conf.sh` script sets up HP Operations Agent, HP Operations Smart Plug-in for Oracle, and OM event collection using only one command.

   Note: The command output shows the list of registered collectors and distributes the list of nodes providing system performance metrics, Oracle performance metrics, or both among the registered collectors.

   Note: The `opsa-collector-auto-conf.sh` script prompts you for the Tenant Admin password for the username you use in the commands shown in this section. See the *opsa-collector-auto-conf.sh* reference page (or the Linux manpage) for more information.

Manual Configuration Method: Do the following for a more manual approach to configuring OpsA for HP Operations Smart Plug-in for Oracle:

1. A node list file contains details about the sources from which you plan to collect information. There are sample nodelist files located in the `$OPSA_HOME/conf/collection/sample` directory. OpsA administrators can use these sample files to publish the node list file. The sample node list file for the HP Operations Agent Smart Plug-in for Oracle collection is `sample_oa_pa_node.properties`. The node list file for the HP Operations Agent Smart Plug-in for Oracle

collection must contain a list of servers that have the HP Operations Agent and the HP Operations Smart Plug-in for Oracle installed. The node list file for the HP Operations Agent Smart Plug-in for Oracle collection must include the information shown in the following table:

Node List File Fields and Values

Field	Value
oanode1.somedomain.com	The fully-qualified domain name of a server that has the HP Operations Agent and the HP Operations Smart Plug-in for Oracle installed.
oanode2.somedomain.com	The fully-qualified domain name of a server that has the HP Operations Agent and the HP Operations Smart Plug-in for Oracle installed.
Add more servers that have the HP Operations Agent and the HP Operations Smart Plug-in for Oracle installed.	The fully-qualified domain name of a server that has the HP Operations Agent and the HP Operations Smart Plug-in for Oracle installed.

To edit the node list file, do the following from the OpsA Server Appliance:

a. Copy the `sample_OA_PA_node.properties` file from the `$OPSA_HOME/conf/collection/sample` directory to some location, such as `/tmp/mynodelist.properties`.

b. Edit the `/tmp/mynodelist.properties` file, adding the appropriate information; then save your work.

2. Run the following command from the OpsA Server Appliance to create the collector configuration:
`$OPSA_HOME/bin/opsa-collection-config.sh -create -nodelist /tmp/mynodelist.properties -collectorhost <fully-qualified domain name of the collector host> -source oa -domain oraperf -group graph -username opsatenantadmin`

Note: The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

Note: The `opsa-collection-config.sh` script uses the values of `source`, `domain`, and `group` to select the right predefined collection template for and create the desired collection configuration.

3. Run the following command from the OpsA Server Appliance to validate the collection configuration you just created:

`$OPSA_HOME/bin/opsa-collection-config.sh -list -allversions -`

`collectorhosts -username opsatenantadmin`
To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its `version`, `domain`, and `group`.

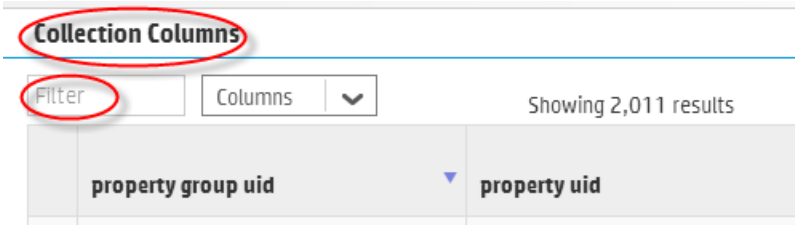
4. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance :

   `$OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost <fully-qualified domain name of the collector host> -username opsatenantadmin`
   The `-publish` option uploads the collection configuration you created to the OpsA Collector Appliance. To verify that the collector configuration published successfully, look for a message stating that the publish was successful, that a table was successfully created, and that the collection was restarted.

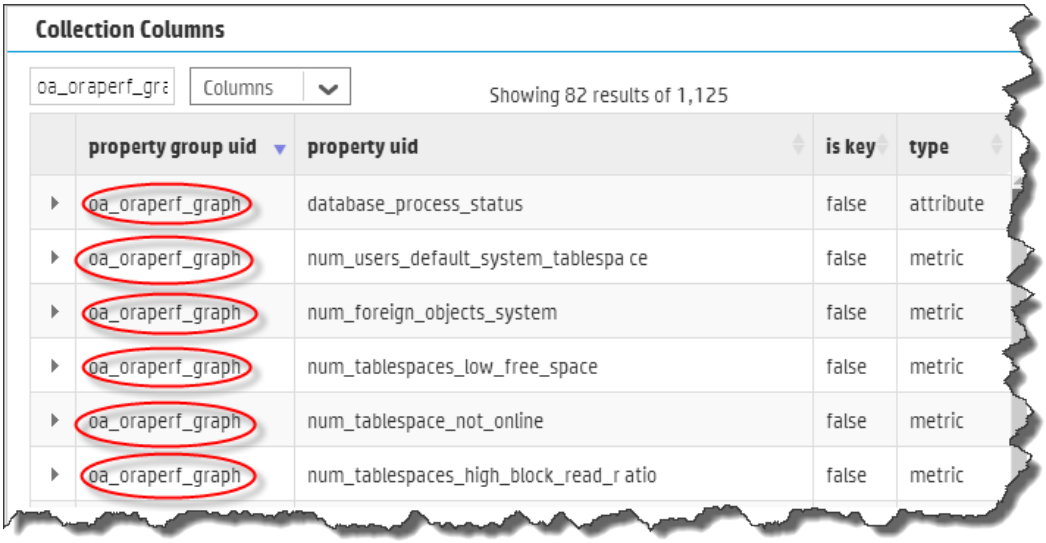
5. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

   Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `oa`, a domain of `oraperf`, and a group of `graph` when creating the collection. The resulting property group uid would be `oa_oraperf_graph`.

   a. Type the property group uid (`oa_oraperf_graph`) for this collection in the **Collection ColumnsFilter:**

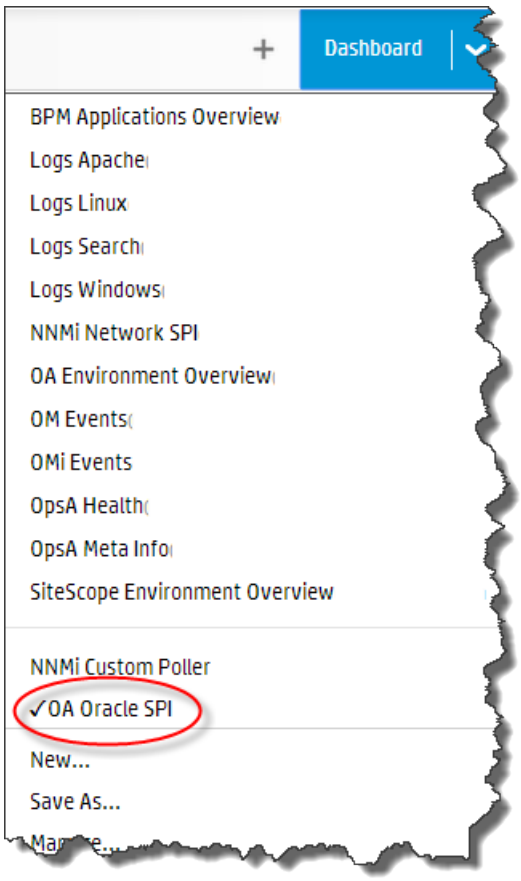


b. After typing property group uid (`oa_oraperf_graph`) for this collection in the **Collection ColumnsFilter**, you should see some information similar to the following for this collection:



6. Create dashboards and query panes for the data you are now collecting. Follow the instructions shown in the *Dashboards and Query Panes* section of the *Operations Analytics Help* for information about creating dashboards and query panes.

7. For this example, assume you created the **OA Oracle SPI** dashboard. From the Operations Analytics console, open the **OA Oracle SPI** dashboard to view some of the collected information for this collection:



8. Create AQL functions for the data you are now collecting. See the instructions shown in the *Define Analytic Query Language (AQL) Functions* section of the *Operations Analytics Help* for information about creating AQL functions.For example, using the property group information shown in the **OpsA Meta Info** dashboard, you might create AQL functions similar to the example shown below.

9. If you want to add tags to an HP Operations Smart Plug-in for Oracle Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

To remove nodes from an HP Operations Smart Plug-in for Oracle Collection follow these steps:

1. Copy the node list file to a temporary location. For example, you might run the following command:
`cp /opt/HP/opsa/conf/collection/config.files/<collectorhost>/<tenant>1.0/oa/1.0/oraperf/graph/nodelist /tmp`

2. Edit the node list file. For example, you might edit the `tmp/nodelist` file, then remove the `HP Operations Smart Plug-in for Oracle Collection` nodes.

3. For this example, you might run the following command:
`opsa-collection-config.sh -nodelist /tmp/nodelist -collectorhost <fully-qualified domain name of the collector host> -source oa -domain oraperf -group graph -username opsatenantadmin.`
Enter `yes` when prompted with, "`Do you want to overwrite the existing node list (instead of appending to it) (Y/N) ?`"

4. For this example, you might run the following command to publish these changes:
`opsa-collection-config.sh -publish -collectorhost <fully-qualified domain name of collector host> -username opsatenantadmin`

Configuring an HP Operations Agent Collection (Detailed Method)

It is recommended that you use a more automated method of configuring this collection. See "Configuring an HP Operations Agent Collection" on page 30 for the recommended method. For a more detailed approach, you can complete the following steps for the HP Operations Agent Collection.

There are two methods of configuring Operations Analytics (OpsA) for HP Operations Agent, the **Automated Configuration Method** and the **Manual Configuration Method**. You must select one of these methods to configure OpsA for HP Operations Agent. Do not attempt to use both the Automated Configuration Method and the Manual Configuration Method.

The HP Operations Agent Collection collects global system information on the host that is running the HP Operations Agent. After you complete the steps in this section, the HP Operations Agent Collection collects raw metrics every 15 minutes, with 5 minute data granularity.

Note: An OpsA Appliance can reliably collect data from approximately 5,000 nodes being monitored by the HP Operations Agent.

Automated Configuration Method: This approach is more automated than the Manual Configuration Method, making configuration easier. Do the following to use the Automated Configuration Method:

1. The Automated Configuration Method provides the sample collection template, `$OPSA_HOME/conf/collection/sample/sample_auto-config-node.properties`, located on the OpsA Server Appliance. Copy the `sample-auto-config-node.properties` file to a separate location, then edit the `sample-auto-config-node.properties` file and add, among other information, the following Operations Manager information.

   Note: The sample file mentioned in this step does not contain a password property. When using the `opsa-collection-config.sh` script (in a later step) to encrypt the password, this

script prompts you for the password, encrypts it, and inserts it into the sample file.

- For an OMW host (HP Operations Manager on a Windows server), the Automated Configuration Method includes the OMW host's fully-qualified domain name and its database details.
- For an OMU host (HP Operations Manager on a UNIX or Linux server), the Automated Configuration Method includes the OMU host's fully-qualified domain name and its database details.

2. Save your work.
3. Run the following command from the OpsA Server Appliance to encrypt the passwords in the `mynodelists.properties` file:
   `$OPSA_HOME/bin/opsa-collection-config.sh -encrypt <path>/sample-auto-config-node.properties`

   Note: You will be prompted to enter the password after running this command.

4. Run the following command from the OpsA Server Appliance to configure the collector hosts and publish the collection information:

   ```
   $OPSA_HOME/bin/opsa-collector-auto-conf.sh -nodelist <path>/<sample-auto-config-node.properties> -collectevents -configuredomain [ORACLE|SYSTEM] -username opsatenantadmin
   ```

   Note: The `opsa-collector-auto-conf.sh` script sets up HP Operations Agent, HP Operations Smart Plug-in for Oracle, and OM event collection using only one command.

   Note: The command output shows the list of registered collectors and distributes the list of nodes providing system performance metrics, Oracle performance metrics, or both among the registered collectors.

   Note: The `opsa-collector-auto-conf.sh` script prompts you for the Tenant Admin password for the username you use in the commands shown in this section. See the *opsa-collector-auto-conf.sh* reference page (or the Linux manpage) for more information.

Manual Configuration Method: Do the following for a more manual approach to configuring OpsA for HP Operations Agent:

1. A properties file contains details about the sources from which you plan to collect information. There are sample properties files located in the `$OPSA_HOME/conf/collection/sample`

directory. OpsA administrators can use these sample files to publish the properties file. The sample properties file for the HP Operations Agent collection is `sample_oa_pa_node.properties`. The properties file for the Operations Agent collection must contain a list of servers that have the HP Operations Agent installed. The properties file for the Operations Agent collection must include the information shown in the following table:

Properties File Fields and Values

Field	Value
panode1.somedomain.com	The fully-qualified domain name of a server that has the HP Operations Agent installed.
panode2.somedomain.com	The fully-qualified domain name of a server that has the HP Operations Agent installed.
Add more servers that have the HP Operations Agent installed	The fully-qualified domain name of a server that has the HP Operations Agent installed.

To edit the properties file, do the following from the Operations Analytics Server Appliance:

a. Copy the `sample_OA_PA_node.properties` file from the `$OPSA_HOME/conf/collection/sample` directory to some location, such as `/tmp/mynodelist.properties`.

b. Edit the `/tmp/mynodelist.properties` file, adding the appropriate information; then save your work.

2. Run the following command from the OpsA Server Appliance to create the collector configuration:
`$OPSA_HOME/bin/opsa-collection-config.sh -create -nodelist /tmp/mynodelist.properties -collectorhost <fully-qualified domain name of the collector host> -source oa -domain sysperf -group global -username opsatenantadmin`

Note: The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

Note: The `opsa-collection-config.sh` script uses the values of `source`, `domain`, and `group` to select the right predefined collection template for and create the desired collection configuration.

3. Run the following command from the OpsA Server Appliance to validate the collection configuration you just created:

```
$OPSA_HOME/bin/opsa-collection-config.sh -list -allversions -collectorhosts -username opsatenantadmin
```

To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its `version`, `domain`, and `group`.

4. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance :

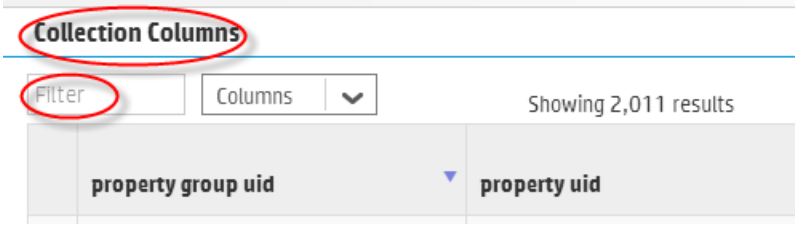
   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost <fully-qualified domain name of the collector host> -username opsatenantadmin
   ```

   The `-publish` option uploads the collection configuration you created to the OpsA Collector Appliance. To verify that the collector configuration published successfully, look for a message stating that the publish was successful, that a table was successfully created, and that the collection was restarted.

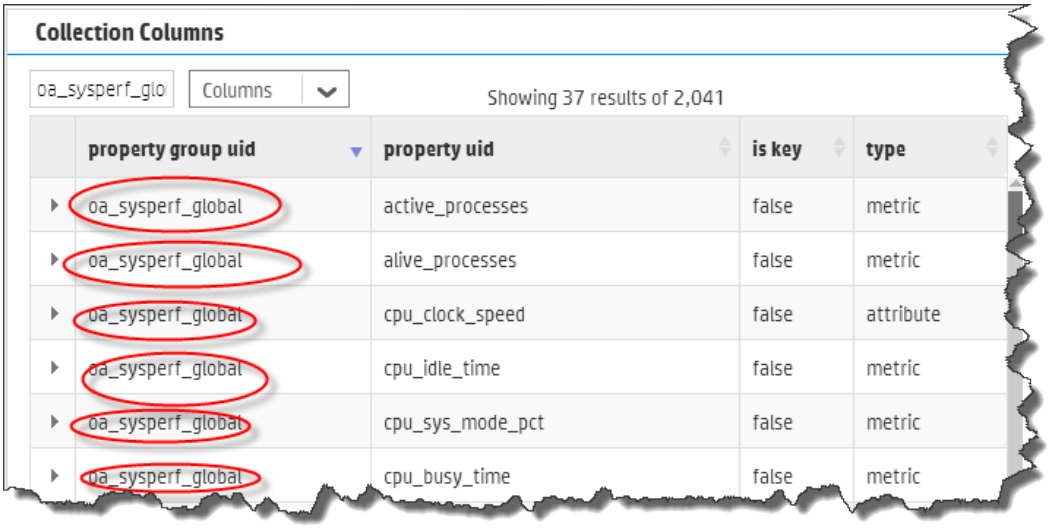
5. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

   Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `oa`, a domain of `sysperf`, and a group of `global` when creating the collection. The resulting property group uid would be `oa_sysperf_global`.

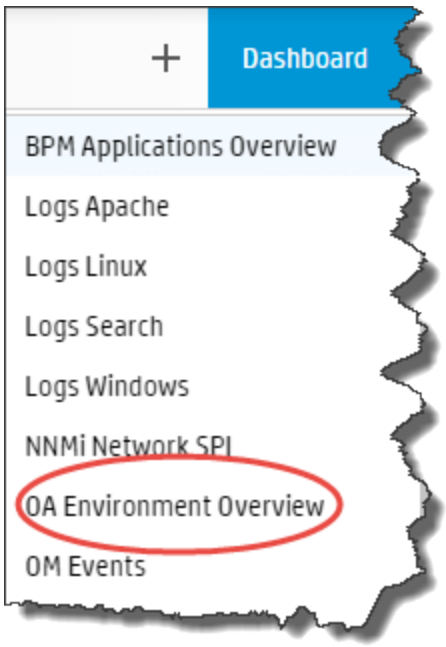
   a. Type the property group uid (`oa_sysperf_global`) for this collection in the **Collection ColumnsFilter:**



b. After typing property group uid (`oa_sysperf_global`) for this collection in the **Collection ColumnsFilter**, you should see information for this collection.



6. From the Operations Analytics console, open the **OA Environment Overview** dashboard to view some of the collected information for this collection:



To remove nodes from an HP Operations Agent Collection follow these steps:

1. Copy the properties file to a temporary location. For example, you might run the following command:
   `cp /opt/HP/opsa/conf/collection/config.files/<collectorhost>/<tenant>/1.0/oa/1.0/sysperf/global/nodelist /tmp`

2. Edit the properties file. For example, you might edit the `tmp/nodelist` file, then remove the `HP Operations Agent Collection` nodes.

3. For this example, you might run the following command:
   `opsa-collection-config.sh -nodelist /tmp/nodelist -collectorhost <fully-qualified domain name of the collector host> -source oa -domain sysperf -group global -username opsatenantadmin.`
   Enter `yes` when prompted with, "`Do you want to overwrite the existing node list (instead of appending to it) (Y/N) ?`"

4. For this example, you might run the following command to publish these changes:
   `opsa-collection-config.sh -publish -collectorhost <fully-qualified domain name of collector host> -username opsatenantadmin`

Configuring an NNM ISPi Performance for Metrics Component Health Collection (Detailed Method)

It is recommended that you use a more automated method of configuring this collection. See "Configuring an NNM ISPi Performance for Metrics Component Health Collection" on page 35 for the recommended method. For a more detailed approach, you can complete the following steps for the NNM iSPI Performance for Metrics Component Health Collection.

After you complete the steps in this section, the NNMi ISPi Performance for Metrics Component Health Collection reads data from the CSV files within 60 seconds of the file being placed in the source directory.

1. For the Collector Appliance to access raw metric information from the NNM iSPI Performance for Metric's component health extension pack, you must export these metrics to CSV files. Run the following command on the NNM iSPI Performance for Metric server to export these metrics to CSV files in the `/csvexports` directory:

   - *Windows (Raw Information)*:
     `<Install_Dir>\NNMPerformanceSPI\bin\configureCsvExport.ovpl -p Component_Health -a "Raw,<Target-Dir>"`

   - *UNIX: (Raw Information)*:
     `/opt/OV/NNMPerformanceSPI/bin/configureCsvExport.ovpl -p Component_Health -a "Raw,<Target-Dir>"`

Note: You must make the exported component health metrics available on the Operations Analytics (OpsA) Collector Appliance in the `/opt/HP/opsa/data/netcomponent` directory.

If you want to use a different directory than `/opt/HP/opsa/data/netcomponent`, do the following:

a. Edit the following collection template:
`/opt/HP/opsa/conf/collection/server/config.templates/nnmispi/1.0/netcomponent/component/nnmispi_netcomponent_component_collection.xml`.

b. Specify a different directory for the `sourcedir` attribute.

Note: The `opsa` user on the OpsA Collector Appliance must have read and write access to the component health metric files in the OpsA Collector Appliance to move them to the processed directory. The default process directory is `/opt/HP/opsa/data/netcomponent_processed`.

For example, to configure read and write access to the component health metric files to the OpsA Collector Appliance when the files are located on a Windows server, do the following:

a. On a Windows server, navigate to **Computer Management** > **System Tools** > **Shares** > **Shared Folders**.

b. Right-click beneath shares and open the new share wizard.

c. Create shares for the directories in which the .csv files are stored.

d. From the OpsA Collector Appliance, add the correct entries to the `/etc/fstab` file. Use the following entries as a model:

```
//10.17.18.19/final /opt/HP/opsa/data/nnm cifs
username=administrator,password=password,uid=opsa,rw 0 0
//10.15.14.13/componentfinal /opt/HP/opsa/data/netcomponent
cifs username=admin,password=passwd,uid=opsa,rw 0 0
//10.15.14.13/interfacefinal /opt/HP/opsa/data/netinterface
cifs username=admin,password=passwd,uid=opsa,rw 0 0
```

e. Use the mount -a command to get the directories mounted.

2. Run the following command from the OpsA Server Appliance to create the collector configuration:
`$OPSA_HOME/bin/opsa-collection-config.sh -create -collectorhost <fully-qualified domain name of the collector host> -source nnmispi -domain netcomponent -group component -username opsatenantadmin`

Note: The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

Note: The `opsa-collection-config.sh` script uses the values of `source`, `domain`, and `group` to select the right predefined collection template for and create the desired collection configuration.

3. Run the following command from the OpsA Server Appliance to validate the collection configuration you just created:

   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -list -allversions -
   collectorhosts -username opsatenantadmin
   ```

   To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its `version`, `domain`, and `group`.

4. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance :

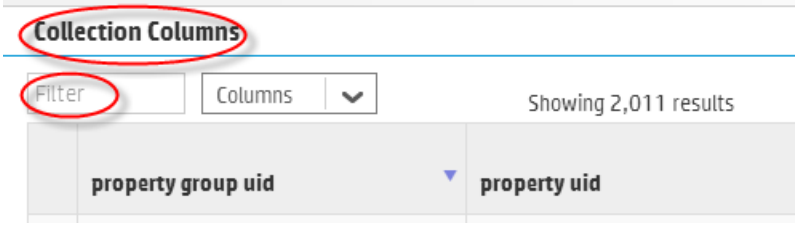
   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost
   <fully-qualified domain name of the collector host> -username
   opsatenantadmin
   ```

   The `-publish` option uploads the collection configuration you created to theOpsA Collector Appliance. To verify that the collector configuration published successfully, look for a message stating that the publish was successful, that a table was successfully created, and that the collection was restarted.

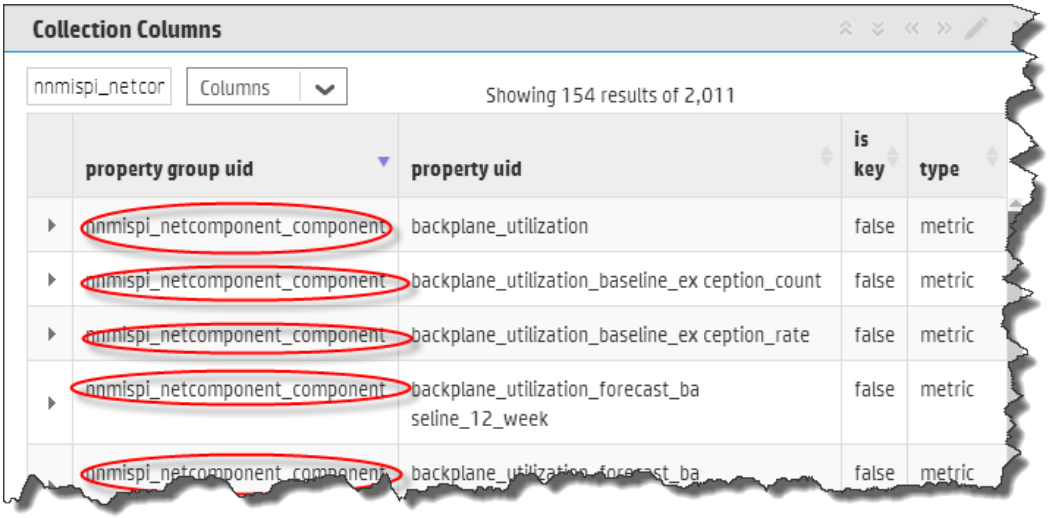
5. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Do the following to look for the **property group uid** for the collection you just created and published.

   Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `nnmispi`, a domain of `netcomponent`, and a group of `component` when creating the collection. The resulting property group uid would be `nnmispi_netcomponent_component`.

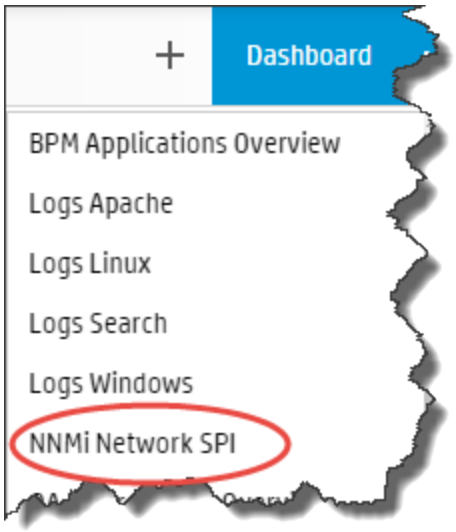
   a. Type the property group uid (`nnmispi_netcomponent_component`) for this collection in the **Collection Columns Filter:**



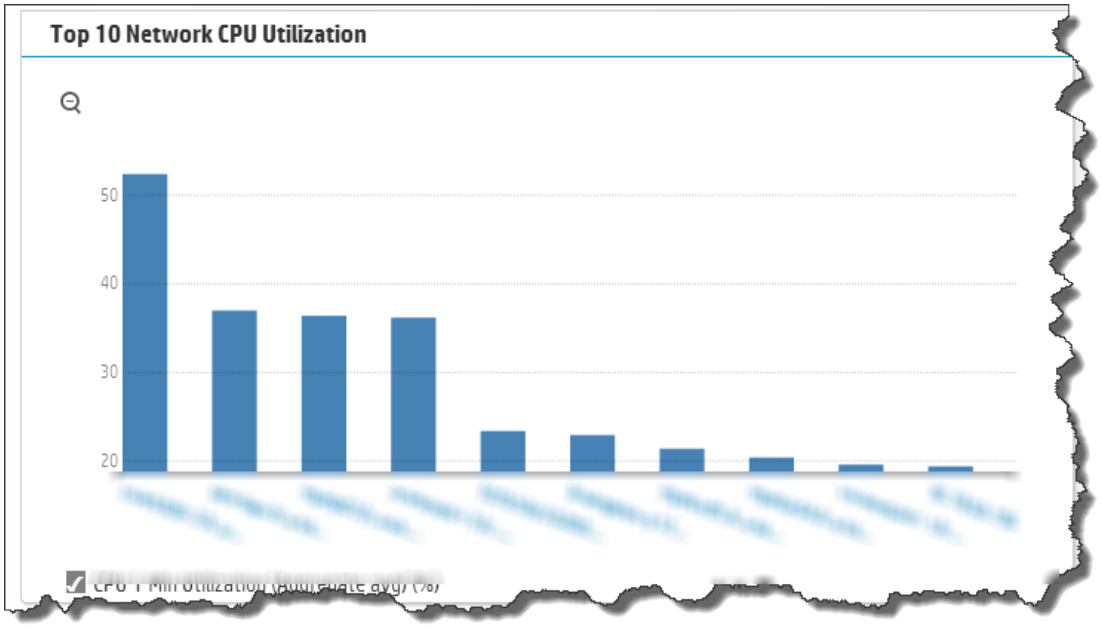
b. After typing property group uid (`nnmispi_netcomponent_component`) for this collection in the **Collection Columns Filter**, you should see information in the resulting table:



6. From the Operations Analytics console, open the **NNMi Network SPI** dashboard to view some of the collected information for this collection:



The following is a small example of NNM ISPi Performance for Metrics Component Health Collection data provided by the **NNMi Network SPI** dashboard.



7. If you want to add tags to an NNM ISPi Performance for Metrics Component Health Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Configuring an NNM ISPi Performance for Metrics Interface Health Collection (Detailed Method)

It is recommended that you use a more automated method of configuring this collection. See "Configuring an NNM ISPi Performance for Metrics Interface Health Collection" on page 40 for the recommended method. For a more detailed approach, you can complete the following steps for the NNM iSPI Performance for Metrics Interface Health Collection.

After you complete the steps in this section, the NNMi ISPi Performance for Metrics Interface Health Collection reads data from the CSV files within 60 seconds of the file being placed in the source directory.

1. For the Collector Appliance to access live metric information from the NNM iSPI Performance for Metric's interface health extension pack, you must export these metrics to CSV files. Run the following command on the NNM iSPI Performance for Metric server to export these metrics to CSV files in the `/csvexports` directory:

- *Windows (Raw Information)*:
  `<Install_Dir>\NNMPerformanceSPI\bin\configureCsvExport.ovpl -p Interface_Health -a "Raw,<Target_Directory">`
- *UNIX (Raw Information*):
  `/opt/OV/NNMPerformanceSPI/bin/configureCsvExport.ovpl -p Interface_Health -a "Raw,<Target_Directory">`

Note: You must make the exported interface health metrics available on the Operations Analytics (OpsA) Collector Appliance in the `/opt/HP/opsa/data/netinterface` directory.

If you want to use a different directory than `/opt/HP/opsa/data/netinterface`, do the following:

a. Edit the following collection template:
   `/opt/HP/opsa/conf/collection/server/config.templates/nnmispi/1.0/netinterface/interface/nnmispi_netinterface_interface_collection.xml`.

b. Specify a different directory for the `sourcedir` attribute.

Note: The `opsa` user on the OpsA Collector Appliance must have read and write access to the interface health metric files in the OpsA Collector Appliance to move them to the processed directory. The default process directory is `/opt/HP/opsa/data/netinterface_processed`.

For example, to configure read and write access to the interface health metric files to the OpsA Collector Appliance when the files are located on a Windows server, do the following:

a. On a Windows server, navigate to **Computer Management** > **System Tools** > **Shares** > **Shared Folders**.

b. Right-click beneath shares and open the new share wizard.

c. Create shares for the directories in which the .csv files are stored.

d. From the OpsA Collector Appliance, add the correct entries to the `/etc/fstab` file. Use the following entries as a model:

```
//10.17.18.19/final /opt/HP/opsa/data/nnm cifs username=administrator,password=password,uid=opsa,rw 0 0
//10.15.14.13/componentfinal /opt/HP/opsa/data/netcomponent cifs username=admin,password=passwd,uid=opsa,rw 0 0
//10.15.14.13/interfacefinal /opt/HP/opsa/data/netinterface
```

```
cifs username=admin,password=passwd,uid=opsa,rw 0 0
```

e. Use the mount -a command to get the directories mounted.

2. Run the following command from the OpsA Server Appliance to create the collector configuration: `$OPSA_HOME/bin/opsa-collection-config.sh -create -collectorhost <fully-qualified domain name of the collector host> -source nnmispi -domain netinterface -group interface -username opsatenantadmin`

   Note: The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

   Note: The `opsa-collection-config.sh` script uses the values of `source`, `domain`, and `group` to select the right predefined collection template for and create the desired collection configuration.

3. Run the following command from the OpsA Server Appliance to validate the collection configuration you just created:

   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -list -allversions -collectorhosts -username opsatenantadmin
   ```

   To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its version, domain, and group.

4. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance :

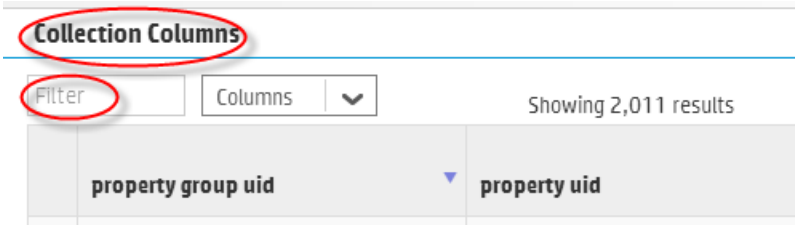
   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost <fully-qualified domain name of the collector host> -username opsatenantadmin
   ```

   The `-publish` option uploads the collection configuration you created to the OpsA Collector Appliance. To verify that the collector configuration published successfully, look for a message stating that the publish was successful, that a table was successfully created, and that the collection was restarted.

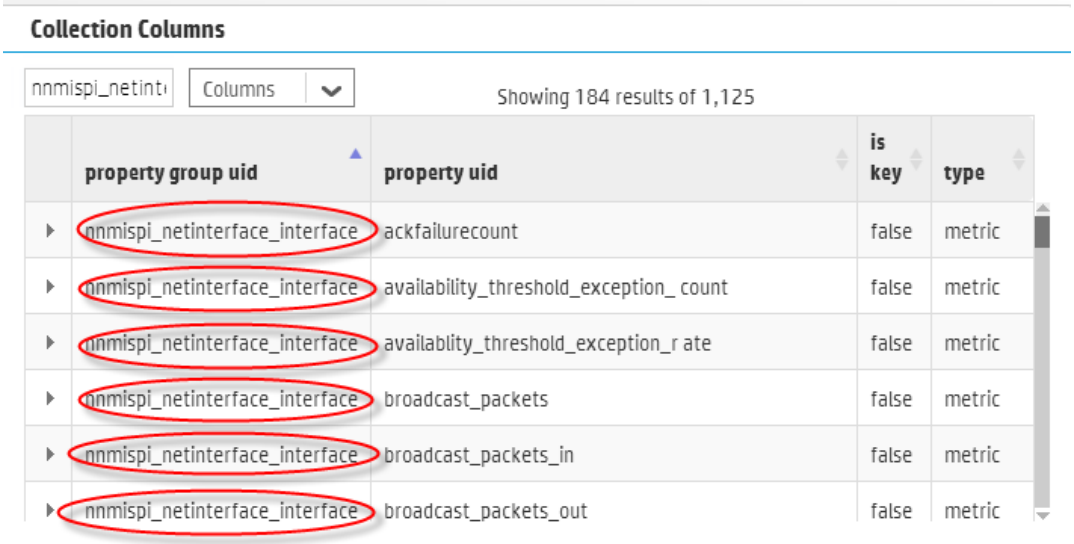
5. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `nnmispi`, a domain of `netinterface`, and a group of `interface` when creating the collection. The resulting property group uid would be `nnmispi_netinterface_interface`.

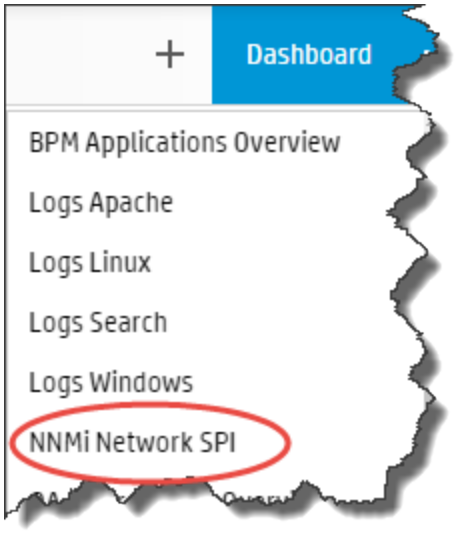
a. Type the property group uid (`nnmispi_netinterface_interface`) for this collection in the **Collection ColumnsFilter:**



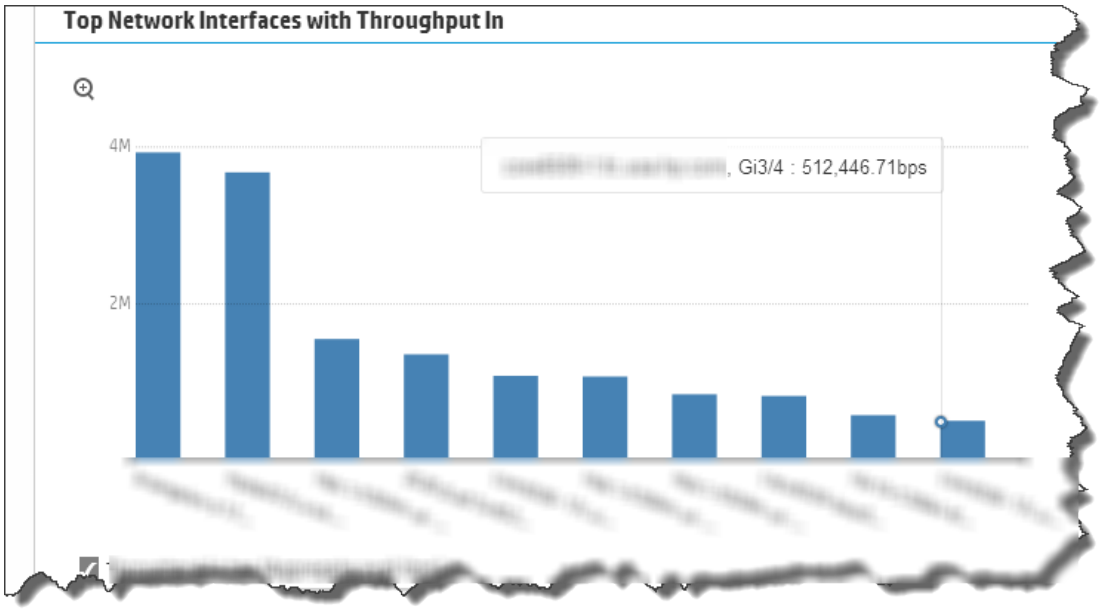
b. After typing property group uid (`nnmispi_netinterface_interface`) for this collection in the **Collection ColumnsFilter**, you should see information in the resulting table:



6. From the Operations Analytics console, open the **NNMi Network SPI** dashboard to view some of the collected information for this collection:



The following is a small example of NNM ISPi Performance for Metrics Interface Health Collection data provided by the **NNMi Network SPI** dashboard.



7. If you want to add tags to an NNM ISPi Performance for Metrics Interface Health Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Configuring the HP OMi Events Collection (Detailed Method)

It is recommended that you use a more automated method of configuring this collection. See "Configuring an HP Operations Manager (OMi) Events Collection" on page 44 for the recommended method. For a more detailed approach, you can complete the following steps for the HP OMi Events Collection.

After you complete the steps in this section, the HP OMi Events Collection collects events every 15 minutes, and collects all OMi events that occurred since the last poll.

Note: To support this collection using Oracle RAC, do the following:

1. Copy the `tnsnames.ora` file from the Oracle server to the following locations:

   OpsA Server: `/opt/HP/opsa/conf/collection/tnsnames.ora`

   OpsA Collector: `/opt/HP/BSM/PMDB/config/tnsnames.ora`

2. Rename the `tnsnames.ora` files:

   OpsA Server: `/opt/HP/opsa/conf/collection/bsm-tnsnames.ora`

   OpsA Collector: `/opt/HP/BSM/PMDB/config/bsm-tnsnames.ora`

Note: The OMi collection is also able to accept events from HP Service Health Analyzer (SHA), a component of Business Service Management (BSM). If you have installed SHA on a BSM server and have configured the OMi collection, the OMi collection automatically accepts SHA events. You can then use the collected SHA event data to anticipate and predict IT problems. The following is a short description of SHA:

HP Service Health Analyzer (SHA): HP Service Health Analyzer analyzes abnormal service behavior and alerts IT managers of service degradation before an issue affects their business.

Note: An Operations Analytics (OpsA) Collector Appliance can only collect data for a single HPOM or OMi event source. If you configure more than one HPOM or OMi event source for an OpsA Collector Appliance, it collects the events from only one of the event sources at every collection interval. It cannot be determined from which event source data collection occurs for a given collection interval. To remedy this, configure a separate OpsA Collector appliance for each HPOM or OMi event source.

Setting the Correct Time Zone

If the (OpsA) Collector Appliance is in a different time zone (not in GMT+0), you must specify that time zone in the collection template file before configuring the OMi Events collection.

Do the following if you want to specify a time zone other than GMT

1. Edit the following collection template file:
   `/opt/HP/opsa/conf/collection/server/config.templates/omi/1.0/events/omievents/omi_collection.xml`

   Note: The default template version is 1.0 as shown in this step. Specify the version as appropriate.

2. Specify the OpsA Collector Appliance time zone offset from GMT by by editing the timezone attribute of the following column element names: `timestamp`, `TimeReceivedTimeStamp`, and `TimeCreatedTimeStamp`.

   For example, if the OpsA Collector Appliance is in GMT+3, you must specify a timezone that is 3 hours ahead of GMT. That means the timezone attribute's value needs to be GMT+3 as shown in the following example:

   ```
   <column name="timestamp" position="2" datatype="datetime" length="0"
   datetype="date" format="MM/dd/yyyy HH:mm:ss" timezone="GMT+3 "key="no"
   mapsto="TimeOfStateChangeTimeStamp" value="" label="Timestamp"
   columnname="timestamp" unit="" tags="" type="attribute" />
   ```

3. Save your changes.

After you complete the above steps to specify your time zone in the collection template file, you can use the instructions in the remainder of this section to configure the HP OMi collection.

Configuring the HP OMi Events Collection (Detailed Method)

It is recommended that you use a more automated method of configuring this collection. See "Configuring an HP Operations Manager (OMi) Events Collection" on page 44 for the recommended method. For a more detailed approach, you can complete the following steps for the OMi collection events:

1. A node list file contains details about the sources from which you plan to collect information. There are sample nodelist files located in the `$OPSA_HOME/conf/collection/sample` directory. Operations Analytics (OpsA) administrators can use these sample files to publish the node list file. Choose the sample node list file based on the database used by your HP OMi application: `sample_OMi_MSSQL_node.properties` or `sample_OMi_ORACLE_node.properties`. The node list file for the HP OMi Event collection must include the information shown in the

following table:

Node List File Fields and Values

Field	Value
omidbserver.hostdnsname	The fully-qualified domain name of the OMi server.
omidbserver.port	`1433`: The port used to connect to the OMi server.
omidbserver.username	`USERNAME`: The user name to use for connecting to the OMi server.
omidbserver.instanceName	`INSTANCE_NAME`
omidbserver.datasource_type	`OMI`
omidbserver.database_type	`ORACLE` or `MSSQL` (check with QA--are both types used by OMi?)
omidbserver.database_name	`DATABASE_NAME`

To edit the node list file, do the following from the OpsA Server Appliance:

a. Copy the `sample_OMi_MSSQL_node.properties` or `sample_OMi_ORACLE_node.properties` file from the `$OPSA_HOME/conf/collection/sample` directory to some location, such as `/tmp/mynodelist.properties`.

> **Note:** Choose the sample node list file based on the database used by your HP OMi application.

> **Note:** The sample file mentioned in this step does not contain a password property. When using the `opsa-collection-config.sh` script (in a later step) to encrypt the password, this script prompts you for the password, encrypts it, and inserts it into the sample file.

b. Edit the `/tmp/mynodelist.properties` file, adding the appropriate information; then save your work.

2. Run the following command from the OpsA Server Appliance to encrypt the password:
`$OPSA_HOME/bin/opsa-collection-config.sh -encrypt /tmp/mynodelists.properties`

> **Note:** You will be prompted to enter the password after running this command.

3. Run the following command from the OpsA Server Appliance to create the collector configuration:
   `$OPSA_HOME/bin/opsa-collection-config.sh -create -nodelist /tmp/mynodelist.properties -collectorhost <fully-qualified-domain-name of collector host> -source omi -domain events -group omievents -username opsatenantadmin`

   Note: The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

   Note: The `opsa-collection-config.sh` script uses the values of `source`, `domain`, and `group` to select the right predefined collection template for and create the desired collection configuration.

4. Run the following command from the OpsA Server Appliance to verify and validate the collection configuration you just created:

   `$OPSA_HOME/bin/opsa-collection-config.sh -list -allversions -collectorhosts -username opsatenantadmin`

   To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its `version`, `domain`, and `group`.

5. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance :

   `$OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost <fully-qualified domain name of the collector host> -username opsatenantadmin`

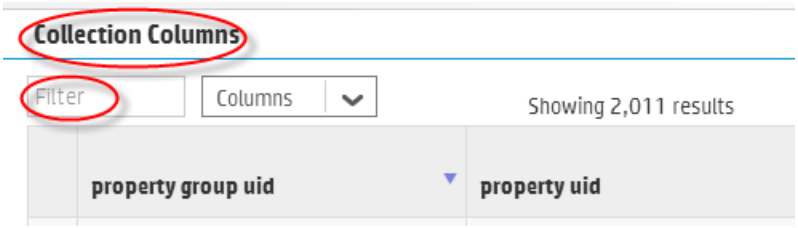
   The `-publish` option uploads the collection configuration you created to the OpsA Collector Appliance. To verify that the collector configuration published successfully, look for a message stating that the publish was successful, that a table was successfully created, and that the collection was restarted.

6. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

   Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `omi`, a domain of `events`, and a group of `omievents` when creating the collection. The resulting

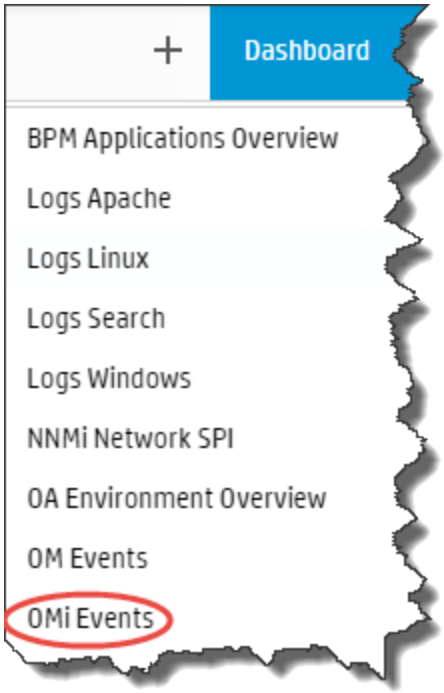
property group uid would be `omi_events_omievents`.

a. Type the property group uid (`omi_events_omevents`) for this collection in the **Collection Columns Filter:**



b. After typing property group uid (`omi_events_omevents`) for this collection in the **Collection ColumnsFilter**, you should see information for this collection.

7. From the Operations Analytics console, open the **OMi Events** dashboard to view some of the collected information for this collection:



Configuring an HP Operations Manager (HPOM) Events Collection (Detailed Method)

It is recommended that you use a more automated method of configuring this collection. See "Configuring an HP Operations Manager (HPOM) Events Collection" on page 49 for the recommended method. For a more detailed approach, you can complete the following steps for the HPOM Events Collection.

For special circumstances related to how Microsoft SQL Server is set up in the HPOM environment, see "Configuring HP Operations Manager (HPOM) (Creating a Database User Account on an HPOM Database Server)" on page 78

Setting the Correct Time Zone

An Operations Analytics Collector Appliance uses the GMT time zone for the HPOM event source. If the HPOM server is in a different time zone, you must specify that time zone in the collection template file before configuring the HPOM collection.

Do the following if you want to specify a time zone other than GMT:

1. Edit the following collection template file:
   `/opt/HP/opsa/conf/collection/server/config.templates/om/1.0/events/om events`

   Note: The default template version is 1.0 as shown in this step. Specify the version as appropriate.

2. Specify the time zone offset from GMT by editing the `timezone` attribute of the following column element names: `timestamp`, `TimeReceivedTimeStamp`, and `TimeCreatedTimeStamp`.

   Specify the total offset between the OpsA Collector Appliance and the OM server (the server from which the OM data comes):

   - If the OpsA Collector Appliance is in GMT+0 (UTC) and the OM server is in GMT+2, the attribute's value needs to be GMT+2.
   - If the OpsA Collector Appliance is in GMT-7 and the OM server is in GMT+0 (UTC), the attribute's value needs to be GMT-7.
   - If the OpsA Collector Appliancer is in GMT-7 and the OM server is in GMT+2, the attribute's value needs to be GMT+9 (the total offset between the servers).

     For this example, set the attribute as follows:
     `<column name="timestamp" position="2" datatype="datetime" length="0"`

```
datetype="date" format="MM/dd/yyyy HH:mm:ss" timezone="GMT+9" key="no"
mapsto="TimeOfStateChangeTimeStamp" value="" label="Timestamp"
columnname="timestamp" unit="" tags="" type="attribute" />
```

3. Save your changes.

After you complete the above steps to specify your time zone in the collection template file, you can use the instructions in the remainder of this section to configure the HPOM collection.

Configuration Steps

After you complete the steps in this section, the HP Operations Manager Events Collection collects events every 15 minutes, and collects all OM events that occurred since the last poll.

Note: An Operations Analytics (OpsA) Collector Appliance can only collect data for a single HPOM or OMi event source. If you configure more than one HPOM or OMi event source for an OpsA Collector Appliance, it collects the events from only one of the event sources at every collection interval. It cannot be determined from which event source data collection occurs for a given collection interval. To remedy this, configure a separate OpsA Collector appliance for each HPOM or OMi event source.

There are two methods of configuring OpsA for the HPOM events collection, the **Automated Configuration Method** and the **Manual Configuration Method**. You must select one of these methods to configure OpsA for the HPOM events collection. Do not attempt to use both the Automated Configuration Method and the Manual Configuration Method.

Automated Configuration Method: This approach is more automated than the Manual Configuration Method, making configuration easier. Do the following to use the Automated Configuration Method:

1. The Automated Configuration Method provides the sample collection template, `$OPSA_HOME/conf/collection/sample/sample_auto_config_node.properties`, located on the Operations Analytics server appliance. Copy the `sample_auto_config_node.properties` file to a separate location, then edit the `sample_om_node.properties` file and add, among other information, the following Operations Manager information:

   Note: The sample file mentioned in this step does not contain a password property. When using the `opsa-collection-config.sh` script (in a later step) to encrypt the password, this script prompts you for the password, encrypts it, and inserts it into the sample file.

   - For an OMW host (HP Operations Manager on a Windows server), the Automated Configuration Method includes the OMW host's fully-qualified domain name and its database details.
   - For an OMU host (HP Operations Manager on a UNIX or Linux server), the Automated Configuration Method includes the OMU host's fully-qualified domain name and its database details.

2. Save your work.

3. Encrypt the passwords in the node properties file by running the following command from the OpsA Server Appliance:
   `$OPSA_HOME/bin/opsa-collection-config.sh –encrypt <path>/sample_auto_config_node.properties`

   Note: You will be prompted to enter the password after running this command.

4. Run the following command from the OpsA Server Appliance to configure the collector hosts and publish the collection information:

   `$OPSA_HOME/bin/opsa-collector-auto-conf.sh -nodelist <path>/<sample_om_node.properties> -collectevents -configuredomain [Oracle|System] -username opsatenantadmin`

   Note: The `opsa-collector-auto-conf.sh` script sets up HP Operations Agent, HP Operations Smart Plug-in for Oracle, and OM event collection using only one command.

   Note: The command output shows the list of registered collectors and distributes the list of nodes providing system performance metrics, Oracle performance metrics, or both among the registered collectors.

   Note: The `opsa-collector-auto-conf.sh` script prompts you for the Tenant Admin password for the username you use in the commands shown in this section. See the *opsa-collector-auto-conf.sh* reference page (or the Linux manpage) for more information.

Manual Configuration Method: Do the following for a more manual approach to configuring OpsA for the HPOM events collection:

1. A node list file contains details about the sources from which you plan to collect information. There are sample nodelist files located in the `$OPSA_HOME/conf/collection/sample` directory. OpsA administrators can use these sample files to publish the node list file. There are two sample node list files for the HPOM Events collection: `sample_OMW_node.properties` (for HP Operations Manager for Windows) and `sample_OMU_node.properties` (for HP Operations Manager for UNIX and Linux). For HPOM events, the node list files contain, among other information, a list of servers that have HPOM installed.

   The node list file for HP Operations Manager for Windows must include the information shown in the following table:

Node List File Fields and Values

Field	Value
omwdbserver.hostdnsname	The fully-qualified domain name of the HPOM server.
omwdbserver.port	`1433`: The port used to connect to the HPOM server.
omwdbserver.username	The user name to use for connecting to the HPOM server. See "Configuring HP Operations Manager (HPOM) (Creating a Database User Account on an HPOM Database Server)" on page 78 for instructions about creating this user.
omwdbserver.instanceName	`OVOPS`
omwdbserver.datasource_type	`OM`
omwdbserver.database_type	`MSSQL`
omwdbserver.database_name	`openview`

The node list file for HP Operations Manager for UNIX and Linux must include the information shown in the following table:

Field	Value
omudbserver.hostdnsname	The fully-qualified domain name of the HPOM server.
omudbserver.port	`1521`: The port used to connect to the HPOM server.
omudbserver.username	`opc_op`: The user name to use for connecting to the HPOM server.
omudbserver.database_name	`null`
omudbserver.instance_name	`openview`
omudbserver.datasource_type	`OM`
omudbserver.database_type	`ORACLE`

To edit the node list file, do the following from the OpsA Server Appliance:

a. Copy the `sample_OMW_node.properties` or `sample_OMU_node.properties` file from the `$OPSA_HOME/conf/collection/sample` directory to some location, such as

`/tmp/mynodelist.properties`.

Note: The sample files mentioned in this step do not contain a password property. When using the `opsa-collection-config.sh` script (in a later step) to encrypt the password, this script prompts you for the password, encrypts it, and inserts it into the sample file.

b. Edit the `/tmp/mynodelist.properties` file, adding the appropriate information; then save your work.

2. Run the following command from the OpsA Server Appliance to encrypt the password:
`$OPSA_HOME/bin/opsa-collection-config.sh -encrypt /tmp/mynodelists.properties`

Note: You will be prompted to enter the password after running this command.

3. Run the following command from the OpsA Server Appliance to create the collector configuration:
`$OPSA_HOME/bin/opsa-collection-config.sh -create -nodelist /tmp/mynodelist.properties -collectorhost <fully-qualified-domain-name of collector host> -source om -domain events -group omevents -username opsatenantadmin`

Note: The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

Note: The `opsa-collection-config.sh` script uses the values of `source`, `domain`, and `group` to select the right predefined collection template for and create the desired collection configuration.

4. Run the following command from the OpsA Server Appliance to validate the collection configuration you just created:

`$OPSA_HOME/bin/opsa-collection-config.sh -list -allversions -collectorhosts -username opsatenantadmin`

To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its `version`, `domain`, and `group`.

5. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance :

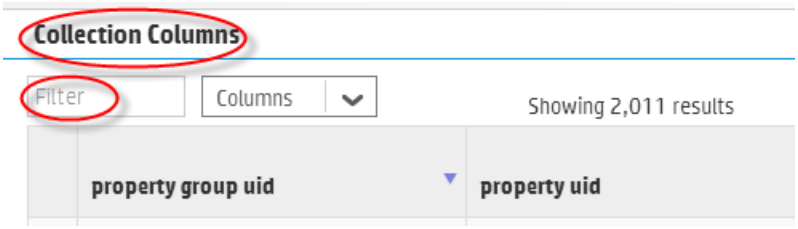
```
$OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost <fully-
qualified domain name of the collector host> –username opsatenantadmin
```

The `-publish` option uploads the collection configuration you created to the OpsA Collector Appliance. To verify that the collector configuration published successfully, look for a message stating that the publish was successful, that a table was successfully created, and that the collection was restarted.

6. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

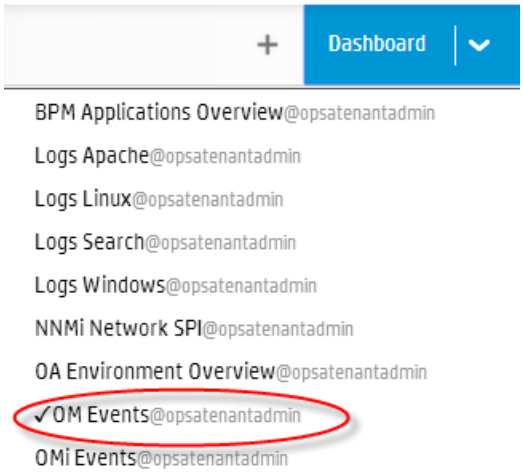
   Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `om`, a domain of `events`, and a group of `omevents` when creating the collection. The resulting property group uid would be `om_events_omevents`.

   a. Type the property group uid (`om_events_omevents`) for this collection in the **Collection ColumnsFilter:**



   b. After typing property group uid (`om_events_omevents`) for this collection in the **Collection ColumnsFilter**, you should see information for this collection.

7. From the Operations Analytics console, open the **OM Events** dashboard to view some of the collected information for this collection:



Configuring an HP BSM RTSM Configuration Item (CI) Collection (Detailed Method)

It is recommended that you use a more automated method of configuring this collection. See "Configuring an HP BSM RTSM Configuration Item (CI) Collection" on page 53 for the recommended method. For a more detailed approach, you can complete the following steps for the HP BSM RTSM Collection.

After you complete the steps in this section, the HP BSM RTSM CI Collection collects data every 6 hours.

Setting the Correct BSM User Name Permissions

When configuring either a BSM RTSM CI collection or a BPM Collection in OpsA you must enter a BSM user name. This BSM user name is used for connecting to the RTSM DPS server, and must be configured for the correct roles.

Note: The user you plan to use for the BPM Collection must be part of the Integration Users Group and include the following OpenAPI required roles:

```
CmdbOpenApiQuery
CmdbOpenApiClassModel
CmdbOpenApiUpdate
CmdbOpenApiImpact
```

These values could be named as shown below in newer versions:

```
RTSMOpenApiQuery
RTSMOpenApiClassModel
RTSMOpenApiUpdate
```

```
RTSMOpenApiImpact
```

Before competing the remaining configuration steps in this section, do the following to test if the user has the required permissions:

1. Try to log on to BSM as your users using the following URL :
   `http://<BSM>:21212/axis2/services/UcmdbService`

   Note: The log on is successful it you see a web page that shows the following message: `Please enable REST support in WEB-INF/conf/axis2.xml and WEB-INF/web.xml.`

   If your credentials are not accepted you are prompted to enter your user name and password. If this happens, the user does not have the required permissions.

2. If the previous step fails, your user is missing some required permissions. Do not continue until you do the following:

   a. Open the RTSM JMX console using the following URL:
   `http://<BSM>:21212/jmx-console/`

   b. Under the **UCMDB** heading, navigate to **UCMDB:service=Security Services**.

   c. Invoke `setRolesForUser JMX` and give the user either the Admin role or all of the OpenAPI roles:
   Admin role:
   `Admin`

   OpenAPI related roles:
   `CmdbOpenApiQuery, CmdbOpenApiClassModel, CmdbOpenApiUpdate, CmdbOpenApiImpact`

   Note: To prevent making mistakes when entering the role names, retrieve the available roles by invoking `getAclController JMX` then copy and paste the role names.

3. Repeat step 1 to verify that you correctly made the permissions changes.

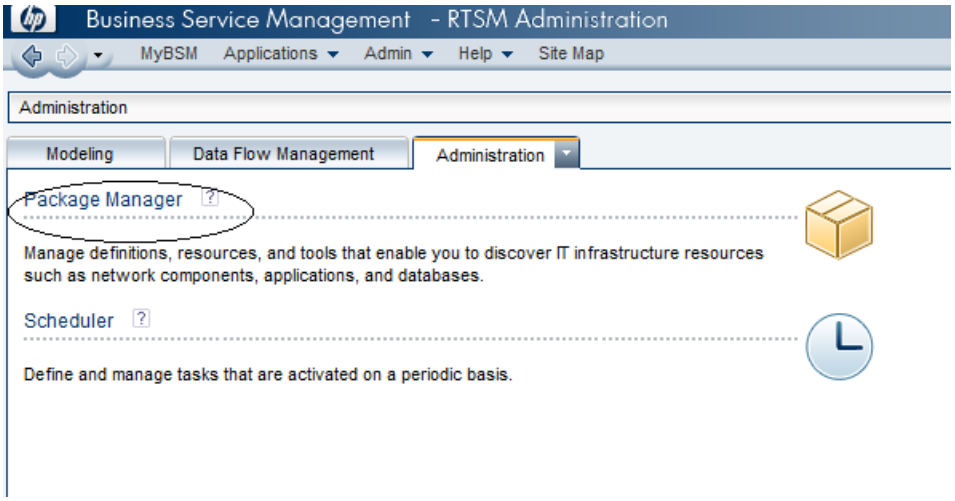
Now the BSM user you tested should have all of the required permissions.

Configuration Steps

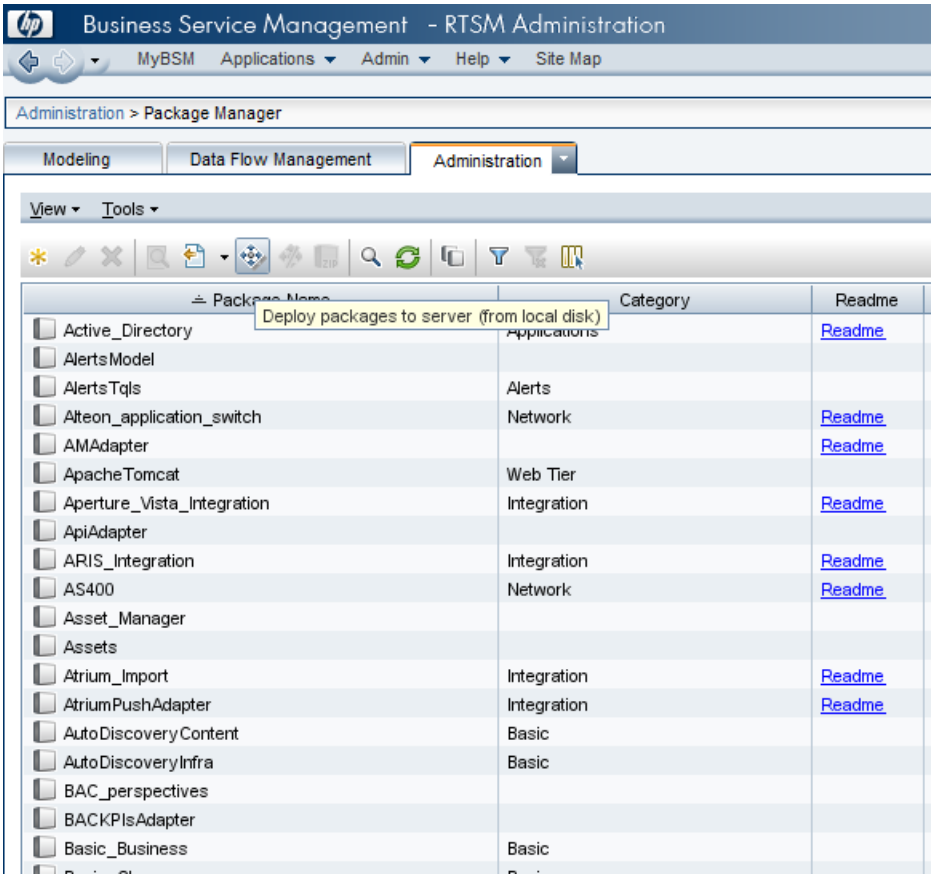
1. Before configuring any of the HP BSM RTSM collections, you must deploy Operations Analytics (OpsA) views on the BSM Server. Do the following:

a. Copy the `$OPSA_HOME/conf/collection/rtsm_views/OPSA_Views.zip` file to the local server from where the BSM UI is launched.

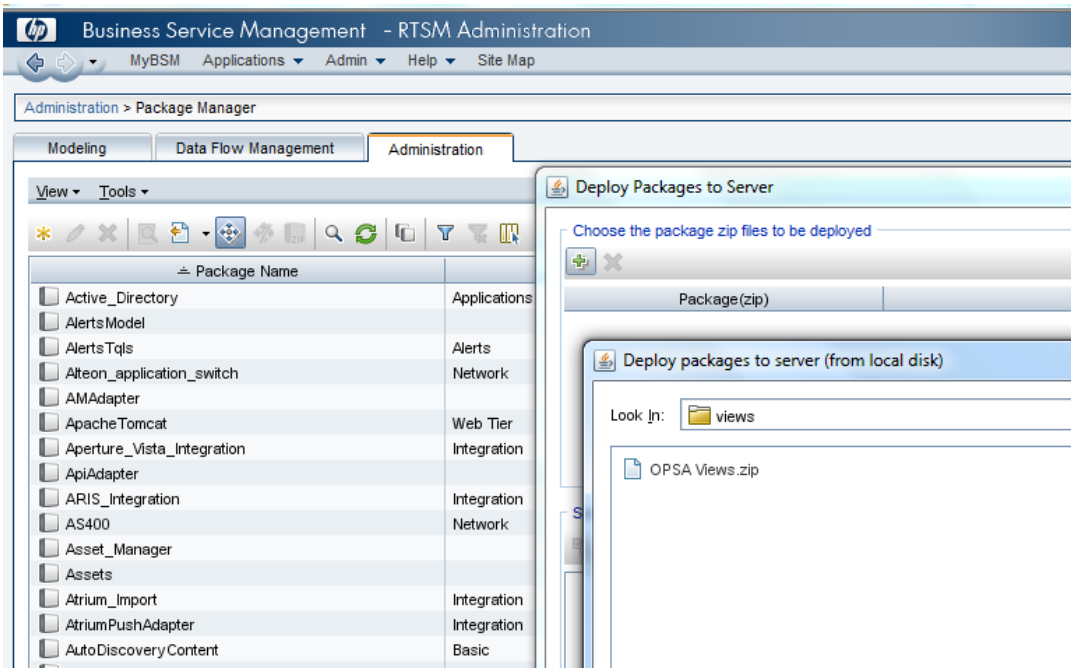
b. Access the **Package Manager** module through the BSM UI:



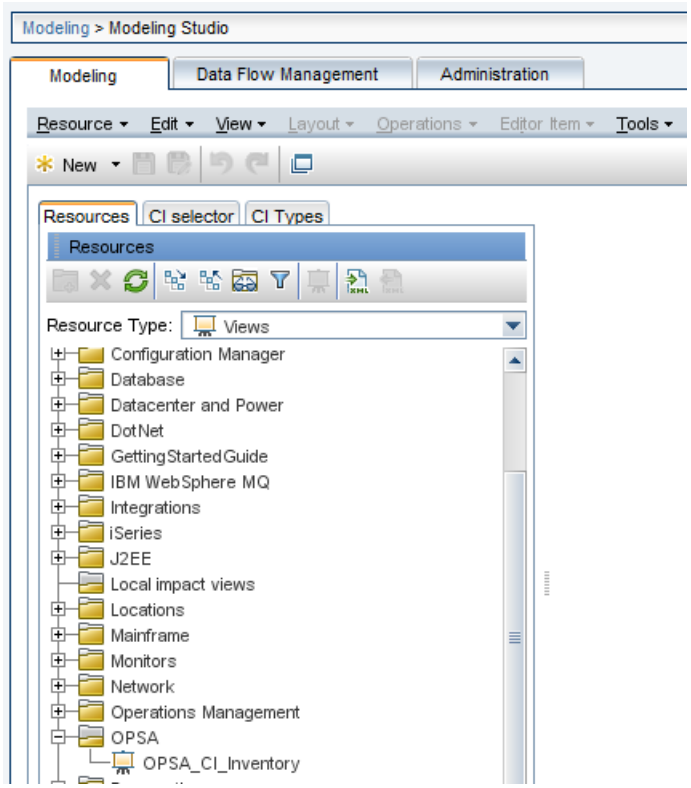
c. Select the **Deploy packages to server (from local disk)** option.



d. Select the **OPSA_Views.zip** file from the local disk as shown in the following screen shot:



e. Once deployed, the views should be visible in the Modeling studio as shown in the following screen shot:



2. A node list file contains details about the sources from which you plan to collect information. There are sample nodelist files located in the `$OPSA_HOME/conf/collection/sample` directory. OpsA administrators can use these sample files to publish the node list file. The sample node list file for the BSM RTSM CI collection is `sample_RTSM_node.properties`.

> **Note:** OpsA administrators can use this sample file to publish the node list file.

The node list file for the BSM RTSM CI collection must include the information shown in the following table:

Node List File Fields and Values

Field	Value
rtsmserver.hostdnsname	The fully-qualified domain name of the RTSM DPS server.
rtsmserver.port	`21212`: The port used to connect to the RTSM DPS server.

Node List File Fields and Values, continued

Field	Value
rtsmserver.username	`admin`: The user name to use for connecting to the RTSM DPS server.
rtsmserver.datasource_type	`rtsm`

To edit the node list file, do the following from the Operations Analytics Server Appliance:

a. Copy the `sample_RTSM_node.properties` file from the `$OPSA_HOME/conf/collection/sample` directory to some location, such as `/tmp/mynodelist.properties`:

> **Note:** The sample file mentioned in this step does not contain a password property. When using the `opsa-collection-config.sh` script (in a later step) to encrypt the password, this script prompts you for the password, encrypts it, and inserts it into the sample file.

b. Edit the `/tmp/mynodelist.properties` file, adding the appropriate information; then save your work.

3. Run the following command from the OpsA Server Appliance to encrypt the password:
`$OPSA_HOME/bin/opsa-collection-config.sh -encrypt /tmp/mynodelist.properties`

> **Note:** You will be prompted to enter the password after running this command.

4. Run the following command from the OpsA Server Appliance to create the collector configuration:
`$OPSA_HOME/bin/opsa-collection-config.sh -create -nodelist /tmp/mynodelist.properties -collectorhost <`*`fully-qualified domain name of the collector host`*`> -source rtsm -domain ci -group inventory -username opsatenantadmin`

> **Note:** The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

> **Note:** The `opsa-collection-config.sh` script uses the values of `source`, `domain`, and `group` to select the right pre-defined collection template to create the desired collection configuration.

5. Run the following command from the OpsA Server Appliance to validate the collection

configuration you just created:

```
$OPSA_HOME/bin/opsa-collection-config.sh
-list -collectorhosts -allversions -username opsatenantadmin
```

To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its `version`, `domain`, and `group`.

6. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance:

   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost
   <fully-qualified domain name of the collector host> -username
   opsatenantadmin
   ```

   The `-publish` option uploads the collection configuration you created to the OpsA collector appliance.

   To verify that the collector configuration published successfully, look for a message stating that the publish was successful, that a table was successfully created, and that the collection was restarted.

7. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

   Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `rtsm`, a domain of `ci`, and a group of `inventory` when creating the collection. The resulting property group uid would be `rtsm_ci_inventory`.

Configuring an HP Business Process Monitor Collection (Detailed Method)

It is recommended that you use a more automated method of configuring this collection. See "Configuring an HP Business Process Monitor Collection" on page 60 for the recommended method. For a more detailed approach, you can complete the following steps for the Business Process Monitor (BPM) Collection.

After you complete the steps in this section, BPM starts sending data to the BPM Collection. The BPM Collection collects metrics related to application transaction response times. The BPM Collection collects data as it arrives from BPM.

Note: You must set the `max heap size` to 3 GB or higher on the Operations Analytics Server Appliance when the following conditions are met:

- OpsA monitors 100 or more BPM applications with 100 or more transactions per application.
- OpsA users make five or more concurrent attempts to open the OpsA default BPM dashboard.

To set the `max heap size`, do the following

1. Edit the `/opt/HP/opsa/jboss/bin/standalone.conf` file.
2. Make the change shown in bold font in the in JAVA_OPTS section:

```
#
# Specify options to pass to the Java VM.
if [ "x$JAVA_OPTS" = "x" ]; then
JAVA_OPTS="-Xms64m -Xmx3072m -XX:MaxPermSize=256m -
Djava.net.preferIPv4Stack=true"
JAVA_OPTS="$JAVA_OPTS -Djboss.modules.system.pkgs=$JBOSS_MODULES_
SYSTEM_PKGS -Djava.awt.headless=true"
else
echo "JAVA_OPTS already set in environment; overriding default
settings with values: $JAVA_OPTS"
fi
```

3. Save your changes.
4. Run the following command to restart the OpsA Server Appliance:

```
$OPSA_HOME/bin/opsa-server restart
```

   See the *opsa-server* reference page (or the Linux manpage) for more information.

Note: Connecting two or more Operations Analytics (OpsA) Collector Appliances to the same BSM server host is not a supported configuration.

Complete the following before proceeding:

Note: For this example, the fully-qualified domain name of the BSM DPS server is `servername.location.domain.com`.

Add an entry for the BSM DPS server to the `/etc/hosts` file in the domain in which the OpsA Collector Appliance resides. For example, you would add a line for the BSM DPS server to the `/etc/hosts` file using the following format:
`10.1.2.3 servername.location.domain.com` *`servername`*.

Note: You must use the alias, *servername*, as the BSM DPS host name in the node list file.

Setting the Correct BSM User Name Permissions

When configuring either a BSM RTSM CI collection or a BPM Collection in OpsA you must enter a BSM user name. This BSM user name is used for connecting to the RTSM DPS server, and must be configured for the correct roles.

Note: The user you plan to use for the BPM Collection must be part of the Integration Users Group and include the following OpenAPI required roles:
`CmdbOpenApiQuery`
`CmdbOpenApiClassModel`
`CmdbOpenApiUpdate`
`CmdbOpenApiImpact`

These values could be named as shown below in newer versions:
`RTSMOpenApiQuery`
`RTSMOpenApiClassModel`
`RTSMOpenApiUpdate`
`RTSMOpenApiImpact`

Before competing the remaining configuration steps in this section, do the following to test if the user has the required permissions:

1. Try to log on to BSM as your users using the following URL :
   `http://<BSM>:21212/axis2/services/UcmdbService`

   Note: The log on is successful it you see a web page that shows the following message: `Please enable REST support in WEB-INF/conf/axis2.xml and WEB-INF/web.xml.`

   If your credentials are not accepted you are prompted to enter your user name and password. If this happens, the user does not have the required permissions.

2. If the previous step fails, your user is missing some required permissions. Do not continue until you do the following:

   a. Open the RTSM JMX console using the following URL:
      `http://<BSM>:21212/jmx-console/`

   b. Under the **UCMDB** heading, navigate to **UCMDB:service=Security Services**.

   c. Invoke `setRolesForUser JMX` and give the user either the Admin role or all of the OpenAPI roles:
      Admin role:
      `Admin`

OpenAPI related roles:
`CmdbOpenApiQuery, CmdbOpenApiClassModel, CmdbOpenApiUpdate, CmdbOpenApiImpact`

Note: To prevent making mistakes when entering the role names, retrieve the available roles by invoking `getAclController JMX` then copy and paste the role names.

3. Repeat step 1 to verify that you correctly made the permissions changes.

Now the BSM user you tested should have all of the required permissions.

Follow the instructions in this section to configure an HP Operations Manager Events Collection for Operations Analytics.

1. A node list file contains details about the sources from which you plan to collect information. There are sample nodelist files located in the `$OPSA_HOME/conf/collection/sample` directory. OpsA administrators can use these sample files to publish the node list file. The sample node list file for the BPM collection is `sample_BPM_node.properties`. The node list file for the BPM collection needs to include a single node from the BSM cluster. This node must be a DPS node. OpsA uses this node to extract BPM data from the BSM cluster. OpsA also uses this node to obtain the BSM location name from BSM's RTSM. The node list file for the BSM BPM collection must include the information shown in the following table:

Node List File Fields and Values

Field	Value
bpmserver.hostdnsname	The fully-qualified domain name of the RTSM DPS server.
bpmserver.port	`21212`: The port used to connect to the RTSM DPS server.
bpmserver.username	`admin`: The user name to use for connecting to the RTSM DPS server.

To edit the node list file, do the following from the OpsA Server Appliance:

a. Copy the `sample_BPM_node.properties` file from the `$OPSA_HOME/conf/collection/sample` directory to some location, such as `/tmp/mynodelist.properties`.

Note: The sample file mentioned in this step does not contain a password property. When using the `opsa-collection-config.sh` script (in a later step) to encrypt the password, this script prompts you for the password, encrypts it, and inserts it into the sample file.

b. Edit the `/tmp/mynodelist.properties` file, adding the appropriate information; then save your work.

2. Run the following command from the Operations Analytics Server Appliance to encrypt the password:
   `$OPSA_HOME/bin/opsa-collection-config.sh -encrypt /tmp/mynodelist.properties`

   Note: You will be prompted to enter the password after running this command.

3. Ping the BSM server from the OpsA Server Appliance to make sure that communication between the OpsA Server Appliance and the BSM Server is functioning.

4. Run the interactive `opsa-collection-setup.sh` script.

5. Enter `8` to begin configuring a BPM Collection.

6. Follow the prompts to enter the requested information.

7. Enter `f` or `finish` after you finish entering all of your information.

Configuration Steps (Manual Method)

To configure a Business Process Monitor (BPM) collection, do the following:

1. A node list file contains details about the sources from which you plan to collect information. There are sample nodelist files located in the `$OPSA_HOME/conf/collection/sample` directory. OpsA administrators can use these sample files to publish the node list file. The sample node list file for the BPM collection is `sample_BPM_node.properties`. The node list file for the BPM collection needs to include a single node from the BSM cluster. This node must be a DPS node. OpsA uses this node to extract BPM data from the BSM cluster. OpsA also uses this node to obtain the BSM location name from BSM's RTSM. The node list file for the BSM BPM collection must include the information shown in the following table:

   Node List File Fields and Values

Field	Value
bpmserver.hostdnsname	The fully-qualified domain name of the RTSM DPS server.
bpmserver.port	`21212`: The port used to connect to the RTSM DPS server.
bpmserver.username	`admin`: The user name to use for connecting to the RTSM DPS server.

   To edit the node list file, do the following from the OpsA Server Appliance:

   a. Copy the `sample_BPM_node.properties` file from the `$OPSA_HOME/conf/collection/sample` directory to some location, such as `/tmp/mynodelist.properties`.

Note: The sample file mentioned in this step does not contain a password property. When using the `opsa-collection-config.sh` script (in a later step) to encrypt the password, this script prompts you for the password, encrypts it, and inserts it into the sample file.

b. Edit the `/tmp/mynodelist.properties` file, adding the appropriate information; then save your work.

2. Run the following command from the Operations Analytics Server Appliance to encrypt the password:
`$OPSA_HOME/bin/opsa-collection-config.sh -encrypt /tmp/mynodelist.properties`

Note: You will be prompted to enter the password after running this command.

3. Run the following command from the OpsA Server Appliance to create the collector configuration:
`$OPSA_HOME/bin/opsa-collection-config.sh -create -nodelist /tmp/mynodelist.properties -collectorhost <fully-qualified-domain-name of collector host> -source bpm -domain application -group performance -username opsatenantadmin`

Note: The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

Note: The `opsa-collection-config.sh` script uses the values of `source`, `domain`, and `group` to select the right predefined collection template to create the desired collection configuration.

4. Run the following command from the OpsA Server Appliance to validate the collection configuration you just created:

`$OPSA_HOME/bin/opsa-collection-config.sh -list -allversions -collectorhosts -username opsatenantadmin>`

To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its `version`, `domain`, and `group`.

5. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance :

`$OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost <fully-qualified domain name of the collector host> -username`

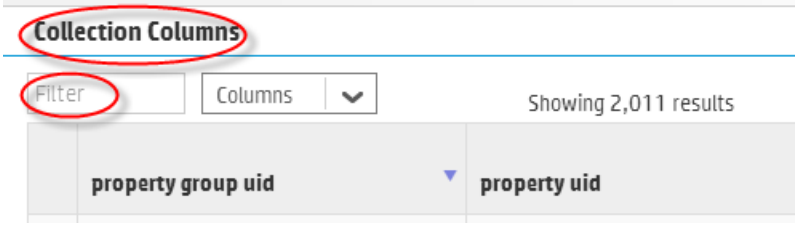
`opsatenantadmin`

The `-publish` option uploads the collection configuration you created to the OpsA Collector Appliance. To verify that the collector configuration published successfully, look for a message stating that the publish was successful, that a table was successfully created, and that the collection was restarted.

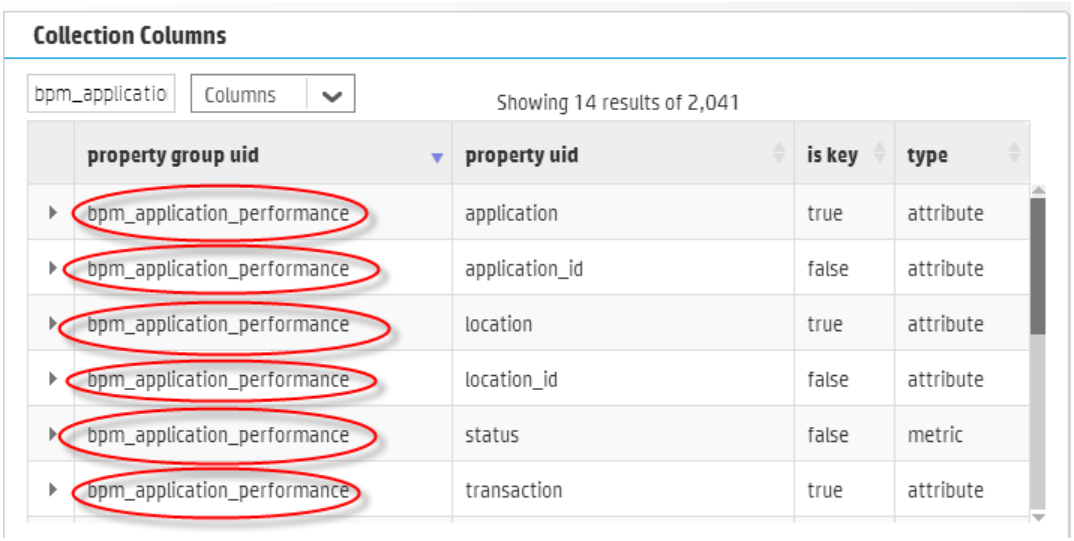
6. Let the collection run for five minutes or longer. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

> **Note:** The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this collection, you used a name of `bpm`, a domain of `application`, and a group of `performance` when creating the collection. The resulting property group uid would be `bpm_application_performance`.

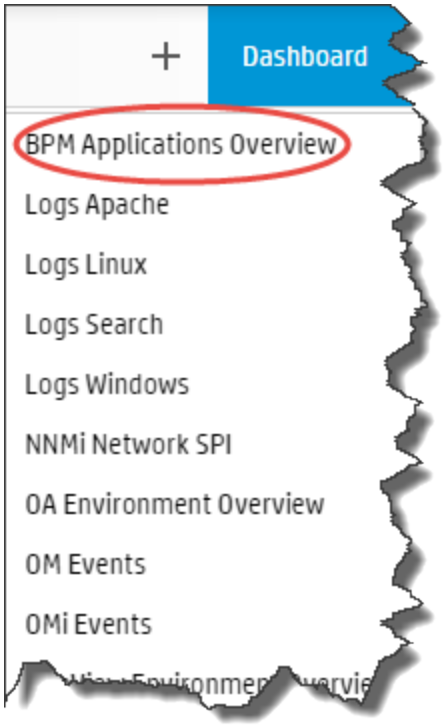
   a. Type the property group uid (`bpm_application_performance`) for this collection in the **Collection ColumnsFilter:**



   b. After typing property group uid (`bpm_application_performance`) for this collection in the **Collection ColumnsFilter**, you should see information for this collection.



7. From the Operations Analytics console, open the **BPM Application Overview** dashboard to view some of the collected information for this collection:



Configuring ArcSight Logger Out of the Box Smart Connector Collections (Detailed Method)

It is recommended that you use a more automated method of configuring this collection. See "Configuring ArcSight Logger Out of the Box Smart Connector Collections" on page 67 for the recommended method. For a more detailed approach, you can complete the following steps for configuring Out of the Box Smart Connectors.

The following steps are required to configure and publish the out of the box SmartConnectors. For more information about these connectors, see *Out of the Box Log Content* in the *Operations Analytics Installation Guide.*

Note: From a resource perspective, there is a limit to the number of Logger sessions supported by Operations Analytics (OpsA) Software. HP strongly recommends that, when you configure Logger collections, you assign those Logger collections to one common OpsA collector. Doing so reduces the number of Logger sessions.

1. Run the following command from the OpsA Server to confirm that you have the required templates:

   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -list -templates -username
   opsatenantadmin
   ```

   If the required templates are present, the result will include the following line (the required template names are bold in this example):

   ```
   1] Source: arcsight
    Version: 1.0, Domain: apache [access, error], Domain: linux
   [syslog], Domain: windows [event], Domain: cisco [ios]
   ```

2. If you do not have the required templates, place the template xml files in the following locations:

   ```
   /opt/HP/opsa/conf/collection/server/config.templates/arcsight/1.0/apa
   che/access/apache_access.xml

   /opt/HP/opsa/conf/collection/server/config.templates/arcsight/1.0/apa
   che/error/apache_error.xml

   /opt/HP/opsa/conf/collection/server/config.templates/arcsight/1.0/lin
   ux/syslog/linux_syslog.xml

   /opt/HP/opsa/conf/collection/server/config.templates/arcsight/1.0/win
   dows/event/windows_event.xml

   /opt/HP/opsa/conf/collection/server/config.templates/arcsight/1.0/cis
   co/ios/cisco_ios.xml
   ```

3. Run the following commands from the Operations Analytics Server Appliance to create the collector configuration:

   Note: You will be prompted to enter the password after running each of the following commands. You changed the opsatenantadmin user password during installation.

   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -create -collectorhost <OpsA
   collector IP Address> -source arcsight -domain apache -group access -
   username opsatenantadmin

   $OPSA_HOME/bin/opsa-collection-config.sh -create -collectorhost OpsA
   collector IP Address> -source arcsight -domain apache -group error -
   username opsatenantadmin

   $OPSA_HOME/bin/opsa-collection-config.sh -create -collectorhost OpsA
   collector IP Address> -source arcsight -domain windows -group event -
   username opsatenantadmin
   ```

```
$OPSA_HOME/bin/opsa-collection-config.sh -create -collectorhost OpsA
collector IP Address> -source arcsight -domain linux -group syslog -
username opsatenantadmin

$OPSA_HOME/bin/opsa-collection-config.sh -create -collectorhost OpsA
collector IP Address> -source arcsight -domain cisco -group ios -
username opsatenantadmin
```

4. Run the following command to confirm that the Logger is configured for the opsa tenant:

   ```
   $OPSA_HOME/bin/opsa-logger-config-manager.sh -list -loginUser
   opsatenantadmin -loginPassword <password>
   ```

   If the result does not include the IP address of the ArcSight Logger machine, run the following command:

   ```
   $OPSA_HOME/bin/opsa-logger-config-manager.sh -add -host <Logger
   machine IP address> -username <Logger user name> -password <Logger
   password> -loginUser opsatenantadmin -loginPassword <password> -
   loggerType arcsight -port 443 -sslEnabled true
   ```

   Note: For the user name in the above command, use the `-username` option for the HP ArcSight Logger username and the `-loginUser` option for the Tenant Admin user. See the *opsa-logger-config-manager.sh* reference page (or the Linux manpage) for more information.

5. A node list file contains details about the sources from which you plan to collect information. You need to modify the IP address in the following node list files, which are located in the following directory: `$OPSA_HOME/conf/collection/sample`

   - apache_access_node.properties
   - apache_error_node.properties
   - linuxlog_node.properties
   - winevent_node.properties

   In each file, set the IP address of ArcSight Logger server in the `arcsightserver1.hostdnsname` variable. For example:

   ```
   server.names = arcsightserver1

   ##node properties for 'Arcsight'
   arcsightserver1.hostdnsname = <ArcSight Logger IP>
   ```

   The IP address should be the same as the IP address displayed in the previous step.

6. As a non-root user, run the following command to publish the node list file to the collector host:

```
$OPSA_HOME/bin/opsa-collection-config.sh -nodelist
/opt/HP/opsa/conf/collection/sample/apache_access_node.properties -
collectorhost OpsA collector IP Address> -source arcsight -domain
apache -group access -username opsatenantadmin -password <password>

./opsa-collection-config.sh -nodelist
/opt/HP/opsa/conf/collection/sample/apache_error_node.properties -
collectorhost OpsA collector IP Address> -source arcsight -domain
apache -group error -username opsatenantadmin -password <password>

./opsa-collection-config.sh -nodelist
/opt/HP/opsa/conf/collection/sample/winevent_node.properties -
collectorhost OpsA collector IP Address> -source arcsight -domain
windows -group event -username opsatenantadmin -password <password>

./opsa-collection-config.sh -nodelist
/opt/HP/opsa/conf/collection/sample/linuxlog_node.properties -
collectorhost OpsA collector IP Address> -source arcsight -domain
linux -group syslog -username opsatenantadmin -password <password>

./opsa-collection-config.sh -nodelist
/opt/HP/opsa/conf/collection/sample/cisco_ios_node.properties -OpsA
collector IP Address> -source arcsight -domain cisco -group ios -
username opsatenantadmin -password <password>
```

7. Run the following command from the OpsA Server Appliance to validate the collection configuration that you just created:

   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -list
    -collectorhosts -allversions -username opsatenantadmin
   ```

   To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its version, domain, and group.

8. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance:

   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost
   <fully-qualified domain name of the collector host> –username
   opsatenantadmin
   ```

   The `-publish` option uploads the collection configuration you created to the OpsA Collector Appliance. To verify that the collector configuration published successfully, look for a message stating that the publish was successful, that a table was successfully created, and that the collection was restarted.

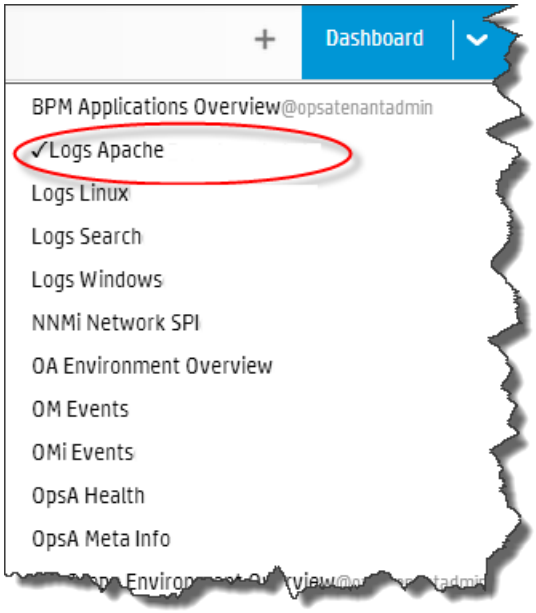
9. To confirm that the process ran correctly, wait five minutes and then open the following log file on

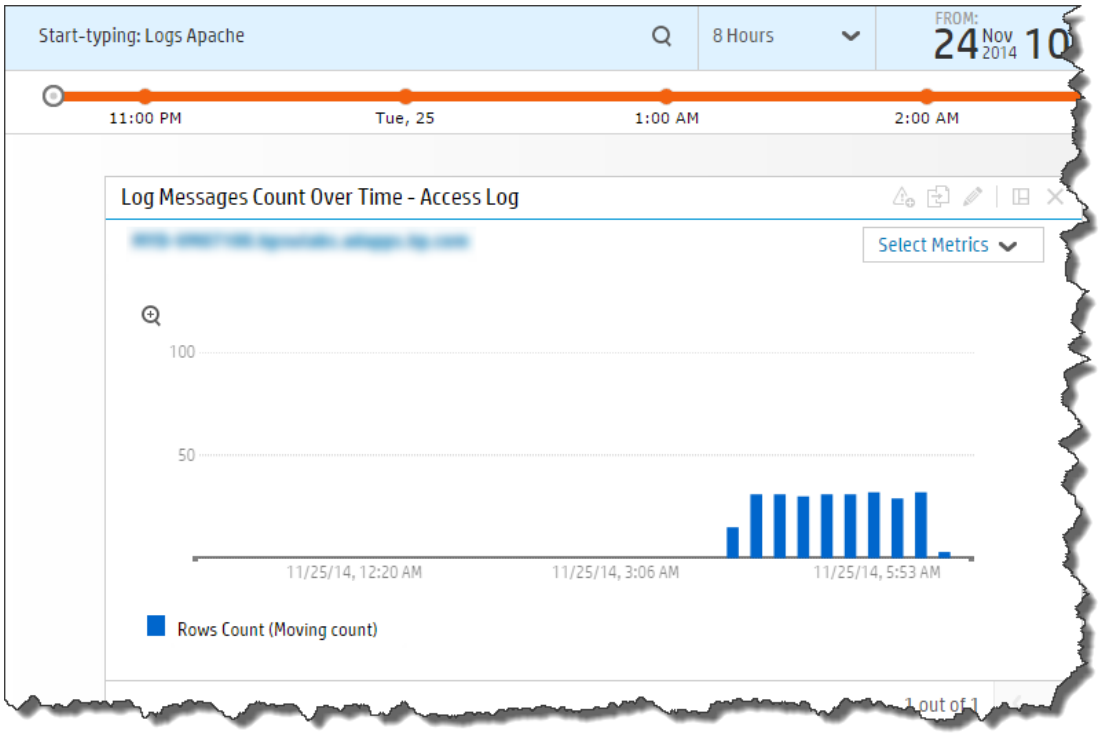
the OpsA Collector server:

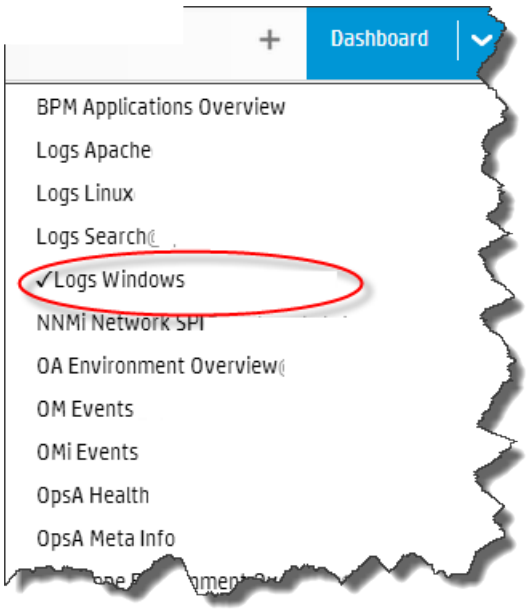
```
$OPSA_HOME/log/opsa-collector.log
```

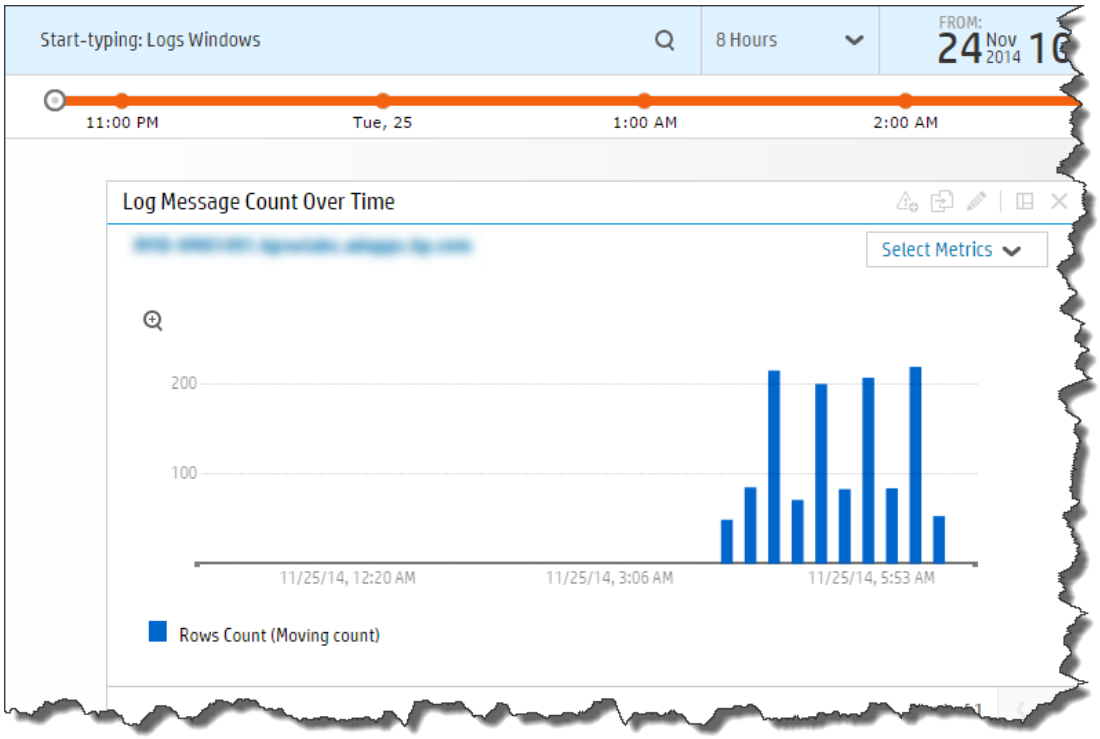
and confirm that it does not contain any errors.

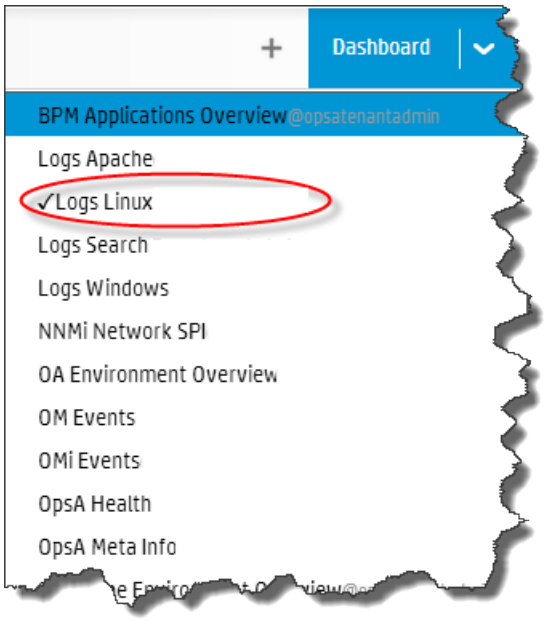
10. From the Operations Analytics console, open the following dashboards to view some of the collected information for these collections:




Start-typing: Logs Apache
8 Hours
FROM:
24 Nov 2014 10
11:00 PM
Tue, 25
1:00 AM
2:00 AM
Log Messages Count Over Time - Access Log
Select Metrics
100
50
11/25/14, 12:20 AM
11/25/14, 3:06 AM
11/25/14, 5:53 AM
Rows Count (Moving count)
1 out of 1


Dashboard
BPM Applications Overview
Logs Apache
Logs Linux
Logs Search
✓Logs Windows
NNMi Network SPI
OA Environment Overview
OM Events
OMi Events
OpsA Health
OpsA Meta Info


Start-typing: Logs Windows
8 Hours
FROM:
24 Nov 2014 16
11:00 PM
Tue, 25
1:00 AM
2:00 AM
Log Message Count Over Time
Select Metrics
200
100
11/25/14, 12:20 AM
11/25/14, 3:06 AM
11/25/14, 5:53 AM
Rows Count (Moving count)


Dashboard
BPM Applications Overview @opsatenantadmin
Logs Apache
✓Logs Linux
Logs Search
Logs Windows
NNMi Network SPI
OA Environment Overview
OM Events
OMi Events
OpsA Health
OpsA Meta Info

Configuring a Custom CSV Collection (Detailed Method)

It is recommended that you use a more automated method of configuring this collection. See "Configuring a Custom CSV Collection" on page 88 for the recommended method. For a more detailed approach, you can complete the following steps for the Custom CSV Collection.

Operations Analytics (OpsA) supports predefined collection templates for configuring data collections using the data sources described in "Configuring Collections using Predefined Templates " on page 25.

To collect data from sources that do not use predefined collection templates, consider configuring a Custom CSV collection. Use the following list to determine if a Custom CSV collection might work for you:

- The data source must provide Comma-separated values (CSV) data. CSV data is the only method that OpsA provides to collect data (instead of those predefined or custom collection methods described in " Configuring Tenants and Collections " on page 17
- The data source must collect CSV data based on time. There has to be a time and date column for each row in the CSV file. Both the time and date must be in that same column.

  Note: If this is not true, you must merge these columns before creating the collection.

- The data source cannot exceed 1549 data columns. If you try to create a Customer CSV collection containing more than 1549 data columns, the collection creation will fail.
- The maximum supported CSV file size is 500 MB.
- Data from the CSV data source must be accessible to the OpsA Collector Appliance.
- The CSV file can be local or remote to the collector and is assumed to be available in the source directory at regular intervals.
- Each column should have a header.
- Column names should not contain any spaces.
- The CSV file will be used to create a table in Vertica. Vertica objects include tables, views, and columns. Your CSV file must use the following naming conventions:
  - A column name must be from 1 to 128 characters long.
  - A column name must begin with a letter (A through Z), diacritic marks, or non-Latin characters (200-377 octal).

- A name cannot begin with an underscore (_). Leading underscores are reserved for system objects.
- Names are not case sensitive. For example, `CUSTOMER` and `Customer` represent the same names. However, if you enclose a name in quotation marks, it is case sensitive.

  Note: Object names are converted to lowercase when they are stored in the Vertica database.

- A name cannot match a Vertica reserved word such as `WHERE`, `VIEW`, `Table`, `ID`, `User`, or `Query`.
- A name cannot match the another Vertica object that has the same type.
- The Maximum number of columns cannot exceed 1549 in a CSV file as that value is a Vertica limitation when creating a table.

- The CSV file format has to be uniform for a single collection. For example, if you create a collection with 10 columns, the subsequent files that are provided for import within OpsA must have same format, including column names and data types.

For an example of data you might choose to collect using a Custom CSV collection, see the *Creating a Content Pack for Operations Analytics* White Paper at https://hpln.hp.com.

Note: Do the following to locate the *Creating a Content Pack for Operations Analytics* White Paper:

1. Select **Products**.
2. Navigate to the **Operations Intelligence (OpsA)** product, then click **Operations Intelligence**.
3. Click **Resources**, then locate the *Creating a Content Pack for Operations Analytics* White Paper.

Important Prerequisite Steps

Complete the following prerequisite work before configuring your Custom CSV Collection using the steps in "Configuration Steps" on page 271:

1. Tenants enable you to separate information from these data sources into groups, such as collections, user accounts, and user groups. When running the commands in this chapter, the tenant model you select affects which Tenant Admin user you will use. Use one of the following tenant models:

- **Default Tenant**: If you plan to use the default tenant, opsa_default, use opsatenantadmin as the tenant admin user and opsatenantadmin as the default tenant admin when running the commands in this chapter.
- **Use your own Tenant**: If you plan to configure a new tenant or use an existing tenant (other than the Default Tenant), see "Creating Tenants " on page 20. If you use this option, you will need to use the tenant admin user and password you created when running the commands in this chapter.

2. For the Custom CSV Collection, your data must be available in CSV format. If your data is not available in CSV format, you must find a way to convert the data, or the Custom CSV Collection will not work for you.

3. Choose the `<filename>.csv` file you want to load into the OpsA database. For OpsA, consider that most `<filename>.csv` files for a CSV collection will have a CSV file with a header and at least one row of data. For this example, assuming this sample file name is `your_file.csv`, copy the `your_file.csv` file to the `/tmp` directory.

   For OpsA, consider that most `<filename>.csv` files for a CSV collection will have a CSV file with a header and at least one row of data. For example, the header could include two columns: one with data and one with the time and date.

   Note: The `your_file.csv` sample file contains a good sample of data. OpsA uses this sample data to determine the data types and meta data to place in the `<your_template_name>.xml` sample file used in these instructions.

   Do not include any of the values from the following table in the header, as Vertica does not permit these values to be used as header names

Reserved Words for Vertica (do not use these values in the header name)

Index Letter	Value
A	ALL, ANALYSE, ANALYZE, AND, ANY, ARRAY, AS, ASC
B	BINARY, BOTH
C	CASE, CAST, CHECK, COLUMN, CONSTRAINT, CORRELATION, CREATE, CURRENT_DATABASE, CURRENT_DATE, CURRENT_SCHEMA, CURRENT_TIME, CURRENT_TIMESTAMP, CURRENT_USER
D	DEFAULT, DEFERRABLE, DESC, DISTINCT, DO
E	ELSE, ENCODED, END, EXCEPT
F	FALSE, FOR, FOREIGN, FROM

Reserved Words for Vertica (do not use these values in the header name), continued

Index Letter	Value
G	GRANT, GROUP, GROUPED
H	HAVING
I	IN, INITIALLY, INTERSECT, INTERVAL, INTERVALYM, INTO
J	JOIN
K	KSAFE
L	LEADING, LIMIT, LOCALTIME, LOCALTIMESTAMP
M	MATCH
N	NEW, NOT, NULL, NULLSEQUAL
O	OFF, OFFSET, OLD, ON, ONLY, OR, ORDER
P	PINNED, PLACING, PRIMARY, PROJECTION
R	REFERENCES
S	SCHEMA, SEGMENTED, SELECT, SESSION_USER, SOME, SYSDATE
T	TABLE, THEN, TIMESERIES, TO, TRAILING, TRUE
U	UNBOUNDED, UNION, UNIQUE, UNSEGMENTED, USER, USING
W	WHEN, WHERE, WINDOW, WITH, WITHIN

4. Choose the following parameter values to use when running the `opsa-csv-template-gen.sh` script:
   - **name**: Choose a name that accurately describes the data you plan to collect. For example, you might choose the name `mycsv` for the source.
   - **domain**: Choose a domain that accurately describes a domain in which the data you plan to collect resides. For example, you might choose the domain `birds`, to support the example in this section.
   - **group**: Choose a group that accurately describes the group for which you plan to collect data. For example, you might choose the domain `eagle`, to support the example in this section.

   See the *opsa-csv-template-gen.sh* reference page (or the Linux manpage), for more information.

5. Choose the following parameter values you plan to use when running the `opsa-collection-config.sh` script:
   - **source** : For the custom CSV collections, always use `custom` for the source.
   - **domain**: Use the domain that you selected in the previous step.

- **group**: Use the group that you selected in the previous step.

See the *opsa-collection-config.sh* reference page (or the Linux manpage), for more information.

Configuration Steps

After you complete the steps in this section, the Custom CSV Collection reads data from the CSV files within 60 seconds of the file being placed in the source directory.

For several examples of data you might choose to collect using a Custom CSV collection, see the *Creating a Content Pack for Operations Analytics* White Paper at https://hpln.hp.com.

Note: Do the following to locate the *Creating a Content Pack for Operations Analytics* White Paper:

1. Select **Products**.
2. Navigate to the **Operations Intelligence (OpsA)** product, then click **Operations Intelligence**.
3. Click **Resources**, then locate the *Creating a Content Pack for Operations Analytics* White Paper.

1. Do the following from the OpsA Server Appliance
   a. Run the following command to create a template for this new collection based on the sample data in the `your_file.csv` file:
   ```
   $OPSA_HOME/bin/opsa-csv-template-gen.sh -inputfile /tmp/your_file.csv -name mycsv -domain birds -group eagle -sourcedir /opt/HP/opsa/data/<mydata> -datecolumn Time -dateformat MM/dd/yyyy hh:mm:ss -timezone GMT+0 -filepattern '*.csv' -grouptype metrics -key String, Usage in MHz
   ```

   Note: To specify a time zone that supports Daylight Savings Time, use the desired daylight savings time value you need as the `timezone` attribute. See "Daylight Savings Time Codes" on page 294 for a list of valid timezone attributes.

   Note: You must define at least one property as a key column using the `-key` option. Do not specify a timestamp or metric as a key column with the `-key` option.

   After this command completes, it creates the `<your_template_name>.xml` file and displays the path to this file. The `<your_template_name>.xml` file is a collection template created from the `your_file.csv` file. Look for a message similar to the following:
   ```
   Generated the Custom CSV collection template
   ```

```
/opt/HP/opsa/conf/collection/server/config.templates/custom/1.0/birds/eagle/mycsv.xml
```

b. Create the following directory on the OpsA Collector Appliance:
`/opt/HP/opsa/data/<mydata>`

c. Run the following command from the OpsA Collector Appliance to set the correct file ownership:
`chown opsa /opt/HP/opsa/data/<mydata>`

See the *opsa-csv-template-gen.sh* reference page (or the Linux manpage) for more information.

Note: The purpose of the `-datecolumn`, `-dateformat`, and `-timezone` options is to identify one column from the `your_file.csv` file as the `timestamp` column for the database table. This column selection is mandatory for OpsA collections using metric tables. These options are provided to help you, as the OpsA administrator, identify the correct column.

Note: When creating a custom CSV template, do not use a column named `timestamp_utc`, as doing so causes and error when you attempt to publish the collection. If you already registered a collection see "Removing a Collection Registration for a Tenant" on page 170 for instructions about removing the registration for this collection.

Note: As an example, suppose you plan to use `your_file.csv` as your CSV file, and that it contains the following information:

a. Using this information in an example, you would use the following command to create your custom CSV template:

```
$OPSA_HOME/bin/opsa-csv-template-gen.sh -inputfile /tmp/your_file.csv -name mycsv -domain birds -group eagle -sourcedir /opt/HP/opsa/data/mydata -datecolumn Time -dateformat MM/dd/yyyy hh:mm:ss -timezone GMT-7 -filepattern *.csv -grouptype metrics -key String
```

After this command completes, look for a message similar to the following:

```
Generated the Custom CSV collection template /opt/HP/opsa/conf/collection/server/config.templates/custom/1.0/birds
/eagle/mycsv.xml
```

b. Create the following directory on the OpsA Collector Appliance:
`/opt/HP/opsa/data/mydata`

c. Run the following command from the OpsA Collector Appliance to set the correct file ownership:
`chown opsa /opt/HP/opsa/data/mydata`

Note: As an example, suppose you plan to use `your_file.csv` as your CSV file, and that it contains the following information:

Time,Value1,String1
```
02/23/2014 23:42:00,6.543,eagle
02/23/2014 23:52:00,7.543,eagle
02/23/2014 23:62:00,8.543,eagle
```

Use the following pattern letters when configuring the date format to use when parsing date strings:

Letter	Date or Time Component	Presentation	Examples
G	Era designator	Text	AD
Y	Year	Year	1996; 96
M	Month in Year	Month	July; Jul; 07
w	Week in Year	Number	27
W	Week in month	Number	2
D	Day in year	Number	189
d	Day in month	Number	10
F	Day of week in month	Number	2
E	Day in week	Text	Tuesday; Tue
a	Am/Pm marker	Text	PM
H	Hour in day (0-23)	Number	0
k	Hour in day (1-24)	Number	24
K	Hour in am/pm (0-11)	Number	0
h	Hour in am/pm (1-12)	Number	12

Letter	Date or Time Component	Presentation	Examples
m	Minute in hour	Number	30
s	Second in minute	Number	55
S	Millisecond	Number	978
z	Time zone	General time zone	Pacific Standard Time; PST; GMT-08:00
Z	Time zone	RFC 822 time zone	-0800

The following examples show how to interpret date and time patterns in the U.S. locale. The given date and time are 2001-07-04 12:08:56 local time in the U.S. Pacific Time time zone.

Date and Time Pattern	Result
"yyyy.MM.dd G 'at' HH:mm:ss z"	2001.07.04 AD at 12:08:56 PDT
"EEE, MMM d, ''yy"	Wed, Jul 4, '01
"h:mm a"	12:08 PM
"hh 'o''clock' a, zzzz"	12 o'clock PM, Pacific Daylight Time
"K:mm a, z"	0:08 PM, PDT
"yyyyy.MMMMM.dd GGG hh:mm aaa"	02001.July.04 AD 12:08 PM
"EEE, d MMM yyyy HH:mm:ss Z"	Wed, 4 Jul 2001 12:08:56 -0700
"yyMMddHHmmssZ"	010704120856-0700
"yyyy-MM-dd'T'HH:mm:ss.SSSZ"	2001-07-04T12:08:56.235-0700
"MM/dd/yyyy hh:mm:ss" To use epoch time, substitute `-dateformat s` for the `-dateformat MM/dd/yyyy hh:mm:ss` option as shown in following example: `-dateformat s 1002197336` Look at the resulting epoch time shown in the next column:	10/04/2001 12:08:56 1002197336 (the epoch equivalent of 10/04/2001 12:08:56)

2. You must have a registered an OpsA Collector Appliance for the Custom CSV collections you plan to configure.

To check the registration status of your collector appliance, do the following:

a. `$OPSA_HOME/bin/opsa-collection-config.sh -list -collectorhosts -allversions -username opsatenantadmin`

b. Review the list of registered collectors. If the collector you plan to register is not on the list, you must register it using the instructions in this section.

See "Registering Each Collector Appliance" for more information.

3. Using this example as a guideline, review the collection properties in the `/opt/HP/opsa/conf/collection/server/config.templates/custom/1.0/birds/eagle/mycsv.xml` file:

   a. Change the property type (`type="attribute"` or `type="metric"`).

   > **Note**: A collection property can be either an attribute (a descriptor for an entity, such as `host_name`) or a metric (typically a measurement), such as `CPU utilization`.

   b. . Review the attributes and decide which of them are key attributes. Set up two three key attributes by setting `key="yes"` for the desired key attribute.

   > **Note**: You can use keys to uniquely identify an entity instance so users can narrow a search within a single collection or match metrics for one entity (a collection row) to the same metric or a related entity across collections.

   > **Note**: When using PQL and the `withkey` command, you can use up to three key column values for a singly query. See *About the Phrase Query Language* in the *OpsA Help* for more information.

   c. Save your work.

4. **Optional Step**: You might have a need to transform data before it is stored in the OpsA database. You can do this by editing the `<your_template_name>.xml` file and adding transform methods.

   OpsA provides the following methods for transforming data:

   - add(x)
   - subtract(x)
   - multiply(x)
   - divide(x)
   - replace(x)
     where x is a float data type.

- concat(str)
- replace(str)
- replacewith(currentStr, newStr)
  where Str represents string for a string data type

For example, consider the following column description:

```
<column name="CPU Utilization" position="9" datatype="float"
label="CPU Utilization" columnname="cpu_util" length="0" key="no"
type="metric" tags="utilization,performance,primary" mapsto=""
unit="%" value="" />
```

In this example, you want the column description to read as follows:

```
<column name="CPU Utilization" position="9" datatype="float"
label="CPU Utilization" columnname="cpu_util" length="0" key="no"
type="metric" tags="utilization,performance,primary" mapsto=""
unit="%" value="multiply(100)" />
```

To add the transform, edit the `<your_template_name>.xml` file, add `value-multiply (100)` to the column for CPU Utilization, then save your work.

Valid Values for unit

You can use any of the following entries for the unit field:

```
"%"
"bytes"
"mbps"
"kbps"
"gbps"
"kb"
"mb"
"gb"
"hz"
"khz"
"mhz"
"ghz"
"BIT"
"PB"
"EB"
"W"
"V"
"A"
"secs"
"millisecs"
"ms"
"pages/sec"
"per second"
"switches/sec"
"bytes/sec"
```

```
"KB/sec"
"interrupts/sec"
"pages/sec"
"errors/sec"
"reads/sec"
"bps"
"per hour"
"per min"
```

5. For this *birds* example, run the following command from the Operations Analytics Server Appliance to create the collector configuration:
`$OPSA_HOME/bin/opsa-collection-config.sh -create -collectorhost <fully-qualified domain name of the collector host> -source custom -domain birds -group eagle -username opsatenantadmin`

   Note:The `opsa-collection-config.sh` script prompts you for the Tenant Admin password for the username you use in the `opsa-collection-config.sh` command.

   Note: When you run the command in this step, always use the `-source custom` argument when creating a custom CSV collector configuration.

   To create and publish collections supported by OpsA, you normally provide source, domain, and group options to the `opsa-collection-config.sh` script. The definition for each of these options is as follows:
   - **source**: Specifies the name of the source collector.
   - **domain**: Specifies the domain name to which the collected data belongs.
   - **group**: Specifies the group name to which the collected data belongs.

   Note: The `opsa-collection-config.sh` script uses the values of `source`, `domain`, and `group` to select the right collection template and create the desired collection configuration.

   To see the predefined values for these options, see the *opsa-collection-config.sh* reference page (or the Linux manpage).

   Note: Although the *opsa-collection-config.sh* reference page provides you with the predefined values for these options, use the `custom` source option along with options that differ from the predefined values for the `domain` and `group` options when creating Custom CSV Collections.

6. Run the following command from the OpsA Server Appliance to publish this collection configuration to the OpsA Collector Appliance :

   `$OPSA_HOME/bin/opsa-collection-config.sh -publish -collectorhost`

```
<fully-qualified domain name of the collector host> –username
opsatenantadmin
```

The `–publish` option uploads the collection configuration you created to the OpsA Collector Appliance. To verify that the collection configuration published successfully, look for a message stating that the publish was successful, that a table was successfully created, and that the collection was restarted.

7. Run the following command from the OpsA Server Appliance to validate the collection configuration you just created:

   ```
   $OPSA_HOME/bin/opsa-collection-config.sh -list -allversions -
   collectorhosts -username opsatenantadmin
   ```

   To verify a successful collection configuration creation, look for a message showing the expected collector host name, tenant name, and tenant version. The message should also include the source template details such as its version, domain, and group.

8. You must copy the data files (or set up some way of automatically copying the data files) from the data source to the OpsA Collector Appliance and set the correct file ownership. Do the following for the collection you plan to configure:
   a. Copy the files to the following directory on the OpsA Collector Appliance: `$OPSA_HOME/data/mydata` (or to the directory that relates to the custom collection you created).

   b. Run the following command from the OpsA Collector Appliance to set the correct file ownership:
   ```
   chown opsa $OPSA_HOME/data/mydata
   ```

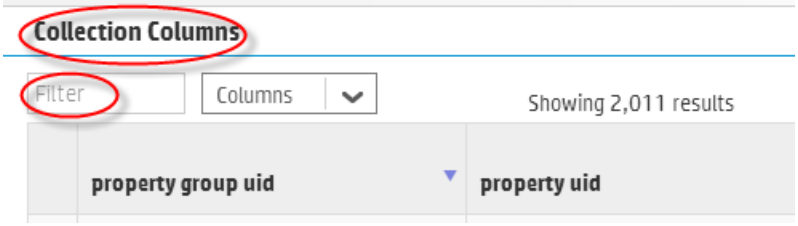
   After completing this step, you should see data in the `$OPSA_HOME/data/mydata_processed` folder within a few minutes.

   Note: After OpsA processes data in the `yourfile.csv` file, it removes the `yourfile.csv` file from the `$OPSA_HOME/data/mydata` directory and creates the `$OPSA_HOME/data/mydata_processed` folder and its contents.

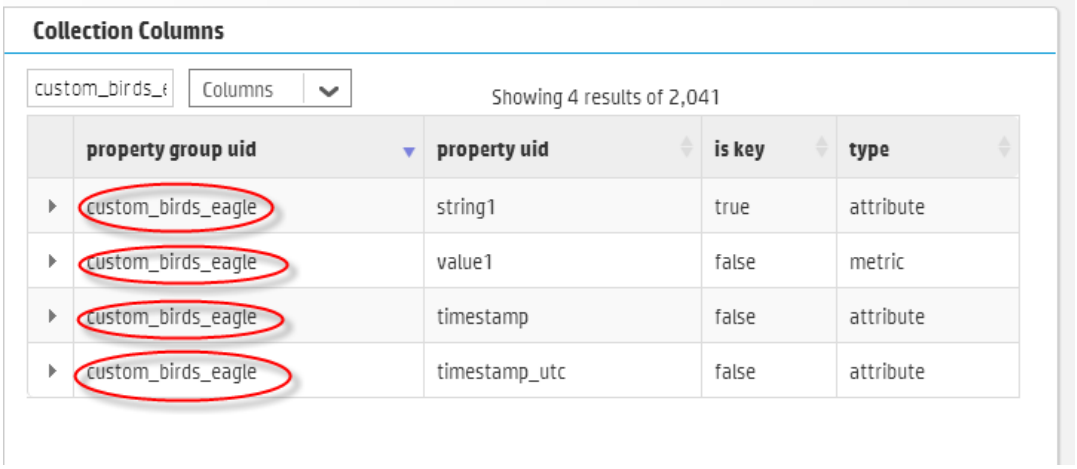
9. From the Operations Analytics console, view the **OpsA Meta Info** dashboard. Look for the **property group uid** for the collection you just created and published.

   Note: The property group uid consists of a combination of the `source`, `domain`, and `group` parameters you used to create the collection. For this example, you would have used a name of `custom`, a domain of `birds`, and a group of `eagle` when creating the collection. The resulting property group uid would be `custom_birds_eagle`.

   a. Type the property group uid (`custom_birds_eagle`) for this collection in the **Collection ColumnsFilter:**



b. After typing property group uid (`custom_birds_eagle`) for this collection in the **Collection ColumnsFilter**, you should see information in the resulting table:



10. Create dashboards and query panes for the data you are now collecting. Follow the instructions shown in the *Dashboards and Query Panes* section of the *Operations Analytics Help* for information about creating dashboards and query panes.

11. Create AQL functions for the data you are now collecting. See the instructions shown in the *Define Analytic Query Language (AQL) Functions* section of the *Operations Analytics Help* for information about creating AQL functions.

12. If you want to add tags to a Custom CSV Collection, use the `opsa-tag-manager.sh` command . See "Creating, Applying, and Maintaining Tags" on page 116 and the *opsa-tag-manager.sh* reference page (or the Linux manpage) for more information.

Troubleshooting the Custom CSV Collection

If you suspect problems with your Custom CSV Collection, do the following:

1. To check the registration status of your collector appliance, do the following:
   a. `$OPSA_HOME/bin/opsa-collection-config.sh -list -collectorhosts -allversions -username opsatenantadmin`

   b. Review the list of registered collectors. If the collector you plan to register is not on the list, you must register it using the instructions in this section.

2. View the collected data to make sure it is what you expect. If it is not, continue checking the remaining items in this list.

3. Review the content of the `your_file.csv` file and the associated `<your_template_name>.xml` file to make sure it is configured to collect the right data.

4. You must use a CSV file for the Custom CSV Collection. Check the `<filename>.csv` file you loaded into the OpsA database. For OpsA, consider that most `<filename>.csv` files for a CSV collection will have a CSV file with a header and at least one row of data.

5. Check the quality of the data you are collecting. If it is not what you expected, review the content of the `<filename>.csv` file you loaded into the OpsA database, as it might not be collecting the right data for you.

Removing the Registration and Data for a Custom CSV Collection

To remove the registration for a Custom CSV Collection, do the following:

1. Run the following command:
   `$OPSA_HOME/bin/opsa-collection-config.sh -unregister -source custom -group group -domain domain -collectorhost <fully-qualified domain name of collector host>`

   Note: If you remove the registration for this CSV collection, and do not complete the remaining steps, remember the following important information:

   - The collected data remains intact and is not removed.
   - If you decide to register this collection again, you must not reuse the `your_file.csv` file, (or whatever csv file name you used to create the collection template), as you run the risk of duplicating the original collection data.
   - It is a best practice to complete all of these removal steps to avoid collecting duplicate data.

2. After unregistering this Custom CSV Collection, remove the collection from the database using the following command:
   `$OPSA_HOME/bin/opsa-collection-config.sh -purgecollection -source custom -domain domain -group group -collectorhost <fully-qualified domain name of collector host> -username opsatenantadmin`
   See the *opsa-collection-config.sh* reference page (or the Linux manpage), for more information.

   Note: The command in this step also removes all Custom CSV Collection data for the

specified tenant from the OpsA database.

Note: After unregistering a Custom CSV Collection, the data remains intact. This means that you can register a Custom CSV Collection that you removed and resume that Custom CSV Collection.

3. Remove the data. For example, for the NOAA example, you would remove the `$OPSA_HOME/data/noaaCustom_processed` directory.

See the *opsa-collection-config.sh* reference page (or the Linux manpage) and "Removing a Collection Registration for a Tenant" on page 170 for more information.

Configuring a Custom SiteScope Collection (Detailed Method)

It is recommended that you use a more automated method of configuring this collection. See "Configuring a Custom SiteScope Collection" on page 95 for the recommended method. For a more detailed approach, you can complete the following steps for the Custom SiteScope Collection.

After you complete the steps in this section, SiteScope starts sending data to the Custom SiteScope Collection. The Custom SiteScope Collection collects data as it arrives from SiteScope.

The following table shows the monitor types currently supported by the Custom SiteScope Collection:

Note: If a SiteScope monitor type has only unsupported counters configured, OpsA ignores that monitor type when creating the collection. OpsA does not support monitor counter names longer than 128 characters. If a supported monitor's counter name is longer than 128 characters, OpsA ignores that counter.

Supported Monitor Types

Apache	Port
BACIntegrationConfiguration	SAPPerformance
BACIntegrationStatistics	Script
Composite	Service
ConnectionStatisticsMonitor	SiebelApplicationServer
CPU	SolarisZones

Supported Monitor Types, continued

DatabaseCounter	SQLQuery
DHCP	SSLCertificatesStatus
Directory	Sybase
DiskSpace	UnixResources
DNS	URLContent
DynamicDiskSpace	URLMonitor
File	URLSequenceMonitor
FTPMonitor	VMware
HealthServerLoadMonitor	VMwareHostCPUMonitor
HyperVMonitor	VMwareHostMemoryMonitor
JMXMonitor	VMwareHostStateMonitor
LDAPMonitor	VMwareHostStorageMonitor
LogEventHealthMonitor	WebServer
LogMonitor	WebService
Memory	WebSphere
MicrosoftWindowsEventLog	WindowsPerformance
MQStatusMonitor	WindowsResources
MSActiveServerPages	WindowsServicesState
MSIISServer	XMLMetrics
MSSQLServer	
MSWindowsMediaServer	
NetworkBandwidthMonitor	
Oracle	
Ping	

Detailed Configuration Steps

Configuring a Collector Appliance by creating custom collector templates is a two-step process:

1. "Generating and Configuring Templates (Custom SiteScope Collection)" below
2. Continuing with step 2 below: Configuring SitesScope for Integrating Data with Operations Analytics

Generating and Configuring Templates (Custom SiteScope Collection)

To configure a Custom SiteScope Collection, you must use SiteScope Unit Of Measurement (UOM) files as an input for the `opsa-sis-collector-auto-conf.sh` script.

1. If you are using OpsA for SiteScope version 11.22IP or newer, skip this step (proceed to step 2). If you are using an earlier version of SiteScope (earlier than SiteScope version 11.22IP), complete one of the following options:

   - **Option 1**: Use the opsa-sis-collector-auto-conf.sh script with the default UOM file. The default UOM file is located at `$OPSA_HOME/conf/collection/sitescope_configuration/uom/data_integration_uom.xml`. Using the default UOM file, proceed to step 2.

     Note: OpsA includes a default UOM file, `$OPSA_HOME/conf/collection/sitescope_configuration/uom/data_integration_uom.xml`, which supports many of the metrics supported by OpsA. Use the `-uomfiles` option to optionally define a UOM folder path containing UOM files you manually extracted.

   - **Option 2**: To use metrics that are not supported by the default UOM file complete the steps shown below.

     i. Complete these substeps for each SiteScope server from which you plan to collect data.

        A. Using the SiteScope UI, navigate to the **Diagnostics Integration Preferences** page (**Using SiteScope** > **Preferences** > **Integration Preferences** > **Diagnostics Integration Preferences**)

        B. Click **Generate UOM XML**. Doing so creates the UOM XML file on the HP SiteScope server in the following location: `%SITESCOPE_HOME%\conf\integration\data_integration_uom.xml`.

     ii. Create an empty directory on the OpsA Server Appliance; then copy the generated UOM files to this newly created directory.

     Note: Rename the UOM files before you copy them to the newly created directory, as many of the generated UOM files might have the same name (`data_`

`integration_uom.xml`).

Note: When creating SiteScope collection templates, only place valid UOM files in the directory. Do not place any other files in that directory.

iii. Optional: You can use the `opsa-sis-collector-auto-conf` script to create complete collection templates for most of the monitor types shown in "Configuring a Custom SiteScope Collection (Detailed Method)" on page 281. However, there are a few created templates you might need to customize after you create them. For example, you might need to customize the template contents of the following SiteScope monitor types, as you should vary the template content to match the data you configure the monitor to collect:

- JMXMonitor
- XMLMetrics

There are two tasks you might need to complete when customizing the creation of a SiteScope collection template for a particular monitor type:

A. **Parsing the counter names to separate out metric names from instance attributes**: Create a regular expression definition in the `/opt/HP/opsa/conf/collection/sitescope_measure_regex_patterns/custom` directory. See the `/opt/HP/opsa/conf/collection/sitescope_measure_regex_patterns/custom/README_BEFORE_CREATING_PATTERNS.txt` file for specific instructions about creating regular expressions to parse the counter names for a SiteScope monitor type.

B. **Defining the data type, tags and units for a parsed metric**: Create a regular expression definition in the `/opt/HP/opsa/conf/collection/sitescope_metadata_patterns/custom` directory. See the `/opt/HP/opsa/conf/collection/sitescope_measure_regex_patterns/custom/README_BEFORE_CREATING_PATTERNS.txt` file for specific instructions about creating regular expression definitions for assigning data types, tags, and units to metrics for a SiteScope monitor type.

After you finish these steps, use the `-uomfiles` option in the next set of steps to define a UOM folder path containing UOM files you manually extracted:

2. Complete the following tasks to configure HP SiteScope to forward data to an Operations Analytics (OpsA) Collector Appliance.

Note: For SiteScope versions earlier than 11.10, you must configure SiteScope for Integrating data with OpsA before completing the steps in this section. To do this, complete

the instructions shown in "Configuring SiteScope for Integrating Data with Operations Analytics (Manual Method)" on page 289.

a. A node list file contains details about the sources from which you plan to collect information. The node list file for the Custom SiteScope Collections must include the information shown in the following table.

Note: Each of the following settings could be configured for a specific SiteScope server, such as `server1`. If the SiteScope server value is missing, the default setting is used. For example, if the "`<server1>.port = `" string does not exist in the node list file, OpsA uses the value of the "`default.port = `" setting for `server1`.

Node List Fields and Values

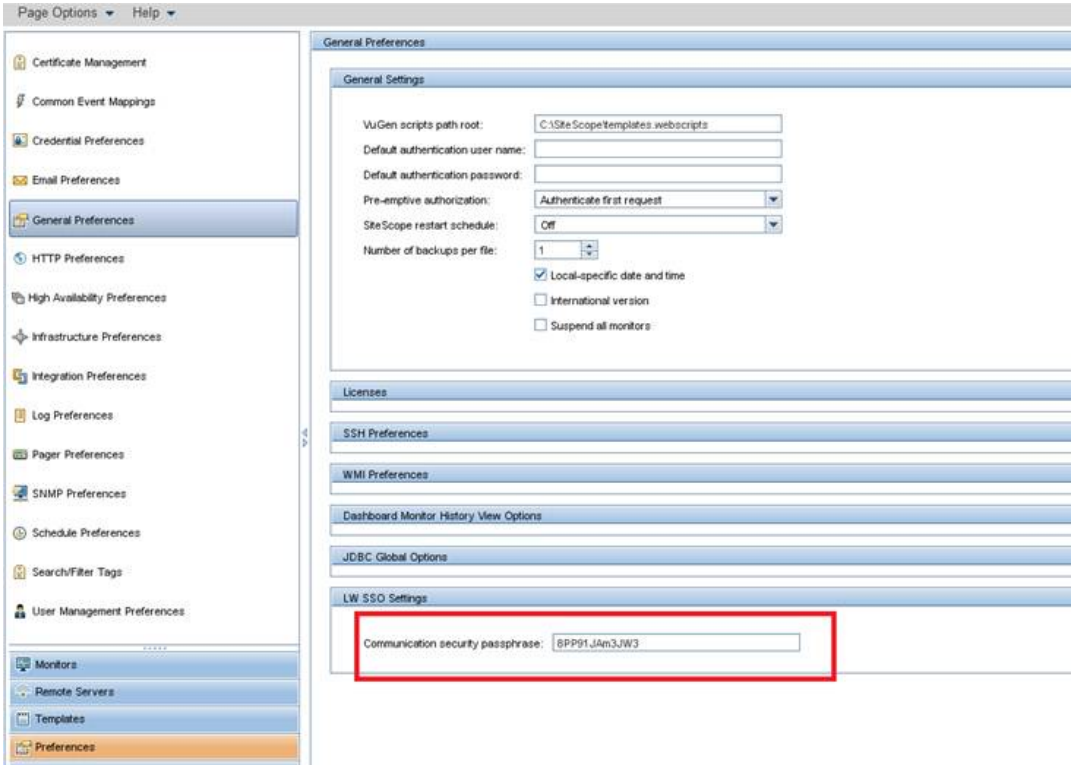
Field	Value
server.names	The aliases of the SiteScope server names, delimited by commas. These are the servers from which you plan to collect SiteScope information.
<*server*>.hostdnsname	IP Address or fully-qualified domain name of the SiteScope servers for which you are configuring collections. If you want to support failover for the SiteScope servers, specify all the SiteScope servers included in the failover configuration.
.port	The port used to connect to the SiteScope server. Set this if a server does not use the default.port value. The `server.port` setting could be configured for a specific server, such as `server1`. If the server value is missing, the default setting is used. For example if the "`<server1>.port =`" string does not exist in the node list file, OpsA uses the value of the "`default.port = `" setting for `server1`.
.username	The default user name used to connect to the SiteScope server. This is typically `admin`. This field might be set to empty (no value).
.initString	The default value of the initstring used for SSL communication with the SiteScope server. **You can obtain this initString from the SiteScope screen shown below this table.**

Node List Fields and Values, continued

Field	Value
.use_ssl	Set this field to `true` to enable SSL communication with the SiteScope server. The default setting is `false`. If you set `default.use_ssl=true`, you must export the certificate from the OpsA Collector Appliance and import this certificate on each SiteScope server. If SiteScope is enabled with SSL, do the following: i. Copy the SiteScope server's root server certificate to each OpsA Server and give the file full permissions. ii. Run the following command: `keytool -importcert -alias <certificate alias> -file <certificate file> -keystore /opt/HP/opsa/jdk/jre/lib/security/cacerts` See *Configuring SiteScope to Use SSL* in the *HP SiteScope Deployment Guide* and the *opsa-collector-manager.sh* reference page for more information.
.opsa_collector	The fully-qualified domain name or the IP address of the common collector that collects data from the SiteScope servers. Do not use `localhost` or `127.0.0.1`. **Note:** This IP address must be accessible from the SiteScope server.

Finding the initstring in SiteScope

Note: For the integration with OpsA to work correctly, you must enter the communication security passphrase (LW SSO Token) as shown in the example in following graphic. If this field is empty, see the SiteScope documentation for the instructions to set a value in this field.



View the sample node list file shown below:

```
server.names=sis01313,sis01388
#properties for sis01313 servers
sis01313.hostdnsname=sis1.somedomain.com
#properties for sis01388 server
sis01388.hostdnsname=sis2.somedomain.com
sis01388.port=18080
#common properties for sis servers
default.port=8080
default.username=admin
default.initString=8PP91JAm3JW3
default.use_ssl=false
default.opsa_collector=opsac
```

To edit the node list file, do the following from the OpsA Server Appliance:

Edit the `$OPSA_HOME/conf/collection/sitescope_configuration/sample_SiteScope_node.properties` file, adding the appropriate information from the examples shown above, then save your work.

Note: The sample file mentioned in this step does not contain a password property. When using the `opsa-collection-config.sh` script (in a later step) to encrypt the password, this script prompts you for the password, encrypts it, and inserts it into the

sample file.

b. Run the following command from the OpsA Server Appliance to encrypt the password:
`$OPSA_HOME/bin/opsa-collection-config.sh -encrypt $OPSA_HOME/conf/collection/sitescope_configuration/sample_SiteScope_node.properties`

Note: If the SiteScope password is empty, edit the nodelist file and remove the value from the appropriate `<server>.password` setting. For example, you might change the value to `sis01.password =`

Note: You will be prompted to enter the password after running this command.

c. Run the following command from the OpsA Server Appliance to create the collector configuration.
`$OPSA_HOME/bin/opsa-sis-collector-auto-conf.sh -nodelist $OPSA_HOME/conf/collection/sitescope_configuration/sample_SiteScope_node.properties -username opsatenantadmin -password <password>[-ignoretag] [-forceupdate] [-forcedelete] [-skipcontent] [-uomfiles] <path to directory containing UOM files>]`

Note: When running the `opsa-sis-collector-auto-conf.sh` script, you might see an error similar to the following :
`No implementation defined for org.apache.commons.logging.LogFactory.`

If this happens, run the command in this step from the `/opt/HP/opsa/bin/support/` directory.

Use the following option definitions for this command:

- The -nodelist options points to the node list file created earlier.
- opsatenantadmin is the default predefined Tenant Admin user for the predefined opsa_default tenant. If you are not using the default tenant, use the Tenant Admin user and password for the tenant you defined for your collections.
- opsatenantadmin is the password for the default predefined Tenant Admin user (for the predefined opsa_default tenant). If you are not using the default tenant, use the Tenant Admin user and password for the tenant you defined for your collections.
- Use the `-ignoretag` option to ignore the step of tagging the monitors within SiteScope. The opsa-sis-collector-auto-conf.sh script creates a tag named `opsa_<tenant-name>` and associates it with the root SiteScope group, which means that all monitors will be

recursively tagged automatically and dynamically. In some cases, you might want to configure only a subset of the monitors. In those situations, use the `ignoretag` option to manually handle the tagging.

- Use the `-forceupdate` option if you did not make any changes since the last time you ran the opsa-sis-collector-auto-conf.sh script, and still want to **force** the script to make changes in already saved SiteScope profiles. If you use the `-forceupdate` option when running the opsa-sis-collector-auto-conf.sh script, it deletes the old integration configuration and replaces it with the new configuration. For example, if you made some manual changes on the SiteScope profile side and want to return to the original configuration, use the `-forceupdate` option.

- Use the `-forcedelete` option if you want to remove SiteScope configurations made since you last ran the `$OPSA_HOME/bin/opsa-sis-collector-auto-conf.sh` script. To do this, remove the corresponding alias from the `server.names=` setting in the nodelist file, then run the `$OPSA_HOME/bin/opsa-sis-collector-auto-conf.sh` script using the `-forcedelete` option.

- As mentioned earlier, OpsA includes a default UOM file, `$OPSA_HOME/conf/collection/sitescope_configuration/uom/data_integration_uom.xml`, which supports many of the metrics supported by OpsA. Use the `-uomfiles` option to optionally define a UOM folder path containing UOM files you manually extracted.

After completing the configuration steps in this section, SiteScope begins forwarding data to the OpsA Collector Appliance based on the configuration choices you made.

Configuring SiteScope for Integrating Data with Operations Analytics (Manual Method)

The following tasks, showing steps and diagrams, explain an example of configuring HP SiteScope to forward data to an Operations Analytics (OpsA) Collector Appliance.

Note: You must complete the step in "Configuring a Custom SiteScope Collection (Detailed Method)" before completing the configuration steps in this section.

To configure SiteScope to send data to OpsA, you must complete 3 tasks:

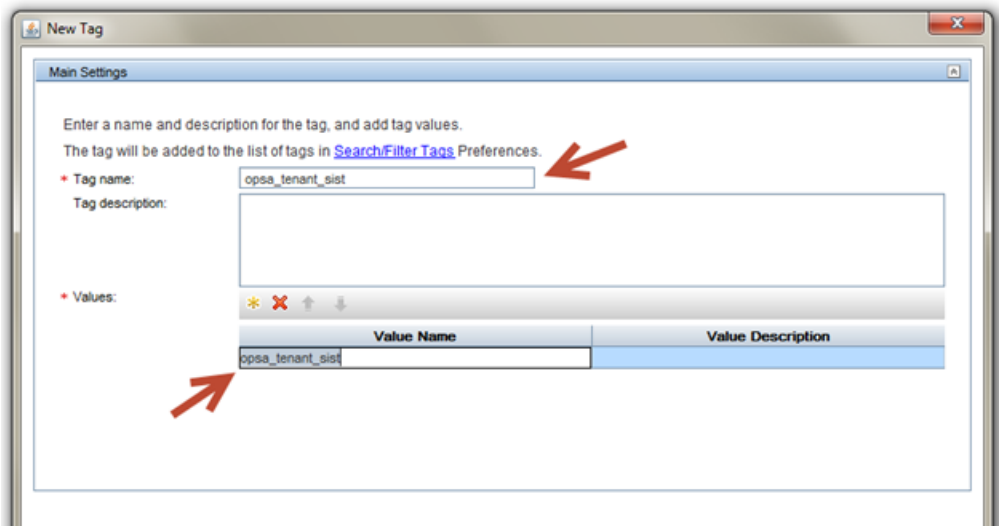
- "Task 1: Creating a SiteScope Tag" below
- "Task 2: Using the New SiteScope Tag to Mark the Monitor or Monitor Groups" on the next page
- "Task 3: Creating a New Data Integration Preference" on page 292

Task 1: Creating a SiteScope Tag

To create a SiteScope tag, do the following:

1. Log on to SiteScope as an **Admin** user.
2. Navigate to **Preferences** > **Search/Filter Tags**
3. Click the **New Tag icon** (the gold-colored star) to create a new tag.

   The following shows the window that should open:



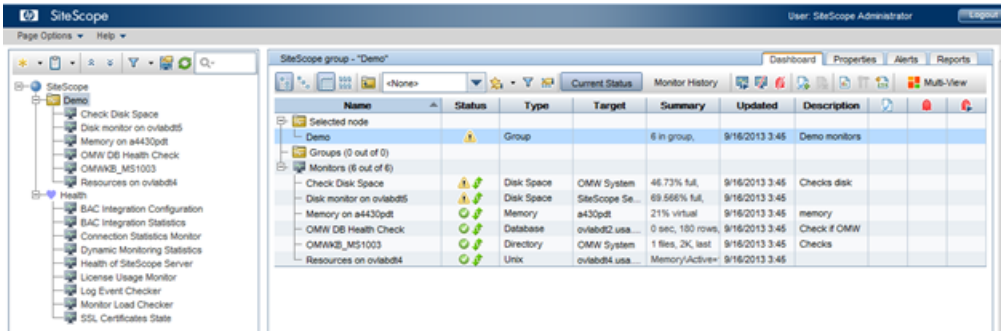
   For the **Tag Name** value, enter the name of your choice. For example, you might enter `opsa_tenant_sist`. Click the gold-colored star in the **Values**area, then enter a **Value Name** using the identical string that you used for the **Tag Name** value (`opsa_tenant_sist` for this example).

4. Click **OK** to save the tag definition.

Task 2: Using the New SiteScope Tag to Mark the Monitor or Monitor Groups

To use the tag you just created to mark the Monitor Groups, individual Monitors, or both, from which you want metrics sent to OpsA, do the following:

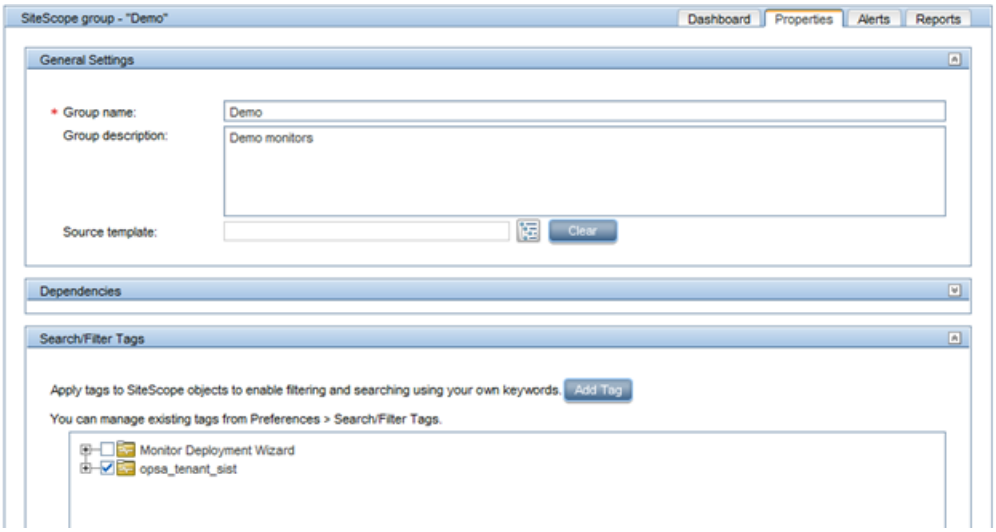
1. Navigate to the Monitors panel in SiteScope. This is normally the main screen you see when you first log on to SiteScope. The following shows an example system:



2. For each Monitor Group or individual Monitor from which you want metrics sent to OpsA, mark the Group or Monitor with the tag you created in "Task 1: Creating a SiteScope Tag" on page 289. For example, to mark the entire **Demo** Monitor Group (as in sending metrics from all of the monitors in the group), follow these steps:

   a. Select the **Monitor Group** name in the hierarchy list on the left of the screen.

   b. Click the **Properties** tab. For a Monitor Group, the following window opens:



   c. In the **Search/Filter Tags** configuration panel, select the checkbox for the tag you created in "Task 1: Creating a SiteScope Tag" on page 289.

   d. Click **Save** to save your changes to the Monitor Group configuration.

> **Note:** If you do not want to send metrics **from all of the monitors within a group**, you must mark each desired monitor individually. The steps are the same as :
>
> i. Select the **Monitor Group** name in the hierarchy list on the left of the screen.
>
> ii. Click the **Properties**tab and a window opens.

iii. Navigate to the **Search/Filter Tags** panel.

iv. Select the checkbox for the tag you created in "Task 1: Creating a SiteScope Tag" on page 289.

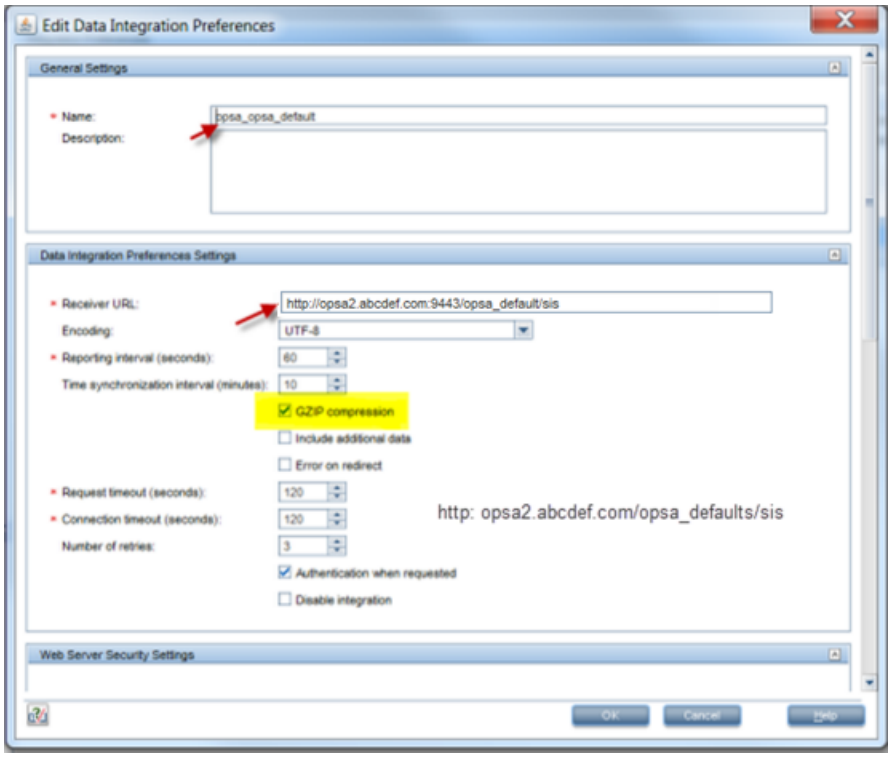
v. Click **Save** to save your changes to the Monitor Group configuration.

Task 3: Creating a New *Data Integration* Preference

In this final task, configure a new `Data Integration` preference that tells SiteScope where to send the marked data metrics:

1. From SiteScope, Navigate to **Preferences** > **Integration Preferences**.

2. Click the gold-colored star (the New Integration icon), then select the **Data Integration** link in the pop-up window. The following configuration window should appear:



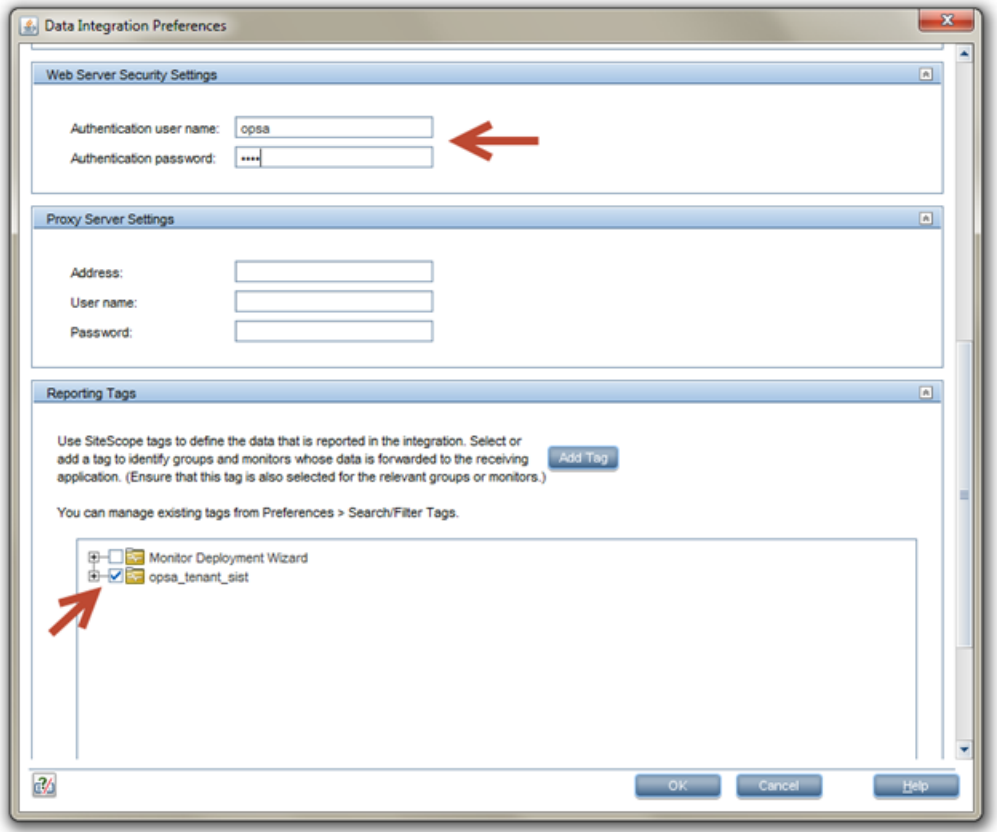
Provide a Name for this Data Integration, then provide the Receiver URL using the following format: `http://<fully-qualified domain name or ip address of the OpsA Collector>:9443/<tenant_name>/sis`

Select the **GZIP compression** option.

In this example, the target OpsA Collector is `opsa2.abcdef.com` and the target OpsA tenant is `opsa_default` (the default tenant). You do not need to change any other settings, as shown in the configuration window above.

3. Scroll down in the configuration window. In the **Web Server Security Settings** panel, authenticate using the credentials for the tenant being configured, which is `opsa` in this example.

> **Note:** These credentials are the same as those you would use to log on to the Operations Analytics console for a given tenant. For the example, the credentials are `opsa` (user name) and the associated password you set for this user during installation.



4. Finally, check the box for the tag that you created earlier in "Task 1: Creating a SiteScope Tag" on page 289. Selecting this tag is the most important setting, as it connects the previously marked **Monitor Groups** and **Monitors** to the Data Integration being configured.

5. Click **OK** to create the new SiteScope Data Integration.

After completing the configuration steps in this section, SiteScope begins forwarding data to the OpsA Collector Appliance based on the configuration choices you made.

Daylight Savings Time Codes

Use the following timezone attribute codes when setting a collection for Daylight Savings Time.

Note: See http://en.wikipedia.org/wiki/List_of_tz_database_time_zones for additional information about some of the time codes shown in the following table.

Supported Daylight Savings Time Codes

Code	Code
ACT	Africa/Freetown
AET	Africa/Gaborone
AGT	Africa/Harare
ART	Africa/Johannesburg
AST	Africa/Juba
Africa/Abidjan	Africa/Kampala
Africa/Accra	Africa/Khartoum
Africa/Addis_Ababa	Africa/Kigali
Africa/Algiers	Africa/Kinshasa
Africa/Asmara	Africa/Lagos
Africa/Asmera	Africa/Libreville
Africa/Bamako	Africa/Lome
Africa/Bangui	Africa/Luanda
Africa/Banjul	Africa/Lubumbashi
Africa/Bissau	Africa/Lusaka
Africa/Blantyre	Africa/Malabo
Africa/Brazzaville	Africa/Maputo
Africa/Bujumbura	Africa/Maseru
Africa/Cairo	Africa/Mbabane
Africa/Casablanca	Africa/Mogadishu
Africa/Ceuta	Africa/Monrovia

Supported Daylight Savings Time Codes, continued

Code	Code
Africa/Conakry	Africa/Nairobi
Africa/Dakar	Africa/Ndjamena
Africa/Dar_es_Salaam	Africa/Niamey
Africa/Djibouti	Africa/Nouakchott
Africa/Douala	Africa/Ouagadougou
Africa/El_Aaiun	Africa/Porto-Novo
Africa/Sao_Tome	America/Bahia
Africa/Timbuktu	America/Bahia_Banderas
Africa/Tripoli	America/Barbados
Africa/Tunis	America/Belem
Africa/Windhoek	America/Belize
America/Adak	America/Blanc-Sablon
America/Anchorage	America/Boa_Vista
America/Anguilla	America/Bogota
America/Antigua	America/Boise
America/Araguaina	America/Buenos_Aires
America/Argentina/Buenos_Aires	America/Cambridge_Bay
America/Argentina/Catamarca	America/Campo_Grande
America/Argentina/ComodRivadavia	America/Cancun
America/Argentina/Cordoba	America/Caracas
America/Argentina/Jujuy	America/Catamarca
America/Argentina/La_Rioja	America/Cayenne
America/Argentina/Mendoza	America/Cayman
America/Argentina/Rio_Gallegos	America/Chicago
America/Argentina/Salta	America/Chihuahua
America/Argentina/San_Juan	America/Coral_Harbour
America/Argentina/San_Luis	America/Cordoba

Supported Daylight Savings Time Codes, continued

Code	Code
America/Argentina/Tucuman	America/Costa_Rica
America/Argentina/Ushuaia	America/Creston
America/Aruba	America/Cuiaba
America/Asuncion	America/Curacao
America/Atikokan	America/Danmarkshavn
America/Atka	America/Dawson
America/Dawson_Creek	America/Indiana/Vevay
America/Denver	America/Indiana/Vincennes
America/Detroit	America/Indiana/Winamac
America/Dominica	America/Indianapolis
America/Edmonton	America/Inuvik
America/Eirunepe	America/Iqaluit
America/El_Salvador	America/Jamaica
America/Ensenada	America/Jujuy
America/Fort_Wayne	America/Juneau
America/Fortaleza	America/Kentucky/Louisville
America/Glace_Bay	America/Kentucky/Monticello
America/Godthab	America/Knox_IN
America/Goose_Bay	America/Kralendijk
America/Grand_Turk	America/La_Paz
America/Grenada	America/Lima
America/Guadeloupe	America/Los_Angeles
America/Guatemala	America/Louisville
America/Guayaquil	America/Lower_Princes
America/Guyana	America/Maceio
America/Halifax	America/Managua
America/Havana	America/Manaus

Supported Daylight Savings Time Codes, continued

Code	Code
America/Hermosillo	America/Marigot
America/Indiana/Indianapolis	America/Martinique
America/Indiana/Knox	America/Matamoros
America/Indiana/Marengo	America/Mazatlan
America/Indiana/Petersburg	America/Mendoza
America/Indiana/Tell_City	America/Menominee
America/Merida	America/Rainy_River
America/Metlakatla	America/Rankin_Inlet
America/Mexico_City	America/Recife
America/Miquelon	America/Regina
America/Moncton	America/Resolute
America/Monterrey	America/Rio_Branco
America/Montevideo	America/Rosario
America/Montreal	America/Santa_Isabel
America/Montserrat	America/Santarem
America/Nassau	America/Santiago
America/New_York	America/Santo_Domingo
America/Nipigon	America/Sao_Paulo
America/Nome	America/Scoresbysund
America/Noronha	America/Shiprock
America/North_Dakota/Beulah	America/Sitka
America/North_Dakota/Center	America/St_Barthelemy
America/North_Dakota/New_Salem	America/St_Johns
America/Ojinaga	America/St_Kitts
America/Panama	America/St_Lucia
America/Pangnirtung	America/St_Thomas
America/Paramaribo	America/St_Vincent

Supported Daylight Savings Time Codes, continued

Code	Code
America/Phoenix	America/Swift_Current
America/Port-au-Prince	America/Tegucigalpa
America/Port_of_Spain	America/Thule
America/Porto_Acre	America/Thunder_Bay
America/Porto_Velho	America/Tijuana
America/Puerto_Rico	America/Toronto
America/Tortola	Asia/Baghdad
America/Vancouver	Asia/Bahrain
America/Virgin	Asia/Baku
America/Whitehorse	Asia/Bangkok
America/Winnipeg	Asia/Beirut
America/Yakutat	Asia/Bishkek
America/Yellowknife	Asia/Brunei
Antarctica/Casey	Asia/Calcutta
Antarctica/Davis	Asia/Choibalsan
Antarctica/DumontDUrville	Asia/Chongqing
Antarctica/Macquarie	Asia/Chungking
Antarctica/Mawson	Asia/Colombo
Antarctica/McMurdo	Asia/Dacca
Antarctica/Palmer	Asia/Damascus
Antarctica/Rothera	Asia/Dhaka
Antarctica/South_Pole	Asia/Dili
Antarctica/Syowa	Asia/Dubai
Antarctica/Vostok	Asia/Dushanbe
Arctic/Longyearbyen	Asia/Gaza
Asia/Aden	Asia/Harbin
Asia/Almaty	Asia/Hebron

Supported Daylight Savings Time Codes, continued

Code	Code
Asia/Amman	Asia/Ho_Chi_Minh
Asia/Anadyr	Asia/Hong_Kong
Asia/Aqtau	Asia/Hovd
Asia/Aqtobe	Asia/Irkutsk
Asia/Ashgabat	Asia/Istanbul
Asia/Ashkhabad	Asia/Jakarta
Asia/Jayapura	Asia/Qatar
Asia/Jerusalem	Asia/Qyzylorda
Asia/Kabul	Asia/Rangoon
Asia/Kamchatka	Asia/Riyadh
Asia/Karachi	Asia/Riyadh87
Asia/Kashgar	Asia/Riyadh88
Asia/Kathmandu	Asia/Riyadh89
Asia/Katmandu	Asia/Saigon
Asia/Kolkata	Asia/Sakhalin
Asia/Krasnoyarsk	Asia/Samarkand
Asia/Kuala_Lumpur	Asia/Seoul
Asia/Kuching	Asia/Shanghai
Asia/Kuwait	Asia/Singapore
Asia/Macao	Asia/Taipei
Asia/Macau	Asia/Tashkent
Asia/Magadan	Asia/Tbilisi
Asia/Makassar	Asia/Tehran
Asia/Manila	Asia/Tel_Aviv
Asia/Muscat	Asia/Thimbu
Asia/Nicosia	Asia/Thimphu
Asia/Novokuznetsk	Asia/Tokyo

Supported Daylight Savings Time Codes, continued

Code	Code
Asia/Novosibirsk	Asia/Ujung_Pandang
Asia/Omsk	Asia/Ulaanbaatar
Asia/Oral	Asia/Ulan_Bator
Asia/Phnom_Penh	Asia/Urumqi
Asia/Pontianak	Asia/Vientiane
Asia/Pyongyang	Asia/Vladivostok
Asia/Yakutsk	Australia/Melbourne
Asia/Yekaterinburg	Australia/NSW
Asia/Yerevan	Australia/North
Atlantic/Azores	Australia/Perth
Atlantic/Bermuda	Australia/Queensland
Atlantic/Canary	Australia/South
Atlantic/Cape_Verde	Australia/Sydney
Atlantic/Faeroe	Australia/Tasmania
Atlantic/Faroe	Australia/Victoria
Atlantic/Jan_Mayen	Australia/West
Atlantic/Madeira	Australia/Yancowinna
Atlantic/Reykjavik	BET
Atlantic/South_Georgia	BST
Atlantic/St_Helena	Brazil/Acre
Atlantic/Stanley	Brazil/DeNoronha
Australia/ACT	Brazil/East
Australia/Adelaide	Brazil/West
Australia/Brisbane	CAT
Australia/Broken_Hill	CET
Australia/Canberra	CNT
Australia/Currie	CST

Supported Daylight Savings Time Codes, continued

Code	Code
Australia/Darwin	CST6CDT
Australia/Eucla	CTT
Australia/Hobart	Canada/Atlantic
Australia/LHI	Canada/Central
Australia/Lindeman	Canada/East-Saskatchewan
Australia/Lord_Howe	Canada/Eastern
Canada/Mountain	Europe/Berlin
Canada/Newfoundland	Europe/Bratislava
Canada/Pacific	Europe/Brussels
Canada/Saskatchewan	Europe/Bucharest
Canada/Yukon	Europe/Budapest
Chile/Continental	Europe/Chisinau
Chile/EasterIsland	Europe/Copenhagen
Cuba	Europe/Dublin
EAT	Europe/Gibraltar
ECT	Europe/Guernsey
EET	Europe/Helsinki
EST	Europe/Isle_of_Man
EST5EDT	Europe/Istanbul
Egypt	Europe/Jersey
Eire	Europe/Kaliningrad
Etc/GMT	Europe/Kiev
Etc/GMT0	Europe/Lisbon
Etc/Greenwich	Europe/Ljubljana
Etc/UCT	Europe/London
Etc/UTC	Europe/Luxembourg
Etc/Universal	Europe/Madrid

Supported Daylight Savings Time Codes, continued

Code	Code
Etc/Zulu	Europe/Malta
Europe/Amsterdam	Europe/Mariehamn
Europe/Andorra	Europe/Minsk
Europe/Athens	Europe/Monaco
Europe/Belfast	Europe/Moscow
Europe/Belgrade	Europe/Nicosia
Europe/Oslo	GB-Eire
Europe/Paris	GM
Europe/Podgorica	GMT0
Europe/Prague	Greenwich
Europe/Riga	HST
Europe/Rome	Hongkong
Europe/Samara	IET
Europe/San_Marino	IST
Europe/Sarajevo	Iceland
Europe/Simferopol	Indian/Antananarivo
Europe/Skopje	Indian/Chagos
Europe/Sofia	Indian/Christmas
Europe/Stockholm	Indian/Cocos
Europe/Tallinn	Indian/Comoro
Europe/Tirane	Indian/Kerguelen
Europe/Tiraspol	Indian/Mahe
Europe/Uzhgorod	Indian/Maldives
Europe/Vaduz	Indian/Mauritius
Europe/Vatican	Indian/Mayotte
Europe/Vienna	Indian/Reunion
Europe/Vilnius	Iran

Supported Daylight Savings Time Codes, continued

Code	Code
Europe/Volgograd	Israel
Europe/Warsaw	JST
Europe/Zagreb	Jamaica
Europe/Zaporozhye	Japan
Europe/Zurich	Kwajalein
GB	Libya
MET	Pacific/Enderbury
MIT	Pacific/Fakaofo
MST	Pacific/Fiji
MST7MDT	Pacific/Funafuti
Mexico/BajaNorte	Pacific/Galapagos
Mexico/BajaSur	Pacific/Gambier
Mexico/General	Pacific/Guadalcanal
Mideast/Riyadh87	Pacific/Guam
Mideast/Riyadh88	Pacific/Honolulu
Mideast/Riyadh89	Pacific/Johnston
NET	Pacific/Kiritimati
NST	Pacific/Kosrae
NZ	Pacific/Kwajalein
NZ-CHAT	Pacific/Majuro
Navajo	Pacific/Marquesas
PLT	Pacific/Midway
PNT	Pacific/Nauru
PRC	Pacific/Niue
PRT	Pacific/Norfolk
PST	Pacific/Noumea
PST8PDT	Pacific/Pago_Pago

Supported Daylight Savings Time Codes, continued

Code	Code
Pacific/Apia	Pacific/Palau
Pacific/Auckland	Pacific/Pitcairn
Pacific/Chatham	Pacific/Pohnpei
Pacific/Chuuk	Pacific/Ponape
Pacific/Easter	Pacific/Port_Moresby
Pacific/Efate	Pacific/Rarotonga
Pacific/Saipan	Turkey
Pacific/Samoa	UCT
Pacific/Tahiti	US/Alaska
Pacific/Tarawa	US/Aleutian
Pacific/Tongatapu	US/Arizona
Pacific/Truk	US/Central
Pacific/Wake	US/East-Indiana
Pacific/Wallis	US/Eastern
Pacific/Yap	US/Hawaii
Poland	US/Indiana-Starke
Portugal	US/Michigan
ROK	US/Mountain
SST	US/Pacific
Singapore	US/Pacific-New
SystemV/AST4	US/Samoa
SystemV/AST4ADT	UTC
SystemV/CST6	Universal
SystemV/CST6CDT	VST
SystemV/EST5	W-SU
SystemV/EST5EDT	WET
SystemV/HST10	Zulu

Supported Daylight Savings Time Codes, continued

Code	Code
SystemV/MST7	
SystemV/MST7MDT	
SystemV/PST8	
SystemV/PST8PDT	
SystemV/YST9	
SystemV/YST9YDT	

Send Documentation Feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Operations Analytics Configuration Guide (Operations Analytics 2.20)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to sw-doc@hp.com.

We appreciate your feedback!