



# Operations Bridge Analytics

Software Version: 3.03

## Administration Guide

Document Release Date: December 2017

Software Release Date: November 2017



**Hewlett Packard**  
Enterprise

## Legal Notices

### Warranty

The only warranties for Hewlett Packard Enterprise 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. Hewlett Packard Enterprise 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 Hewlett Packard Enterprise 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

© 2016 - 2017 Hewlett Packard Enterprise Development LP

### Trademark Notices

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

AMD, the AMD Arrow symbol and ATI are trademarks of Advanced Micro Devices, Inc.

Citrix® and XenDesktop® are registered trademarks of Citrix Systems, Inc. and/or one more of its subsidiaries, and may be registered in the United States Patent and Trademark Office and in other countries.

Google™ and Google Maps™ are trademarks of Google Inc.

Intel®, Itanium®, Pentium®, and Intel® Xeon® are trademarks of Intel Corporation in the U.S. and other countries.

iPad® and iPhone® are trademarks of Apple Inc.

Java is a registered trademark of Oracle and/or its affiliates.

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

Microsoft®, Windows®, Lync®, Windows NT®, Windows® XP, Windows Vista® and Windows Server® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

NVIDIA® is a trademark and/or registered trademark of NVIDIA Corporation in the U.S. and other countries.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

Red Hat® is a registered trademark of Red Hat, Inc. in the United States and other countries.

SAP® is the trademark or registered trademark of SAP SE in Germany and in several other countries.

UNIX® is a registered trademark of The Open Group.

## Documentation Updates

To check for recent updates or to verify that you are using the most recent edition of a document, go to: <https://softwaresupport.hpe.com/>.

This site requires that you register for an HP Passport and to sign in. To register for an HP Passport ID, click **Register** on the HPE Software Support site or click **Create an Account** on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HPE sales representative for details.

## Support

Visit the HPE Software Support site at: <https://softwaresupport.hpe.com/>.

Most of the support areas require that you register as an HP Passport user and to sign in. Many also require a support contract. To register for an HP Passport ID, click **Register** on the HPE Support site or click **Create an Account** on the HP Passport login page.

To find more information about access levels, go to: <https://softwaresupport.hpe.com/web/softwaresupport/access-levels>.

**HPE Software Solutions Now** accesses the HPSW Solution and Integration Portal website. This site enables you to explore HPE Product Solutions to meet your business needs, includes a full list of Integrations between HPE Products, as well as a listing of ITIL Processes. The URL for this website is <https://softwaresupport.hpe.com/km/KM01702731>.

# Contents

|   |    |
|---|----|
| Maintenance .....   | 6  |
| Maintain the database .....   | 7  |
| Decommission collectors and perform disaster recovery .....   | 12 |
| Restart processes .....   | 14 |
| Tune Apache Kafka disk partition capacity .....   | 16 |
| Rebalancing the Kafka cluster .....   | 17 |
| Manage collected data file usage .....  | 19 |
| Modify unit scaling on collected data .....   | 21 |
| Content Packs .....   | 22 |
| Change the audit logging level .....  | 23 |
| Add more application servers .....  | 25 |
| Monitor processes .....   | 26 |
| Check the status of your servers and operations .....   | 27 |
| Improve performance .....   | 31 |
| Improve log analytics collection performance .....  | 31 |
| Configuring Logger to Forward CEF Messages to Operations Analytics .....  | 32 |
| Increase JVM memory to improve collection performance .....   | 34 |
| Increase the entity index size .....  | 35 |
| Manage users and tenants .....  | 36 |
| Register collector hosts .....  | 48 |
| Resolve host aliases .....  | 50 |
| Set up security .....   | 53 |
| Encrypting Operations Bridge Analytics .....  | 53 |
| Other security considerations .....   | 54 |
| SSL for OBA components .....  | 55 |
| Configuring SSL for the OBA application servers and collectors .....  | 55 |
| This section includes the following: .....  | 56 |
| Configuring SSL communication to the OBA application server<br>and collector with a Certificate Authority (CA) signed certificate ..... | 56 |
| Import a certificate created and signed by an external CA .....   | 63 |

|  |     |
|--|-----|
| Configuring SSL communication to the OBA application server and collector with a self-signed certificate ..... | 64  |
| Editing the SSL configuration for the OBA application server or collector .....                                | 71  |
| Disabling the SSL configuration for the OBA application server or collector .....                              | 73  |
| Managing the keystore and truststore for the OBA application server and collector .....                        | 75  |
| Two-Way SSL for accessing ArcSight Logger .....  | 78  |
| SSL for communication between Vertica and OBA .....  | 80  |
| Enabling SSL Communications between the OBA Application Server and Vertica .....                               | 81  |
| Disabling SSL Communications between the OBA Application Server and Vertica .....                              | 85  |
| Enabling SSL Communications between the Operations Bridge Analytics Collector Host and Vertica .....           | 86  |
| Disabling SSL Communications between the Operations Bridge Analytics Collector Host and Vertica .....          | 86  |
| Adjusting Operations Bridge Analytics for RC4 Cipher Security Changes .....                                    | 87  |
| SSL for the SMTP Server used for OBA Alerts .....  | 88  |
| HTTP and HTTPS .....   | 91  |
| Configure the HTTP and HTTPS port for the OBA collector host .....   | 91  |
| Configure the HTTP and HTTPS user name and password for the OBA collector host .....                           | 92  |
| Single Sign-On .....   | 93  |
| Configure and enable single sign-on to access OBA .....  | 93  |
| Disable SSO access to OBA .....  | 95  |
| Configure LDAP authentication .....  | 97  |
| PKI .....  | 102 |
| Configure user authentication using PKI to access OBA .....  | 102 |
| Disable user authentication using PKI to access OBA .....  | 105 |
| Edit user authentication using PKI to access OBA .....   | 105 |
| Resetting user passwords .....   | 106 |
| Change the password of a collector host .....  | 107 |
| Changing the port used by the OBA console .....  | 107 |

|                                   |     |
|-----------------------------------|-----|
| Send documentation feedback ..... | 109 |
|-----------------------------------|-----|

# Maintenance

This section provides information about maintaining your OBA deployment.

## Maintain the database

Use the instructions in this section to maintain the Operations Bridge Analytics databases. You can monitor the size of the Vertica database, reset the database password, and modify the collection retention periods.

## Learn more

### Backing up and restoring data

To back up or restore data for the OBA Application Server and Collector hosts, see the *Backing up and Restoring the Database* section of the [Vertica Administrator's Guide](#).

## Tasks

### How to monitor the size of the database

By default, the Operations Bridge Analytics 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 Operations Bridge Analytics database.

To check or verify the size of the Operations Bridge Analytics 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 ["How to modify the collection retention periods" on page 10](#) 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.

**Note:** Over time you might find that Operations Bridge Analytics collection data approaches or exceeds the storage space you configured in Vertica. To remedy this storage space issue, use the procedure shown in [Moving Data Storage Locations](#) to create a new storage location for select collection (tables) and move its existing content to the new location that you create.

#### How to reset the database password

If you must change the Vertica dbadmin password, use the instructions in this section. The steps shown in this section assume that Operations Bridge Analytics is already running.

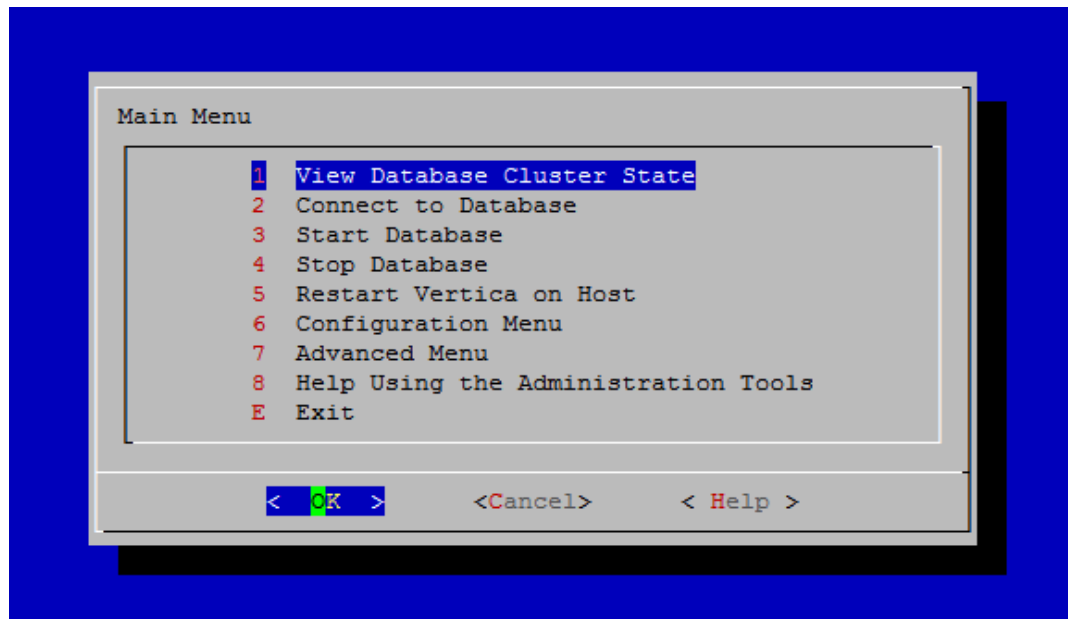
1. The Vertica database admin user is dbadmin and its default password is dbadmin. Do the following to change the password:
  - a. Run the following command to log on to the opsadb database using the vsql tool:
 

```
/opt/vertica/bin/vsql -h hostname -p 5433 -U dbadmin -w dbadmin -d opsadb
```

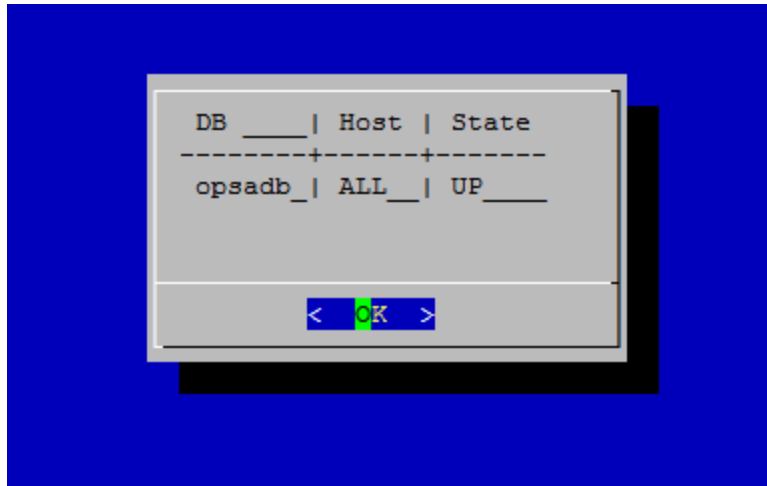
**Note:** opsadb is the Vertica database created during the Vertica installation.



- b. Run the following command to change the password:  
`alter user dbadmin identified by '<new password>';`
    - c. Enter `\q` to quit the `vsq1` tool.
  2. Run the `postinstall` script on the OBA Application Server and Collector host:
    - a. Log on to the OBA Application Server and run the `opsa-server-postinstall.sh` script using the `scaleout` flag: `opsa-server-postinstall.sh -scaleout`. After you run this script, you will be asked to provide the database connect strings. During this step, provide the new password of the Vertica `dbadmin` user. Here is an example of the command sequence to use:
      - i. `[opsa@ACEVM145563026 opsa]$ cd /opt/HP/opsa/bin/`
      - ii. `[opsa@ACEVM145563026 bin]$ ./opsa-server-postinstall.sh -scaleout`
    - b. Log on to the to each OBA Collector host and run the following script: `opsa-collector-postinstall.sh`
  3. Do the following to check the database:
    - a. Run the `su - dbadmin` command.
    - b. Run the `/opt/vertica/bin/adminTools` command. You should see a screen similar to the following:



- c. The opsadb database should have been created when you first installed Operations Bridge Analytics. Enter 1 to view the state of the database; then click **OK**. You should see a screen similar to the following if the opsadb database is running:



- d. Click **OK** twice to exit the adminTools interactive command.

**Note:** If you must stop or restart the database, you can always do it from the first screen shown in this step. You can also (carefully) complete other administrative operations using this tool.

#### How to modify the collection retention periods

By default, Operations Bridge Analytics'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 Bridge Analytics 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 man page) 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, Operations Bridge Analytics removes any data record with a time stamp older than the listed retention period for those collections.

# Decommission collectors and perform disaster recovery

To troubleshoot your Kafka cluster, you can decommission collectors or perform disaster recovery. Both might be necessary if one or more of the collectors in your Kafka cluster become unavailable. For details, see [Collectors are currently not reachable](#).

## How to decommission collectors

Collectors can be decommissioned regardless of whether they are up and running, or down. The process remains the same.

1. Stop all collections that are running on the collector, either using the GUI or the CLI `opsa-collection-config.sh`. For details, see [Deleting Source Types](#).
2. Unregister the collector from the tenant:

```
opsa-collection-config.sh -unregister -collectorhost <collector> -username  
<tenantadmin>
```

3. Remove the collector permanently from the cluster:

```
opsa-cluster-manager.sh -r <collector_host>
```

If the OBA processes are still running on the collector host, a notification will be displayed when they should be shut down.

## How to reregister decommissioned collectors

After you decommissioned a collector, it is possible to register the collector again.

1. Delete everything in the folder `/opt/HP/opsa/data/kafka`
2. Execute the following command to register the collector again:

```
opsa-collector-postinstall.sh
```

## How to perform disaster recovery

A disaster occurs when more than one collector is down and cannot be recovered. In that case, all other

collectors are compromised as well.

To recover from this, you must decommission all collectors and set up new ones.

1. Stop all collections that are running on the registered collectors, either using the GUI or the CLI `opsa-collection-config.sh`. For details, see [Deleting Source Types](#).

2. Unregister all collectors from their tenants:

```
opsa-collection-config.sh -unregister -collectorhost <collector> -username  
<tenantadmin>
```

3. Remove the collectors permanently from the cluster:

```
opsa-cluster-manager.sh -r <collector_host>
```

4. If there are some collectors that are still operational, log on to the machine's file system and delete everything in the folder `/opt/HP/opsa/data/kafka`.

5. Execute the following command:

```
opsa-collector-postinstall.sh
```

6. *Optional*. Set up new collectors as replacement for the old ones.

7. Register all collectors by using `opsa-collection-config.sh`. For details, see ["Register collector hosts" on page 48](#).

# Restart processes

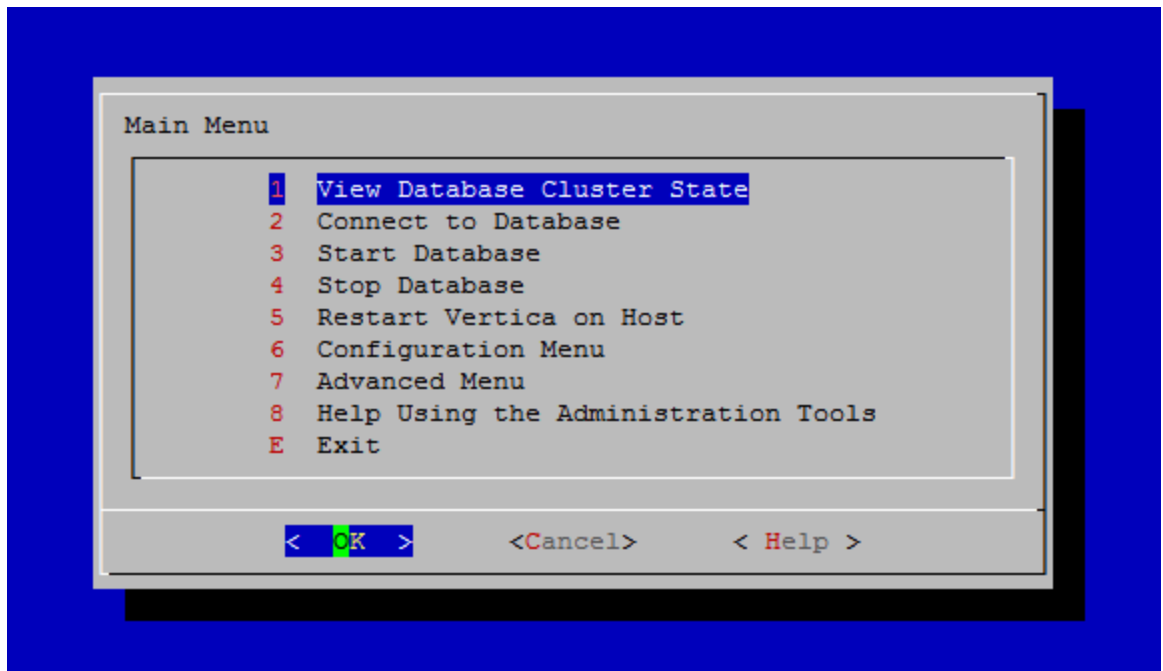
There are times when Operations Bridge Analytics might abruptly shut down, as in during a power outage, network issue, or other unintended shutdown. For the OBA processes to function correctly, the Vertica database must completely start up before restarting the OBA processes. If the Vertica database is not available when the OBA processes start up, these processes might not function correctly.

## Tasks

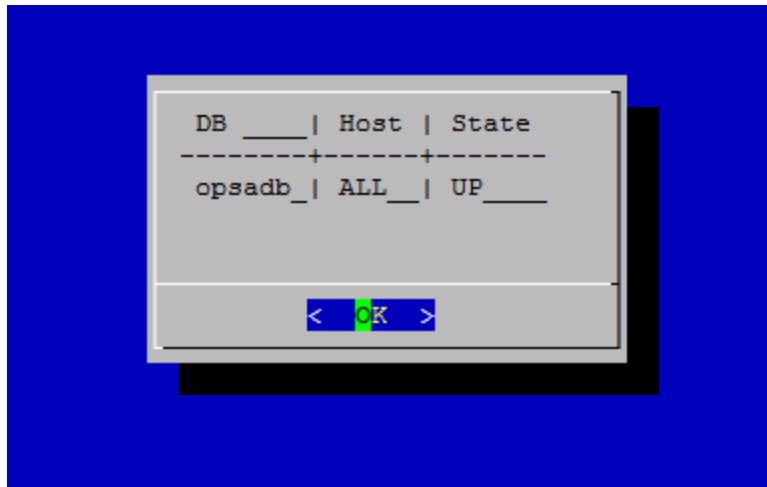
How to restart the OBA processes

To make sure the OBA processes start up correctly, do the following:

1. Check the database on the Vertica server:
  - a. Run the `su - dbadmin` command.
  - b. Run the `/opt/vertica/bin/adminTools` command. You should see a screen similar to the following:



- c. Enter 1 to view the state of the database; then click **OK**. You should see a screen similar to the following if the opsadb database is running:



- d. Click **OK** twice to exit the adminTools interactive command.
- e. If the database is not up, wait a few minutes, then rerun the previous steps to recheck the database.

**Note:** Do not start the Operations Bridge Analytics processes until the Vertica database is running.

2. Run the `opsa status` command on the OBA Application Servers and all of the OBA Collector hosts. For each server that does not have processes running, run the `opsa start` command.
3. After five minutes, check to see that you can open the Operations Bridge Analytics console.

#### How to restart the OBA application server and OBA collector host

If you suspect that Vertica has stopped functioning, you can restart the Operations Bridge Analytics services on the OBA Application Server and Collector hosts. The symptom you might see is that new data is no longer being collected with old data still available for viewing.

To restart the OBA Application Server and Collector hosts, do the following on each server: `$OPSA_HOME/bin/opsa restart`

See the *opsa* reference page (or the Linux man page) for more information.

## Tune Apache Kafka disk partition capacity

Apache Kafka is the messaging system that Operations Bridge Analytics uses for its collection processes. As you add more collections to Operations Bridge Analytics, the utilization of the Kafka repository at `/opt/HP/opsa/data/kafka` increases. If the `/opt/HP/opsa/data/kafka` partition approaches 100% utilization you must adjust the number of days that data is being retained. An optimal value of data to retain should be 80% or less of the disk capacity.

To define the retention policy so that disk capacity doesn't exceed 80%, do the following:

1. Edit the `/opt/HP/opsa/conf/deployment/opsa-deployment.xml` file.
2. Complete only one of the following actions to set the retention policy to meet your needs:
  - **Base the retention policy on time:** Set the `log.retention.hours` parameter to a value less than the default of 168. This value is in hours.
  - **Base the retention policy on size:** Set the `log.retention.bytes` and `log.segment.bytes` parameters to identical values. These values reflect the size per topic, and the default value is 2 GB per topic.
3. Save your changes.
4. Run the following commands to restart the Kafka process and commit your changes:
  - a. `/opt/HP/opsa/scripts/opsa-deployment-manager.sh`
  - b. `/opt/HP/opsa/scripts/opsa-deployment-loader.sh`
  - c. `/opt/HP/opsa/bin/opsa-kafka restart`

Continue to monitor the `/opt/HP/opsa/data/kafka` partition and make additional changes if it begins to exceed 80% disk capacity.



## Rebalancing the Kafka cluster

Apache Kafka is the messaging system that Operations Bridge Analytics uses for its collection processes. Your registered collectors form a Kafka cluster. When collectors are added or removed from OBA, the Kafka cluster may become unbalanced. This means that data is unevenly distributed between the collectors. This situation is not critical, as the Kafka cluster still works and there is no direct risk of data loss. However, you still need to rebalance the cluster so that the load is evenly distributed.

To rebalance the Kafka cluster, use the tool `opsa-cluster-manager.sh` as described below.

**Note:** After you complete the appropriate action as shown below and refresh the Operations Bridge Analytics console (or restart Operations Bridge Analytics), the notification is cleared.

### How to rebalance the Kafka cluster

**Caution:** Do not rebalance the Kafka cluster if one of the collectors is down. For details, see [Collectors are currently not reachable](#).

Rebalancing the Kafka cluster will result in a high network load between the collectors, as the data needs to be transferred between them. The collectors and the Kafka cluster are operational during the rebalancing, but it can happen that overall performance is decreased due to the additional overhead.

Run the `opsa-cluster-manager.sh` tool to rebalance the cluster: `opsa-cluster-manager.sh -b`

A warning is displayed about the increased network traffic between the collectors.

For a list of all options that can be used with `opsa-cluster-manager.sh`, see ["opsa-cluster-manager.sh usage" below](#).

### opsa-cluster-manager.sh usage

The `opsa-cluster-manager.sh` tool can be used as follows:

```
opsa-cluster-manager.sh [-b | -h | -k | -l | -r <host>]
```

```
-b, --      Balances the Kafka cluster.  
balance
```

- h, --help Displays help about the usage.
- k, --kafka Shows the status of the Kafka cluster: the replication factor, the number of replicas per cluster node, if the cluster is operational, or if the cluster must be rebalanced.
- l, --list Lists the OBA collectors that are part of the cluster deployment.
- r, --remove Permanently removes the specified host from the OBA cluster deployment.  
<host>

# Manage collected data file usage

OBA Collector hosts store collected data on their file systems. Each OBA Collector host 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 OBA Collector host.

To configure the parameters associated with these delete policies, do the following from each OBA Collector host you want to control:

1. Edit the `/opt/HP/opsa/conf/opsa-collector.properties` file.
2. Using the 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 = 15`  
`#com.hp.opsa.collector.file.garbage.diskfreepct.start = 30`  
`#com.hp.opsa.collector.file.garbage.diskfreepct.stop = 50`  
`#com.hp.opsa.collector.file.garbage.max.daysold = 5`  
`#com.hp.opsa.collector.file.garbage.enabled = true`
3. Save the file.
4. Run the following command from the OBA Collector host

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

Now the collected data on the OBA Collector hosts 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 OBA Collector host 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 OBA Collector host is configured as shown 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                     |

# Modify 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 by using the following options:

`%,mbps,kbps,gbps,kb,mb,gb,hz,khz,mhz,ghz,bytes,BIT,PB,EB,W,V,A,secs,millisecs,ms,page  
s/sec,per  
second,switches/sec,bytes/sec,KB/sec,interrupts/sec,packets/sec,errors/sec,reads/sec,bps,pe  
r hour,per min`

then specify the factor by which you want to multiply the incoming data.

**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>
```

## Content Packs

You can combine additional information with the data collected by Operations Bridge Analytics by using content packs. You can browse and download content pack at [ITOM Marketplace](#). We recommend that you regularly check this link for new content packs, as new ones are frequently released.

## Change the audit logging level

Operations Bridge Analytics provides audit log files to audit events associated with account and application activity. This audit activity does not include any information that might be considered sensitive in nature. Operations Bridge Analytics 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 Operations Bridge Analytics 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 OBA Application Server reside in the following location:

```
$OPSA_HOME/log/audit/opsa-server-audit.log
```

There are several logging levels supported by the Operations Bridge Analytics 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 OBA Application Server, edit the following file and follow the file's instructions:

```
$OPSA_HOME/jboss/standalone/configuration/standalone.xml
```

**Tip:** Back up the `standalone.xml` file before saving your changes.

For example, to turn off logging, do the following:

1. Open the `standalone.xml` file on the OBA Application Server.
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.example.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`

**Note:** After running an `opsa-server restart` command, information logged in the `/opt/HP/opsa/log/opsa-server.log` file is removed. To view the files that retain this information across restarts, see the log files located in the `/opt/HP/opsa/jboss/standalone/log` directory. Look for log file names that begin with `server.log`.



# Add more application servers

As your Operations Bridge Analytics environment expands, you might need to add more OBA application servers. OBA supports a maximum of three servers. To add another application server, do the following:

1. Install a new OBA Application Server as shown in the *OBA Installation Guide*.
2. Run the `$OPSA_HOME/bin/opsa-server-postinstall.sh -scaleout` command to add the new OBA Application Server. For more information, see the *opsa-server-postinstall.sh* reference page (or the Linux man page).

**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 OBA Application Server.

3. Reboot all of the OBA application servers.
4. After all of the servers finish rebooting, you must reboot all of the OBA Collector hosts so they can identify the newly added server.
5. An Alerts Collection gets created each time an OBA Collector host is registered to an OBA Application Server. When adding OBA Application Servers, do the following:

Run the following command from each newly added OBA Application Server 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>
```

# Monitor processes

Operations Bridge Analytics provides the `opsa` script to check status or control OBA services. See the `opsa` reference page (or the Linux man page) for more information.

OBA depends on several different services to be active on the deployed application server and collector hosts. These services start automatically when booting up your hosts (you do not need to specifically configure these services).

You can use the `opsa` script to do several things:

- Run the `$OPSA_HOME/bin/opsa status` command script to check the status of all of these services at once.
- When necessary, you can control all of the OBA-related services on any application server and collector hosts (for example, in preparation for installing a software patch):
  - Start Operations Bridge Analytics services: `$OPSA_HOME/bin/opsa start`
  - Stop Operations Bridge Analytics services: `$OPSA_HOME/bin/opsa stop`

**Note:** A network disruption can cause Operations Bridge Analytics services to stop functioning. If you suspect that the OBA Application Server and Collector hosts lost connectivity to the network, you might restart them using `$OPSA_HOME/bin/opsa restart` command.

Operations Bridge Analytics also provides the `opsa-process-manager.sh` script to stop and start the OBA Process Manager service on a single OBA application server or collector host. You can also use the `opsa-process-manager.sh` script to monitor Operations Bridge Analytics processes. See the `opsa-process-manager.sh` reference page (or the Linux man page) for more information.

It is common for users to manually manage all the Operations Bridge Analytics processes together by using the `opsa` script. It is also possible for users to stop and start individual components using commands such as `opsa-server` or `opsa-collector`. The `opsa-process-manager.sh` script recognizes the processes that users manually stop and does not attempt to restart these processes.

**Note:** A network disruption can cause this process management feature to stop functioning. If you suspect that the OBA Application Server and Collector hosts lost connectivity to the network, restart them as detailed in ["How to restart the OBA application server and OBA collector host" on page 15](#) after the network connectivity is restored.

# Check the status of your servers and operations

You can check the status of your Operations Bridge Analytics servers and operations to help you assess the health of your OBA environment. To check the status of your servers, you can use CLI commands or the Health Overview dashboard.

## Learn more

### About the Health Overview dashboard

The Health Overview dashboard contains predefined panels to help you assess the health of your Operations Bridge Analytics servers and integrations. The tops panels in the display show Operations Agent performance data from hosts that are in the Operations Bridge Analytics Service Topology definition. Nodes that are configured as part of the Operations Bridge Analytics Service Topology include your OBA Application Servers, OBA Collector hosts, and Vertica database hosts.

The **Host System Metrics Over Time** pane shows metrics such as `cpu` and `peak disk utilization` from Operations Agent data being collected on hosts in the Operations Bridge Analytics Service Topology.

The **Service Topology** pane shows a pie-chart view of performance metrics from Operations Agent for the OBA Application Server and Collector hosts.

The **Row Count of Collected Metrics and Log** pane is useful for confirming that the collections are running as expected. As you configure additional Operations Bridge Analytics collections, the number of collections shown in this **Row Count of Collected Metrics and Log** pane increases.

For usability, augment the color coding by selecting the **Show Values** checkbox on the right. Hover over the left side collection labels to bring up a screen tip showing the full name of each collection. Without any configuration applied, you will see the following entries: "log\_group\_0\_metrics", "log\_group\_1\_metrics", and "log\_group\_2\_metrics". These entries are automatically generated collections related to the log file tracking facility.

An "opsa\_collection\_alerts" entry in the table tracks triggered alerts seen over time. If you configure Operations Agent collections, you will see "OA\_sysperf\_global" values coming in every 15 minutes, adding three rows for each Operations Agent node from which you collect data. Each additional collection adds more lines to this health display, although it may take up to 15 minutes after you configure a new collection for data to show up in this dashboard.

**Note:** A SiteScope collection may add up to 50 rows to this panel, which can make it more complicated to navigate. To make this easier, use the **Resize Pane > Increase Height** function in the upper right of the **Row Count of Collected Metrics and Log** pane to increase the pane height. Doing so reduces the number of pages you must navigate through using the page control on the lower right.

The next pane in the dashboard, **Configured Collections Dictionary**, shows a table of information that includes collection property information for each OBA Collector host.

The last pane, **Log Messages(100+)**, shows the results from the self-monitoring feature. It contains all log information from the OBA Application Server and Collector hosts that are running self-monitoring.

## Tasks

How to configure OBA to monitor its own health

To configure Operations Bridge Analytics to monitor its own active components, do the following:

1. Make sure the Operations Bridge Analytics software is installed and configured as shown in the *Operations Bridge Analytics Installation Guide*.

2. Edit the `/etc/yum.conf` file and add the proxy information for your network.

Your 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 the file.

3. Install Operations Agent on the Vertica database server using the information shown in the [Operations Agent and Infrastructure SPIs Installation Guide](#).
4. Configure the syslogs from the Vertica database server, the OBA Collector host, and the OBA Application Server to forward to the Operations Analytics Data Pipe server by appending `"*. * @@<logger_hostname>:515"` to the `/etc/rsyslog.conf` file.)
5. Run the following command to restart the `rsyslog` service:
 

```
service rsyslog restart
```

## How to check the OBA status using the CLI

The table below describes the commands used to check the status of Operations Bridge Analytics:

| Command  | Description   |
|--|---|
| <code>\$OPSA_HOME/bin/opsa-collection-config.sh -list -collectorhost &lt;collector hostname&gt; -username opsatenantadmin</code> | <i>Run on the OBA application server.</i> Lists the collections deployed to that OBA Collector host.          |
| <code>opsa-server status</code>  | <i>Run on the OBA application server.</i> Checks the status of the Operations Bridge Analytics service.       |
| <code>opsa-collector status</code>   | <i>Run on the Collector Appliance.</i> Checks the status of the collector service on the Collector Appliance. |
| <code>opsa-loader status</code>  | <i>Run on the Collector Appliance.</i> Checks the status of the loader service on the Collector Appliance.    |

## How to check the OBA status using the Health Overview dashboard

Use the Health Overview dashboard to investigate the health of the Operations Bridge Analytics servers. The table below describes the query panes available.

**Note:** If you view the message that no data is available, this might mean you do not have the required software to collect the expected data. See the **Required Software** column of the table below.

| Query Pane                    | Description  | Required Software |
|-------------------------------|--|-------------------|
| Host System Metrics over Time | Use this visualization to determine server health for the OBA application servers.<br><br>Shows the average value over time for the following metrics for each server running the Operations Bridge Analytics service: <ul style="list-style-type: none"> <li>• System up time</li> <li>• CPU utilization</li> </ul> | Operations Agent  |
| Service Topology              | Use this visualization to determine the servers running Operations Bridge Analytics software.  | Operations Bridge |

| Query Pane                        | Description  | Required Software                |
|-----------------------------------|--|----------------------------------|
|                                   | <p>Shows topology information for the Operations Bridge Analytics service, including the following servers:</p> <ul style="list-style-type: none"> <li>• Operations Bridge Analytics application servers</li> <li>• Operations Bridge Analytics collector servers</li> <li>• Operations Analytics Data Pipe servers</li> </ul> <p>Also shows the CPU utilization and system up time for each of the Operations Bridge Analytics servers.</p> | Analytics only                   |
| Collected Metric - Row Counts     | Shows a row for the data 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+)               | <p>Use this visualization to troubleshoot any Operations Bridge Analytics log file error messages.</p> <p>Shows all log file messages for servers running the Operations Bridge Analytics service.</p> <p>Shows the results from the self-monitoring feature. It contains all log information from the OBA Application Server and Collector hosts that are running self-monitoring.</p>  | Operations Bridge Analytics only |

# Improve performance

This section provides information about improving the performance of your OBA system. It includes the following topics:

- ["Improve log analytics collection performance" below](#)
- ["Configuring Logger to Forward CEF Messages to Operations Analytics" on the next page](#)
- ["Increase JVM memory to improve collection performance" on page 34](#)

## Improve log analytics collection performance

To improve the performance of the Log Analytics Collection, you can choose to configure dedicated OBA Collector hosts for the data streaming component (Apache Storm).

To do this, disable Apache Storm on the OBA Collector hosts that have existing Log Analytics Collections configured.

1. On each OBA Collector host that has Log Analytics Collections configured, do the following:
  - a. Edit the file: `/opt/HP/opsa/conf/deployment/opsa-deployment.xml`
  - b. Locate a line that looks similar to the following: `<process id="storm-supervisor" active="true" runsOnAppliance="Processing">`
  - c. Change **true** to **false** on that line as shown in the following bold text: `<process id="storm-supervisor" active="false" runsOnAppliance="Processing">`
  - d. Save the file.
  - e. Load the configuration change you just made by running the following commands:

```
/opt/HP/opsa/scripts/opsa-deployment-manager.sh  
  
/opt/HP/opsa/scripts/opsa-deployment-loader.sh
```
2. On the OBA Application Server, do the following to resubmit the Storm topology:
  - a. Run the following command: `/opt/HP/opsa/scripts/opsa-storm-kill-topology.sh`
  - b. Run the following command: `/opt/HP/opsa/scripts/opsa-storm-submit-topology.sh`

After completing the above steps, you should see performance improvement in the Log Analytics Collections in which you made these changes.

## Configuring Logger to Forward CEF Messages to Operations Analytics

Before configuring a Structured Log Collection, you might want to configure the TCP Forwarding feature, also known as TCP forwarding, which is explained in this section.

After you complete the configuration instructions in this section, the performance of the Operations Analytics Collector host significantly improves. You will also observe more real-time log messages in Operations Analytics.

The TCP Forwarding feature does not support a secured connection between the ArcSight Logger (Logger) and the Operations Analytics Collector host.

The configuration shown in this section changes the default way that the Logger passes data to the Operations Analytics Collector host. By default the Operations Analytics Collector host pulls data from Logger (instead of Logger pushing it to the Operations Analytics Collector host).

To configure Operations Analytics to receive messages from Logger, do the following:

**Note:** If you complete the instructions in this section you will be able to use the `-passive` mode with the `opsacollection-config.sh` script when publishing a structured log collection. The `-passive` mode is used to support Log Analytics and Structured Log collections shipped with Operations Analytics.

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
```

1. From Logger, navigate to **Configuration > Forwarders**.
2. Set the values as below in the Add Forwarder window then click **Next**.



| Field       | Value         |
|-------------|---------------|
| Name        | OpsaForwarder |
| Type        | TCP Forwarder |
| Filter Type | Unified Query |

3. Enter the values as below in the Edit Forwarder window and then click **Save**.

| Field                            | Value  |
|----------------------------------|--|
| Query                            | deviceVendor != "ArcSight"<br><br>Make sure the query you enter in the Query field represents the collection you plan to configure   |
| Preserve Syslog Timestamp        | false  |
| Perserver Original Syslog Sender | false  |
| IP/Host                          | Fully-qualified-domain-name of collector host. You must use the same value that is returned when you run the command:<br>opsacollection- config.sh -list - collectorhosts -username opsatenantadmin. |
| Port                             | 4888   |

4. Enable the new configuration by clicking the check box for the new forwarder you added.

Now the Logger should forward near real-time CEF messages to Operations Analytics.

**Note:** If you completed these instructions, you must use the -mode passive option with the opsacollection- config.sh script when creating the associated structured log collection. For example, run the following command:

```
$OPSA_HOME/bin/opsa-collection-config.sh -create -  
nodelist/tmp/mynodelist.properties -collectorhost <fully-qualified-domain-name  
of collector host> -source arcSight -domain <domain from template files> -group  
<domain from template files> -username opsatenantadmin - mode passive
```

## Increase JVM memory to improve collection performance

OBA Collector hosts use more memory resource if they contain large numbers of collections or if the collections they contain are configured to frequently collect data. In OBA Collector hosts in either of these configurations, you might need to increase its JVM memory allocation.

To adjust the JVM memory allocation for an OBA Collector host, do the following:

1. As a root user, edit the `/opt/HP/opsa/conf/opsa-collector-env` file.
2. Change the `Xmx` value to 4096 or 8196, depending on how much resource your OBA Collector host is using.
3. Save the file.
4. As a root user, run the following command from the OBA Collector host:

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

Now the changes you made to the JVM memory allocation on the OBA Collector host are in place.

## Increase the entity index size

Operations Bridge Analytics uses a repository for storing various configuration data. OBA tracks the usage of this memory and notifies you when it needs to increase in size.

The creation of keys and tags is important to keep the entity index size lower than the threshold. The entity index table is supposed to remain relatively small, and just contains keys for common data such as hostname, application, location, and other items. Creating wrong or incorrect keys and tags can cause the entity index key to become too large. See [Using Tags for Collections](#) for more information about correctly configuring keys and tags.

If OBA notifies you that you must increase the entity index size, do the following:

1. Run the following command:

```
select count(*) as count ,property_group_uid from opsa_default.entity_index
group by property_group_uid order by count desc
```

**Note:** The first rows show you which collection has been incorrectly defined.

2. Fix the keys used in the collection by using the instructions shown in [Using Tags for Collections](#).

You need to register the collection with the right keys as follows:

**Note:** The entity\_index could be just truncated and the instructions in this step provide an option to run the job to remedy the issue.


3.
  - a. Access `http://<OBA_application_server>:29902/mbean?objectname=OPSA-Infrastructure%3AService%3DSchedulerTaskManager`
  - b. displayJobsInfo
  - c. Entity Instance Loader Job
  - d. Execute Job now

The above steps result in the recreation of the entity index table with the correct data (after you recreate one or more problem collections with the correct keys).

# Manage users and tenants

This topic defines user accounts, user groups, and tenants and contains the procedures required to work with them.

To access

Click  **Settings** and select **User Manager**.

## Learn more

### About user accounts

As an Operations Bridge Analytics administrator, you must configure a user account for each user who needs to access the Operations Bridge Analytics graphical user interface.

Note the following:

- User accounts must be unique across all Tenants.

**Tip:** To ensure the user name is globally unique, enter a user's email address as the user name.

- Each user account must be assigned to a user group.

To create a user account, see ["How to add a user account" on page 41](#), `opsa-tenant-manager.sh` (available from `help > reference pages`), and ["How to create tenants using a script" on page 45](#).

The first time you log on, you will need to change the default password. Follow the password guidelines shown in the **Change Password** dialog box.

After ten failed attempts to access Operations Bridge Analytics from a specific user account, OBA denies access to users attempting access with this user account. This account restriction lasts for ten minutes. If you have any access problems, discuss them with your Operations Bridge Analytics administrator.

By default, new passwords must be selected for every user every 182 days. This time can be modified by an administrator. For details, see the *OBA Hardening Guide*.

## About user groups

User groups are predefined in Operations Bridge Analytics and determine which tasks each user account that is assigned to the user group can perform.

### Note:

- User accounts must be unique across all tenants.
- All user groups have access to the Operations Bridge Analytics graphical user interface.
- You cannot add a new user group to Operations Bridge Analytics.
- A user account was assigned to the **Super Admin** user group when Operations Bridge Analytics was installed.
- See `opsa-tenant-manager.sh` (available from help > reference pages) and ["How to create tenants using a script" on page 45](#) for information about assigning a user to a user group.

### Predefined user groups

| User Group   | Description  | Supported Tasks  |
|--------------|--|--|
| Super Admin  | <p><b>Note:</b> Operations Bridge Analytics permits only one Super Admin user.</p> <p>The user account assigned to this user group has access to the following information for each tenant defined:</p> <ul style="list-style-type: none"> <li>• User Accounts</li> <li>• User Groups</li> </ul>                     | <p>Add, modify, and delete tenants.</p> <p>Add, modify, and delete user accounts assigned to the Tenant Admin user group.</p>        |
| Tenant Admin | <p>User accounts assigned to this User Group have access to the following information only for the tenant to which they are assigned:</p> <ul style="list-style-type: none"> <li>• Collectors</li> <li>• Collections</li> <li>• Meta Data</li> <li>• Tags</li> <li>• User Accounts</li> <li>• User Groups</li> </ul> | <p>Add, modify, and delete user accounts.</p> <p>Manage the collectors, collections, meta data, and tags for a specified tenant.</p> |

**Predefined user groups, continued**

| User Group | Description   | Supported Tasks   |
|------------|---|---|
| User       | User accounts assigned to this User Group have access to the Operations Bridge Analytics graphical user interface and to only the meta data and data for the tenant to which they are assigned. | <p>Access and perform tasks using the Operations Bridge Analytics Dashboards.</p> <p><b>Note:</b> Users assigned to this user group can also add and delete tags from a collection. See <code>opsa-tag-manager.sh</code> (available from <code>help &gt; reference pages</code>) and <a href="#">"How to create tenants using a script "</a> on <a href="#">page 45</a> for more information.</p> |

New users are automatically assigned to a predefined user group. The user group to which a new user is assigned depends on the user group to which you are assigned when adding a new user.

**User groups assigned to new users**

| Your User Group | User Group Automatically Assigned to the New User |
|-----------------|---|
| Super Admin     | Tenant Admin                                      |
| Tenant Admin    | User  |

**About tenants**

Operations Bridge Analytics supports multi-tenancy. This means one instance of Operations Bridge Analytics can serve multiple customers. Tenants ensure isolation of meta data and data across customers. The meta data includes the following:

- Collections
- Database schema
- Tags
- Dashboards
- User Accounts

For example, if you are a Manage Service Provider or Software as a Service Provider with multiple customers, tenants enable you to ensure that each customer accesses only the data for its data center or network.

When you install Operations Bridge Analytics, by default Operations Bridge Analytics creates the **opsa\_default** tenant.

To create one or more tenants, see `opsa-tenant-manager.sh` (available from `help > reference` pages) and ["How to create tenants using a script" on page 45](#) for more information.

Important: tenant strategy

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

Before creating collections using the **Source Type 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 ["Register collector hosts" on page 48](#).
- Decide on which existing tenant to use.
- Create a new tenant and its corresponding Tenant Admin.

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 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.

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.

Operations Bridge Analytics provides the following predefined User Groups:

- **Super Admin:** During installation, the `opsaadmin` user gets created, and is assigned to the Super Admin user group. You set the password for this user during the 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 ["Manage users and tenants" on page 36](#)

for more information. See the *opsa-tenant-manager.sh* reference page (or the Linux man page) for information about creating and managing tenants.

- **Tenant Admin:** During installation, the `opsatenantadmin` user gets created, and is assigned to the Tenant Admin user group. You set the password for this user during the 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 ["Manage users and tenants" on page 36](#) for more information. See the *opsa-tenant-manager.sh* reference page (or the Linux man page) for information about creating and managing users for a tenant.
- **User:** During installation, the `opsa_default` user gets created, and is 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 Bridge 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 and tenants" on page 36](#) for more information.

If you plan to use a tenant model, you can create additional tenants from the Operations Bridge Analytics console or by using the `$OPSA_HOME/bin/opsa-tenant-manager.sh` script. See ["Manage users and tenants" on page 36](#) for more information about creating a tenant using the Operations Bridge 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 `$OPSA_HOME/bin/opsa-tenant-manager.sh` command from the OBA Application Server as a user assigned to the Super Admin User Group. See *Managing Users and Tenants* in the *Operation Analytics Help* for information about managing users and tenants. See the *opsa-tenant-manager.sh* reference page (or the Linux man page) for information about managing tenants.
2. Enter **Add a new tenant** and follow the interactive commands to add the new tenant.
3. Run the `$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 man page) or *Manage Users* in the *OBA 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-tenant-manager.sh` script. Do the following:




1. Run the `$OPSA_HOME/bin/opsa-tenant-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 man page) for more information.

## Tasks

### How to add a user account

1. Click  **Settings** and select **User Manager**.

Operations Bridge Analytics shows the **User Manager** form.

**Note:** You must belong to either the Super Admin or Tenant Admin User Group to access the **User Manager** option.

2. Click **Add User**.

Operations Bridge Analytics shows the **Add User** form. Follow the password guidelines shown in the **Add User** dialog box.

3. In the **User Name** attribute, enter the user account name.
  - Local Authentication
    - Enter the user account name into the **User Name** field
    - Enter the **Password** following the password guidelines.
  - LDAP Authentication

Enter the user account name into the **User Name** field.

**Note:** The user account you created will be automatically assigned to the current tenant.

**Tip:** If the User Naming Attribute in the LDAP Configuration is `userdn` `=userPrincipalName`, the user name must be an email address.

**Note:** When adding an LDAP authenticated user, Operations Bridge Analytics searches for the user being added in the configured LDAP server or servers. Operations Bridge Analytics adds the user only if it can find the user in one of the configured LDAP servers. If the user cannot be found in one of the configured LDAP servers, no user is added.

4. Finish entering your passwords for a locally authentication user, then click **Add**.

Operations Bridge Analytics lists the new user account in the **Users Manager** table with its associated user group and tenant.

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

5. Do the following:

- a. If you are using LDAP authentication, select the **LDAP Authenticated User** checkbox.

**Note:** If you are using LDAP authenticated users, you must follow the instructions shown in the OBA User Guide for the **LDAP Authenticated User** checkbox to appear.

- b. Finish entering your passwords, then click **Save**.

Operations Bridge Analytics lists the new user account in the **Users Manager** table with its associated user group and tenant.

See the `opsa-user-manager.sh` reference page (or the Linux man page) for more information.

You can also add a user account using the `opsa-user-manager.sh` script. Run the following command:

```
$OPSA_HOME/bin/opsa-user-manager.sh -add -loginUser <Super Admin or Tenant Admin User Name> -loginPassword <password> -newUser <new username> -newUserPassword <new user password>
```

**Note:** See the `opsa-user-manager.sh` reference page (or the Linux man page) for more information.

After creating a new user use the `opsa-user-manager.sh` script, to show a list of users run the commands shown in the following examples:

- **To list Tenant Admin users:** `$OPSA_HOME/bin/opsa-user-manager.sh -list -loginUser opsaadmin -loginPassword <opsaadmin password>`
- **To list users by Tenant:** `$OPSA_HOME/bin/opsa-user-manager.sh -list -loginUser <Tenant Admin User> -loginPassword <Tenant Admin Password>`

You can delete a user account using the `opsa-user-manager.sh` script. Run the following command:

```
$OPSA_HOME/bin/opsa-user-manager.sh -delete -loginUser <Tenant Admin User> -
loginPassword <Tenant Admin Password> -user <username>
```

## How to change your user account password

You can change your user local account password at any time. The password for an LDAP authenticated account can only be changed on the LDAP server.

### To change your user local account password:

1. In the upper right corner of the Operations Bridge Analytics console, click your user account name.
2. Select **Change Password**.

The **Change Password** dialog box appears (only for users that are using a local account). Follow the password guidelines shown in the **Change Password** dialog box and change your password.

3. Click **Update** after you finish to save your changes.

You can also modify the password for a user account using the `opsa-user-manager.sh` script. Run the following command:

```
$OPSA_HOME/bin/opsa-user-manager.sh -modify -loginUser <username> -loginPassword
<password> -newUserPassword <new user password>
```

#### Note:

- Run the `opsa-user-manager.sh` command as an `opsa` user, not as a root user. Running `opsa-user-manager.sh` as a root user is not supported.
- See the `opsa-user-manager.sh` reference page (or the Linux man page) for more information.

## How to change the account locking threshold

If the number of times you fail to successfully log on to the Operations Bridge Analytics console exceeds the default locking threshold, you will be locked out of Operations Bridge Analytics. Do the following to change the default locking threshold:

1. As an `opsa` user, edit the `/opt/HP/opsa/conf/opsa-config.properties` file on the OBA Application Server.
2. Change the `failed.counter.threshold` value to the value you desire for the number of failed log ons.
3. Change the `user.account.lockout.timeout` value to the value you desire of the amount of time to wait before the user account lock expires.

4. Save your work.
5. Run the following command from the OBA Application Server to implement these property changes:

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

See the `opsa-server` reference page (or the Linux man page) for more information.

## How to add a tenant

As an Operations Bridge Analytics administrator, if you belong to the **Super Admin** User Group, you can add one or more tenants.

### Note:

- You can also use `opsa-tenant-manager.sh` (available from help > reference pages) to add tenants to Operations Bridge Analytics.
- If you do not configure one or more tenants, Operations Bridge Analytics stores all of the meta data, collection and query information in the **opsa\_default** tenant.
- User account names must be unique across all tenants.

### To add a tenant and a tenant admin:

1. Click  **Settings** and select **User Manager**.

Operations Bridge Analytics shows the **User Manager** form.

**Note:** You must belong to either the Super Admin or Tenant Admin User Group to access the **User Management** option.

2. Click **Add User**.

Operations Bridge Analytics shows the **Add User** form.

3. If you belong to the Super Admin User Group, in the **Tenant** attribute, enter the name of a tenant you want to create. Tenant names cannot begin with a number. The initial alpha character can be followed by alphanumeric characters (including an underscore).

**Note:** OBA converts all tenant names to lowercase.

4. Click **No matches found - Click to Add**.
5. In the **Add Tenant** dialog, click **OK**.

6. Add a Tenant Admin to the current Tenant.

For the **User Name** attribute, enter the user account name. Select one of following options for authentication:

- Local Authentication
  - Enter the user account name into the **User Name** field.
  - Enter the **Password** following the password guidelines.
- LDAP Authentication

Enter the user account name into the **User Name** field.

7. Click **OK** to add the Tenant Admin.

#### How to create tenants using a script

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 `$OPSA_HOME/bin/opsa-tenant-manager.sh` command from the OBA Application Server as a user assigned to the Super Admin User Group. See *Managing Users and Tenants* in the *Operation Analytics Help* for information about managing users and tenants. See the *opsa-tenant-manager.sh* reference page (or the Linux man page) for information about managing tenants.
2. Enter **Add a new tenant** and follow the interactive commands to add the new tenant.
3. Run the `$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 man page) or *Manage Users* in the *OBA 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-tenant-manager.sh` script. Do the following:

1. Run the `$OPSA_HOME/bin/opsa-tenant-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 man page) for more information.

#### How to delete a tenant

To delete a tenant from Operations Bridge Analytics, you must delete the tenant, then remove files from the OBA Collector host 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](#) for more information.
2. There are two methods to use to delete a tenant from Operations Bridge Analytics. To delete a tenant from Operations Bridge Analytics, **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 man page) for information about creating and managing tenants.

3. To remove files from your configured collectors, do the following:
  - a. From each OBA Collector host that contains collectors for the tenant being removed, run only one of the following commands to remove the tenant collection configuration:
    - If the OBA Collector host is only collecting data for the tenant being removed:
 

```
rm -rf /opt/HP/opsa/conf/collection/config.files/<collector host>
```
    - If the OBA Collector host is collecting data for multiple tenants:
 

```
rm -rf /opt/HP/opsa/conf/collection/config.files/<collector host>/<tenant>
```
  - b. *Only complete this step if an OBA Collector host currently collects data for tenants other than the one being deleted.* Run the following command from the OBA Application Server to publish this collection configuration to the OBA Collector host. Use a Tenant Admin user for one of the

other active tenants for which that this OBA Collector host 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 OBA Collector host. To verify that the collector configuration published successfully, look for a message stating that the publish was successful, a table was successfully created, and the collection was restarted.

- c. From the OBA Collector host, run the following commands to remove specific files from the OBA Collector host 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>`

# Register collector hosts

Register the OBA Collector host you plan to use with the OBA Application Server.

**Note:** Before completing the steps in this section, it is recommended that you add an entry to the `/etc/hosts` file for the OBA Collector host you plan to register.

## Automatically Creating Alert Collections

All of the alerts generated by Operations Bridge Analytics are stored as collections. This collected information is used to show 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. A new Alerts Collection gets created with each newly created Operations Bridge Analytics tenant.

See [Alerts](#) for more details about the Alerts feature.

## Registering an OBA Collector Host

To register an OBA Collector host with an OBA Application Server, do the following:

1. Run the following command on the OBA Collector host 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 OBA Application Server:

```
$OPSA_HOME/bin/opsa-collection-config.sh -register -collectorhost <fully  
qualified domain name of the collector host> -port 9443 -username  
opsatenantadmin
```

If you have the OBA Collector host 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 OBA Collector host, use the `-coluser` and `-colpass` option in this command. You must also use the fully qualified domain name of the OBA Collector host when using this command. See the `opsa-collection-config.sh` reference page (or the Linux man page) for more information.

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 OBA listens is 9443. You can modify this port in cases of port conflicts. For more information, see the *OBA User Guide*.

If the script communicates successfully with the OBA Collector host, it registers it in the OBA Application Server database and displays a success message.

### Checking the Registration Status of a OBA Collector Host

To check the registration status of your OBA Collector host, do the following:

1. Run the following command: `$OPSA_HOME/bin/opsa-collection-config.sh -list -collectorhosts -allversions -username opsatenantadmin`
2. Review the list of registered OBA Collector hosts. If the OBA Collector host you plan to register is not on the list, you must register it using the instructions in this section.

## Resolve host aliases

If you have multiple collections importing data from the same source, the source may have multiple aliases. For example, in one collection it may be identified by the IP address and in another collection it may be identified by the FQDN. Operations Bridge Analytics automatically detects host aliases and incorporates the data into your dashboards and search results.

To access

Click **Data Manager > Host Aliases Settings**.

## Learn more

### About host aliases

If you have multiple collections importing data from the same source, the source may have multiple aliases. For example, in one collection it may be identified by the IP address and in another collection it may be identified by the FQDN. Operations Bridge Analytics automatically detects host aliases and incorporates the data into your dashboards and search results.

Host alias data is used when displaying search results for a specific host, or in dashboards when a host is specified in an AQL. When metric data is displayed as a search result, data for each alias is displayed separately.

In an AQL query, data for all aliases is returned unless you use a double equals symbol == in place of a single one. In such a case, only the results exactly matching the specified host identifier will be displayed.

In the topology manager, if you add a host that has aliases the aliases will be added as well.

# Tasks

## How to configure host alias settings

1. Click **Data Manager** on the left side of the Operations Bridge Analytics console; then select **Host Aliases Settings**.
2. By default, host aliases are enabled and function automatically, using DNS resolving to identify host aliases.

Hosts Aliases Back

Enabling Identification of Host aliases allows Operations Analytics to automatically discover and use host aliases. DNS resolving is used in this process, or you can manually import a host file.

☒ Enable Identification of Host aliases

☒ Use DNS resolving to identify Host aliases


To manually add Host Aliases, upload a host file: ? Upload

☐ Remove manually imported aliases if DNS resolving detects a mismatch

Download All Saved Aliases ?

3. If DNS resolving cannot function in your environment, you can manually add a list of hosts and their aliases in the form of a host or .csv file.  
  
Disable the **DNS resolving** checkbox and use the **Upload** button to upload a file. Each line in the file should represent one host and its aliases.
4. Periodically, the list of host aliases is verified to make sure nothing in your environment has changed. In version 3.0, this process occurs once per week. If a change is detected, and you want the mismatches to be removed even though the data was uploaded manually as a file, select the **Remove manually imported aliases...** checkbox.
5. To download the current list of aliases in the form of a .csv file, select **Download All Saved Aliases**. Each line in the file represents one host and its aliases.

## How to see if data is coming from a host alias

1. In any query pane, select **More Pane Actions**  and click **View Data Origins**.
2. The original host identifier is specified in the **Original Name** column.

If the data displayed has been modified because of a detected host alias, the **Displayed Name** will be different from the **Original Name**.

## How to reload host aliases after deletions

If Operations Bridge Analytics shows any inaccurate or incorrect aliases you might want to delete the current aliases. Use the information in this section to clean up the current aliases and restart the alias resolving.

**Note:** For the steps in this section to be successful, you must have configured collections and they must be actively collecting data. After you delete any host aliases, Operations Bridge Analytics does not retrieve new aliases for hosts that are already known.

If you want to Operations Bridge Analytics to start over and resolve the aliases again, do the following:

1. Run the following commands from the Vertica server to delete the content from the `host_lookup_aliases` and `host_lookup_unique_hosts` tables from the Vertica database.
  - a. `TRUNCATE TABLE <tenant> host_lookup_aliases;`
  - b. `TRUNCATE TABLE <tenant> host_lookup_unique_hosts;`
2. Run the following command from the OBA Application Server to restart the `opsa-server` process: `$OPSA_HOME/bin/opsa-server restart`
3. Run the following command from the OBA Application Server to restart the `opsa-task-manager` process: `$OPSA_HOME/bin/opsa-task-manager restart`
4. Run the following commands from the OBA Collector host to resubmit the aliases resolving processing:
  - a. `$OPSA_HOME/scripts/opsa-post-persist-processing-kill-topology.sh`
  - b. `$OPSA_HOME/scripts/opsa-post-persist-processing-submit-topology.sh`

## Set up security

There are several security methods you can configure for user access, authentication, and protecting data using tenants for Operations Bridge Analytics.

The following information is a summary of the security hardening recommendations for OBA. The hardening instructions in this section are optional.

This section includes the following topics:


- ["SSL for OBA components" on page 55](#)
- ["HTTP and HTTPS" on page 91](#)
- ["Single Sign-On" on page 93](#)
- ["Configure LDAP authentication" on page 97](#)
- ["PKI" on page 102](#)
- ["Resetting user passwords" on page 106](#)
- ["Changing the port used by the OBA console" on page 107](#)

## Encrypting Operations Bridge Analytics

Each OBA application server uses a separate encryption key to provide secure data for each OBA deployment.

OBA provides the `opsa-key-manager.sh` script. If you want to modify the encryption password and salt for an Operations Bridge Analytics installation, do the following from the OBA Application Server:

1. Run the `opsa-key-manager.sh` script as a user with super-admin credentials.
2. When prompted, follow the instructions shown by the `opsa-key-manager.sh` script.
3. After the `opsa-key-manager.sh` script completes, Operations Bridge Analytics has a new encryption key and salt.

See the `opsa-key-manager.sh` reference page (or the Linux man page), for more information. To view Operations Bridge Analytics reference pages, select  > **Reference Pages** in the Operations Analytics console.

## Other security considerations

If you have multiple tenants, make sure that you use different OBA collector hosts for each tenant to ensure data separation for each tenant.

When selecting credentials to connect to the OMi database, HPE recommends that you select a user with minimal credentials for reading the required information. Selecting a more powerful user could present a security vulnerability.

## SSL for OBA components

This section provides information about securing the communication between OBA components. If you choose to configure SSL, HPE recommends that you secure all OBA connections, and not just individual components.

This section includes:

- [SSL for Servers and Collectors](#)
- ["Two-Way SSL for accessing ArcSight Logger" on page 78](#)
- ["SSL for communication between Vertica and OBA" on page 80](#)
- ["SSL for the SMTP Server used for OBA Alerts" on page 88](#)

## Configuring SSL for the OBA application servers and collectors

One-way SSL provides secure communication between the client and the OBA Application Server and the Collector host. 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. SSL should be enabled for both the OBA application server and the collectors.

The workflow is as follows:

1. Unregister the collector from the application server
2. Enable SSL communication to the application server
3. Enable SSL communication to the collector
4. Reregister the collector using SSL

This section includes the following:

- Information on configuring SSL for the OBA application server and collector. If you are using a self-signed certificate, perform the workflow starting with ["Configuring SSL communication to the OBA application server and collector with a self-signed certificate" on page 64](#). If you are using a certificate authority (CA) signed certificate, perform the workflow as described in the section ["Configuring SSL communication to the OBA application server and collector with a Certificate Authority \(CA\) signed certificate " below](#). If your environment contains multiple application servers and collectors, you must issue a new certificate for each system, not one certificate for all systems. On each server, import the certificate issued for the server and import the CA's root certificate to the trust store on each server.
- [Editing the SSL Configuration for the OBA Collector Host](#)
- [Disabling the SSL Configuration for the OBA Collector Host](#)
- [Managing the Keystore and Truststore for the OBA Collector Host](#)

## Configuring SSL communication to the OBA application server and collector with a Certificate Authority (CA) signed certificate

If you set up SSL connection for the OBA application server(s), you must also set up SSL connection for the OBA collector(s). For environments with more than one server, you must issue a different certificate for each of your servers, and import the CA's root certificate into the truststore on each server. This must be repeated for each OBA application server and each OBA collector.

### 1. Unregister the collections and collectors

If OBA is already configured with collections (the collector is already registered) you must unregister all collections either by using the OBA user interface or the command line. To unregister collections, do the following:

**Note:** The procedure below shows the command line procedure, but if one collection collects many elements, it is easier to delete the collection in the user interface. Go to the **Source Type Manager**, select the collection, and click **Delete**.



- a. Unregister the collector:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -username <tenant admin username> -password <tenant admin
password>
```

If the collector is successfully unregistered, continue with Step e: "Check that the collector was unregistered successfully".

If there are any collectors still registered, continue with Step b: "Run the following command to list the collections".

- b. Run the following command to list the collections:

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

- c. Unregister the collections that were listed with the previous command:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -source <source> -domain <domain> -group <group> -username
<tenant admin user name> [-password <tenant admin password>]
```

- d. Unregister the collector:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -username <tenant admin username> -password <tenant admin
password>
```

- e. Check that the collector was unregistered successfully:

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

2. **Configure SSL communication to the OBA application server.** Complete the following steps to enable SSL communication to the OBA application server using a CA signed certificate:

- a. Before enabling SSL to the OBA Application Server, complete this step on the OBA Application Server to create a user in JBoss **Management Realm**.

Do the following:

- i. Run the `$OPSA_HOME/jboss/bin/add-user.sh` script as the `opsa` user.
- ii. 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.

- iii. When queried about the group to which the user should belong, accept the default settings.

When queried if the user will be used for one AS process to connect to another AS process, select No.

If the script prompts you to add the JBOSS secret, you can safely ignore this. OBA does not use this secret.

- b. As the `opsa` user, change the directory to `/opt/HP/opsa` and run the `$OPSA_HOME/bin/opsa-server-manager.sh` script on the OBA Application Server host. See the *opsa-server-manager.sh* reference page (or the Linux man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

Make the following selections in the resulting flow of options:

- i. Select option **2: Configure SSL**.
- ii. Select option **1: Generate self signed certificate for OPSA** to generate a selfsign key pair.
- iii. Select option **2: Generate certificate signing request** to generate a certificate request for a signed CA certificate. Choose the `opsa_server` alias and save the certificate to `/tmp/opsa_server_crf.csr`.
- c. After creating the request file, sign the Certificate Request on your CA. The certificate must be signed for server and client authentication. Download the certificate chain on based 64 encoded format (p7b extension) and copy the file to the OBA application server in the `/tmp` folder.
- d. Select option **3: Import CA signed certificate to OPSA keystore**. Specify the absolute path to the certificate.
- e. Download the root CA certificate from your CA and copy the file to the OBA application server in the `/tmp` folder.
- f. Select option **4: Import trusted certificate to OPSA truststore**. Enter the exact path to the root CA certificate that you downloaded.

Option 4 asks you to run a command as root user. Therefore, open another terminal, make sure you are root user, and copy and paste the requested command from the option 4 terminal to the new terminal where you are root user.

- g. As the **opsa** user in the original tab, select option **8: Enable/Disable 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 the `opsa_server` alias from the list of key aliases in the OPMA keystore.
- h. Select the option **13: Go back to main menu**.
- i. Select option **7: Exit**.
- j. As the root user, restart all OMA processes by calling `sudo /opt/HP/opsa/bin/opsa restart`.

**Note:** Your configuration changes will not occur unless the application server is restarted. If you receive an error, follow the instructions in the error messages.

3. **Configure SSL communication to the OMA collector.** Complete the following steps to enable SSL communication to the OMA collector using a CA signed certificate:

- a. As the `opsa` user, change the directory to `/opt/HP/opsa` and run the `$OPMA_HOME/bin/opsa-collector-manager.sh` script on the OMA Collector host. See the `opsa-collector-manager.sh` reference page (or the Linux man page), for more information.

**Note:** This script must be run out of the `$OPMA_HOME` directory.

Make the following selections in the resulting flow of options:

- i. Select option **1: Configure SSL**.
- ii. Select option **1: Generate a self signed certificate for OPMA**.
- iii. Select option **2: Generate a certificate signing request**. Choose the `opsa_server` alias. Save the certificate to `/tmp/opsa_collector_crf.csr`.
- b. After creating the request file, sign the Certificate Request on your CA. The certificate must be signed for server and client authentication. Download the certificate chain on base64 encoded format (p7b extension). Copy this file to the Operations Bridge Analytics collector in the `/tmp` folder.
- c. Select option **3: Import CA signed certificate to OPMA keystore**. Specify the absolute path to the certificate.
- d. Download the root CA certificate from your CA and copy the file to the OMA application server in the `/tmp` folder.
- e. Select option **4: Import trusted certificate to OPMA truststore**. Enter the exact path to the

root CA certificate that you downloaded.

Option 4 asks you to run a command as root user. Therefore, open another terminal, make sure you are root user, and copy and paste the requested command from the option 4 terminal to the new terminal where you are root user.

- f. Select option **8: Enable/Disable SSL**. The `opsa_collector_manager.sh` script prompts you for the certificate alias to be used for SSL communication. Enter the `opsa_server` alias from the list of key aliases in the OPSA keystore.
- g. Select option **13: Go back to main menu**.
- h. Select option **5: Exit**.
- i. Restart all OBA processes by calling `sudo /opt/HP/opsa/bin/opsa restart`.

**Note:** Your configuration changes will not occur unless the collector is restarted. If you receive an error, follow the instructions in the error messages.

#### 4. Reregister the OBA collector.

- a. Run the following command as the **opsa** user on the OBA application server to register the OBA collector with the SSL command:

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

- b. Reregister all collections that you unregistered in step 1.

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

**Note:** You must complete this certificate import on both the OBA Application Server (for the rawlog query) and the OBA Collector host (for the structured log query). Follow these steps:

- a. Log on to the ArcSight Logger UI, 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 OBA Collector host and on the OBA Application Server.

- e. Complete this step on both the OBA Collector host and on the OBA Application Server to import the certificate.
6. *OMi integration only.* This step is only required when root certificates for OMi and OBA are issued by different CAs.

#### Import the OMi certificate into OBA.

- a. On an OMi data processing server, export the certificates by running the following command:

```
ovcert -exporttrusted -file /tmp/omi.cer
```

- b. Copy the file to all of the OBA collectors.
- c. As the **opsa** user on each collector, change the directory to `/opt/HP/opsa` and run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script on the OBA Collector host. See the *opsa-collector-manager.sh* reference page (or the Linux man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

- d. Make the following selections in the resulting flow of options:
  - i. Select option **1: Configure SSL**.
  - ii. Select option **4: Import trusted certificate to OPSA truststore**. Specify the file exported in the previous steps, for example `/tmp/omi.cer`. If prompted to run a procedure manually, execute it in another shell.
  - iii. Select option **13: Go back to main menu**.
  - iv. Select option **5: Exit**.
  - v. Restart all OBA processes by calling `sudo /opt/HP/opsa/bin/opsa restart`.

**Note:** Your configuration changes will not occur unless the collector is restarted. If you receive an error, follow the instructions in the error messages.

7. *OMi integration only.* This step is only required when root certificates for OMi and OBA are issued by different CAs.

#### Import the OBA certificate into OMi.

- a. On all OBA application servers, export the OBA server certificate:
  - i. As the **opsa** user, change the directory to `/opt/HP/opsa` and run the `$OPSA_HOME/bin/opsa-server-manager.sh` script on the OBA Application Server. See the *opsa-server-manager.sh* reference page (or the Linux man page), for more

information.

**Note:** This script must be run out of the \$OPSA\_HOME directory.

Make the following selections in the resulting flow of options:

- A. Select option **2: Configure SSL**.
  - B. Select option **5: Export certificate from OPSA keystore**. Store the file, for example as /tmp/opsa\_server.cer.
  - C. Select option **13: Go back to main menu**.
  - D. Select option **7: Exit**.
- b. Copy the exported OBA server certificates to one of the OMi gateway servers.
  - c. Import all the OBA server certificates to OMi:
    - i. On one of the OMi gateway servers, run the following command for all OBA server certificates:
 

```
<OMi_HOME>/bin/opr-cert-mgmt.[bat|sh] -import opsa_server <path>
```

For more information on the opr-cert-mgmt tool, see the *OMi Extensibility Guide*.
  - d. Export the OBA collector certificates by doing the following on all OBA collectors:
    - i. As the opsa user, change the directory to /opt/HP/opsa and run the \$OPSA\_HOME/bin/opsa-collector-manager.sh script on the OBA Collector host. See the *opsa-collector-manager.sh* reference page (or the Linux man page), for more information.
    - ii. Select option **1: Configure SSL**.
    - iii. Select option **5: Export certificate from OPSA keystore**. Store the file, for example as /tmp/opsa\_collector.cer.
    - iv. Select option **13: Go back to main menu**.
    - v. Select option **5: Exit**.
    - vi. Convert the certificate to PEM format by running the following commands (including example file names):
 

```
-keytool -import -alias opsa_server -file /tmp/opsa_collector.cer -
keystore /tmp/test.keystore -storepass 123456

-keytool -exportcert -alias opsa_server -file /tmp/opsa_collector.pem -
rfc -keystore /tmp/test.keystore -storepass 123456
```
    - vii. Copy the /tmp/opsa\_collector.pem certificate to one OMi data processing server and

import the certificate:

```
ovcert -importtrusted -file /tmp/opsa_collector.pem
```

OMi automatically distributes the certificate to all OMi data processing and gateway servers.

- e. Restart OMi on all servers to distribute the trust(s) to all of the servers' truststores.

## Import a certificate created and signed by an external CA

To import a certificate created and signed by an external CA, not issued by OBA issued certificate signing request (CSR), perform the following steps:

1. Save the certificate to a temp location in the PKCS12 format – Client and Server Certificate. For example:

```
/tmp/server_certificate.p12.
```

2. Retrieve the certificate's alias using the following command:

```
keytool -list -keystore /tmp/server_certificate.p12 -storetype pkcs12
```

Sample output:

```
Keystore type: PKCS12
```

```
Keystore provider: SunJSSE
```

```
Your keystore contains 1 entry
```

```
<alias>, Dec 7, 2016, PrivateKeyEntry,
```

```
Certificate fingerprint (SHA1):
```

```
6F:AF:A6:87:BE:9A:42:10:83:89:C0:79:B2:B6:CA:90:43:71:47:5B
```

3. Change the password of the OPSA keystore and truststore using the command `opsa-server-manager.sh`.

As the `opsa` user, change the directory to `/opt/HP/opsa` and run the `$OPSA_HOME/bin/opsa-server-manager.sh` script on the OBA Application Server.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

Make the following selections in the resulting flow of options:

- a. Select option **2: Configure SSL**.
- b. Select option **6: Modify OPSA keystore/truststore password**

- c. Specify a new password.
4. Import the server certificate into the OPSA keystore using the following command:

```
keytool -importkeystore -srckeystore /tmp/server_certificate.p12 -srcstoretype
pkcs12 \
-srcalias <alias> -destkeystore /opt/HP/opsa/conf/ssl/opsa-keystore.jks -
deststoretype jks \
-deststorepass <password> -destkeypass <password> -destalias opsa_server
```

Where <alias> is the certificate's alias as retrieved in Step 2 and <password> is the new password specified in Step 3.

5. Import the root CA certificate that was used to sign the just imported certificate into the OPSA truststore using the command `opsa-server-manager.sh`.  
As the `opsa` user, change the directory to `/opt/HP/opsa` and run the `$OPSA_HOME/bin/opsa-server-manager.sh` script on the OBA Application Server.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

Make the following selections in the resulting flow of options:

- a. Select option **2: Configure SSL**.
- b. Select option **4: Import trusted certificate to OPSA truststore**
6. Restart OMi on all servers to distribute the trust(s) to all of the servers' truststores.

## Configuring SSL communication to the OBA application server and collector with a self-signed certificate

To set up SSL communication to the OBA application server and collector using a self-signed certificate, perform the following steps:

### 1. Unregister the collections and collectors

If OBA is already configured with collections (the collector is already registered) you must unregister all collections either by using the OBA user interface or the command line. To unregister collections, do the following:



**Note:** The procedure below shows the command line procedure, but if one collection collects many elements, it is easier to delete the collection in the user interface. Go to the **Source Type Manager**, select the collection, and click **Delete**.

- a. Unregister the collector:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -username <tenant admin username> -password <tenant admin
password>
```

If the collector is successfully unregistered, continue with Step e: "Check that the collector was unregistered successfully".

If there are any collectors still registered, continue with Step b: "Run the following command to list the collections".

- b. Run the following command to list the collections:

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

- c. Unregister the collections that were listed with the previous command:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -source <source> -domain <domain> -group <group> -username
<tenant admin user name> [-password <tenant admin password>]
```

- d. Unregister the collector:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -username <tenant admin username> -password <tenant admin
password>
```

- e. Check that the collector was unregistered successfully:

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

2. **Configure SSL communication to the OBA application server.** Complete the following steps as the opsa user to enable SSL communication to the OBA application server using a self signed certificate:

- a. Before enabling SSL to the OBA Application Server, complete this step on the OBA Application Server to create a user in JBoss **Management Realm**.

Do the following:

- i. Run the `$OPSA_HOME/jboss/bin/add-user.sh` script.
- ii. For the first question, answer **"a" - Management User (mgmt-users.properties)**, then follow the instructions.
- iii. When queried about the group to which the user should belong, accept the default settings.

When queried if the user will be used for one AS process to connect to another AS process, select No.

If the script prompts you to add the JBOSS secret, you can safely ignore this. OBA does not use this secret.

- b. Set the self-signed certificate attributes, like common name, country, and validity, by editing the `/opt/HP/opsa/conf/ssl/cert/opsa-self-signed-cert.template` file.
- c. As the `opsa` user, change the directory to `/opt/HP/opsa` and run the `$OPSA_HOME/bin/opsa-server-manager.sh` script on the OBA Application Server host. See the *opsa-server-manager.sh* reference page (or the Linux man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

Make the following selections in the resulting flow of options:

- i. Select option **2: Configure SSL**.
- ii. Select option **1: Generate self signed certificate for OPSA** to generate a self sign key pair.
- iii. Select option **5: Export certificate from OPSA keystore**. Enter the `opsa_server` alias from the list of key aliases in the OPSA keystore. Store the file, for example as `/tmp/opsa_server.cer`.
- iv. Select option **4: Import trusted certificate to OPSA truststore**. Specify the file exported in the previous step, for example `/tmp/opsa_server.cer`. If prompted to run a procedure manually, execute it in another shell.
- v. All server certificates must be imported to the truststore of all collectors, and all collector certificates must be imported to the truststore of all servers (import of collector certificates to other collectors and server certificates to other servers is not required).
- d. Select option **8: Enable/Disable SSL** and select Yes when asked if SSL should be enabled. . 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 the `opsa_server` alias from the list of key aliases in the

OPSA keystore.

- e. Select the option **13: Go back to main menu**.
- f. Select option **7: Exit**.
- g. Restart all OBA processes by calling `sudo /opt/HP/opsa/bin/opsa restart`.

**Note:** Your configuration changes will not occur unless the application server is restarted. If you receive an error, follow the instructions in the error messages.

3. **Configure SSL communication to the OBA collector.** Complete the following steps to enable SSL communication to the OBA collector using a self-signed certificate:

- a. Set the self-signed certificate attributes, like common name, country, and validity, by editing the `/opt/HP/opsa/conf/ssl/cert/opsa-self-signed-cert.template` file.
- b. As the `opsa` user, change the directory to `/opt/HP/opsa` and run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script on the OBA Collector host. See the *opsa-collector-manager.sh* reference page (or the Linux man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

Make the following selections in the resulting flow of options:

- i. Select option **1: Configure SSL**.
- ii. Select option **1: Generate a self signed certificate for OPSA**.
- iii. Select option **5: Export certificate from OPSA keystore**. Enter the `opsa_collector` alias from the list of key aliases in the OPSA keystore. Store the file, for example as `/tmp/opsa_collector.cer`.
- iv. Select option **4: Import trusted certificate to OPSA truststore**. Specify the file exported in the previous step, for example `/tmp/opsa_collector.cer`. If prompted to run a procedure manually, execute it in another shell.
- v. All server certificates must be imported to the truststore of all collectors, and all collector certificates must be imported to the truststore of all servers (import of collector certificates to other collectors and server certificates to other servers is not required).
- c. Select option **8: Enable/Disable SSL** and answer Yes when asked if SSL should be enabled.
- d. Select option **13: Go back to main menu**.
- e. Select option **5: Exit**.
- f. Restart all OBA processes by calling `sudo /opt/HP/opsa/bin/opsa restart`.

#### 4. Reregister the OBA collector.

- a. On the OBA application server as the `opsa` user, run the following command to register the collector with the SSL command:

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

5. Log to OBA using `https://<oba server IP or FQDN>:8443/opsa`.
6. *Optional.* Select the **Import trusted certificate to OPSA server truststore** option to import trusted certificates (if any). For example, you can add ArcSight Logger's server certificate to the Operations Bridge Analytics truststore file.

**Note:** You must complete this certificate import on all the OBA Collector hosts (for the structured log query). Follow these steps:

- a. Log on to the ArcSight Logger UI, 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 OBA Collector host and on the OBA Application Server.
  - e. Complete this step on both the OBA Collector host and on the OBA Application Server to import the certificate.
7. *OMi integration only.* This step is only required when root certificates for OMi and OBA are issued by different CAs.

#### Import the OMi certificate into OBA.

- a. On an OMi data processing server, export the certificates by running the following command:

```
ovcert -exporttrusted -file /tmp/omi.cer
```

- b. Copy the file to all of the OBA collectors.
- c. As the `opsa` user on each collector, change the directory to `/opt/HP/opsa` and run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script on the OBA Collector host. See the *opsa-collector-manager.sh* reference page (or the Linux man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

- d. Make the following selections in the resulting flow of options:
  - i. Select option **1: Configure SSL**.
  - ii. Select option **4: Import trusted certificate to OPSA truststore**. Specify the file exported in the previous steps, for example `/tmp/omi.cer`. If prompted to run a procedure manually, execute it in another shell.
  - iii. Select option **13: Go back to main menu**.
  - iv. Select option **5: Exit**.
  - v. Restart all OBA processes by calling `sudo /opt/HP/opsa/bin/opsa restart`.

**Note:** Your configuration changes will not occur unless the collector is restarted. If you receive an error, follow the instructions in the error messages.

8. *OMi integration only*. This step is only required when root certificates for OMi and OBA are issued by different CAs.

#### Import the OBA certificate into OMi.

- a. On all OBA application servers, export the OBA server certificate:
  - i. As the `opsa` user, change the directory to `/opt/HP/opsa` and run the `$OPSA_HOME/bin/opsa-server-manager.sh` script on the OBA Application Server. See the *opsa-server-manager.sh* reference page (or the Linux man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

Make the following selections in the resulting flow of options:

- A. Select option **2: Configure SSL**.
  - B. Select option **5: Export certificate from OPSA keystore**. Store the file, for example as `/tmp/opsa_server.cer`.
  - C. Select option **13: Go back to main menu**.
  - D. Select option **7: Exit**.
- b. Copy the exported OBA server certificates to one of the OMi gateway servers.
  - c. Import all the OBA server certificates to OMi:
    - i. On one of the OMi gateway servers, run the following command for all OBA server certificates:

```
<OMi_HOME>/bin/opr-cert-mgmt.[bat|sh] -import opsa_server <path>
```

For more information on the `opr-cert-mgmt` tool, see the *OMi Extensibility Guide*.

d. Export the OBA collector certificates by doing the following on all OBA collectors:

- i. As the `opsa` user, change the directory to `/opt/HP/opsa` and run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script on the OBA Collector host. See the *opsa-collector-manager.sh* reference page (or the Linux man page), for more information.
- ii. Select option **1: Configure SSL**.
- iii. Select option **5: Export certificate from OPSA keystore**. Store the file, for example as `/tmp/opsa_collector.cer`.
- iv. Select option **13: Go back to main menu**.
- v. Select option **5: Exit**.
- vi. Convert the certificate to PEM format by running the following commands (including example file names):

```
-keytool -import -alias opsa_server -file /tmp/opsa_collector.cer -  
keystore /tmp/test.keystore -storepass 123456
```

```
-keytool -exportcert -alias opsa_server -file /tmp/opsa_collector.pem -  
rfc -keystore /tmp/test.keystore -storepass 123456
```

- vii. Copy the `/tmp/opsa_collector.pem` certificate to one OMi data processing server and import the certificate:

```
ovcert -importtrusted -file /tmp/opsa_collector.pem
```

OMi automatically distributes the certificate to all OMi data processing and gateway servers.

e. Restart OMi on all servers to distribute the trust(s) to all of the servers' truststores.

## Editing the SSL configuration for the OBA application server or collector

### How to change the certificate alias used for SSL communication with the application server

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 man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

2. Select option **1: Configure SSL**.
3. Select option **7: Change key alias to be used for SSL communication**.
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 option **13: Go back to main menu** option, then select option **6: Restart OPSA server** option to restart the OBA Application Server.

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

### How to change server certificate used for SSL communication with the collector

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

1. Unregister the collector. Run the following command as the "opsa user":

#### **Unregister the collections and collectors**

If OBA is already configured with collections (the collector is already registered) you must unregister all collections either by using the OBA user interface or the command line. To unregister collections, do the following:

**Note:** The procedure below shows the command line procedure, but if one collection collects many elements, it is easier to delete the collection in the user interface. Go to the **Source**

**Type Manager**, select the collection, and click **Delete**.

- a. Unregister the collector:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -username <tenant admin username> -password <tenant admin
password>
```

If the collector is successfully unregistered, continue with Step e: "Check that the collector was unregistered successfully".

If there are any collectors still registered, continue with Step b: "Run the following command to list the collections".

- b. Run the following command to list the collections:

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

- c. Unregister the collections that were listed with the previous command:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -source <source> -domain <domain> -group <group> -username
<tenant admin user name> [-password <tenant admin password>]
```

- d. Unregister the collector:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -username <tenant admin username> -password <tenant admin
password>
```

- e. Check that the collector was unregistered successfully:

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

2. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script. See the *opsa-collector-manager.sh* reference page (or the Linux man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

3. Select option **1: Configure SSL**.
4. Select option **7: Change key alias to be used for SSL communication**.



5. 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.
6. Select option **13: Go back to main menu**, then select option **4: Restart OPSA Collector** option to restart the OBA Collector host.

**Note:** Your configuration changes will not occur unless the OBA Collector host is restarted.

7. Run the following command as the "opsa" user to re-register the collector with the SSL:

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

## Disabling the SSL configuration for the OBA application server or collector

### How to disable the SSL configuration for the OBA application server

To disable the SSL communication, do the following:

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script from the OBA Application Server as the "opsa" user.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

2. Select option **2: Configure SSL**.
3. Select option **8: Enable/Disable SSL**.
4. The `$OPSA_HOME/bin/opsa-server-manager.sh` script prompts you for a confirmation. Enter yes to disable the SSL communication.
5. Select option **13: Go back to main menu** option, then select option **6: Restart OPSA server** option to restart the OBA Application Server.

### How to disable the SSL configuration for the OBA collector

To disable the SSL communication, do the following:

1. Unregister the collector from the application server. Run the following command as the "opsa" user:

### Unregister the collections and collectors

If OBA is already configured with collections (the collector is already registered) you must unregister all collections either by using the OBA user interface or the command line. To unregister collections, do the following:

**Note:** The procedure below shows the command line procedure, but if one collection collects many elements, it is easier to delete the collection in the user interface. Go to the **Source Type Manager**, select the collection, and click **Delete**.

- a. Unregister the collector:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -username <tenant admin username> -password <tenant admin
password>
```

If the collector is successfully unregistered, continue with Step e: "Check that the collector was unregistered successfully".

If there are any collectors still registered, continue with Step b: "Run the following command to list the collections".

- b. Run the following command to list the collections:

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

- c. Unregister the collections that were listed with the previous command:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -source <source> -domain <domain> -group <group> -username
<tenant admin user name> [-password <tenant admin password>]
```

- d. Unregister the collector:

```
/opt/HP/opsa/bin/opsa-collection-config.sh -unregister -collectorhost
<collector FQDN> -username <tenant admin username> -password <tenant admin
password>
```

- e. Check that the collector was unregistered successfully:

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

2. Run the `$OPSA_HOME/bin/opsa-collector-manager.sh` script. See the *opsa-collector-manager.sh* reference page (or the Linux man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

3. Select option **1: Configure SSL**.
4. Select option **8: Enable/Disable SSL**.
5. The `$OPSA_HOME/bin/opsa-collector-manager.sh` script prompts you for a confirmation. Enter `yes` to disable the SSL communication.
6. Select option **13: Go back to main menu** option, then select the **Restart OPSA Collector** option to restart the OBA Collector host.
7. Run the following command as the "opsa" user to register the collector:

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

## Managing the keystore and truststore for the OBA application server and collector

### How to modify the OBA keystore and truststore password

To modify the Operations Bridge Analytics keystore and truststore password, do the following:

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

**Note:** This script must be run out of the `$OPSA_HOME` directory.

2. Select the **Configure SSL** option.
3. Select the **Modify OPSA keystore/truststore password** option.

4. The 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** or the **Restart OPSA collector** option to restart the application server or collector.

## How to delete a certificate in the OBA keystore and truststore

To delete a certificate from the Operations Bridge Analytics keystore and truststore, do the following:

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

**Note:** This script must be run out of the `$OPSA_HOME` directory.

2. Select the **Configure SSL** option.
3. Select the **Delete certificate from OPSA server keystore** or **Delete certificate from OPSA server truststore** option for the server, or the **Delete certificate from OPSA keystore** or **Delete certificate from OPSA truststore** option for the collector.
4. The 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 OBA Application Server or the **Restart OPSA collector** option to restart the collector.

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

## How to export a certificate from the OBA keystore

To export a certificate from the Operations Bridge Analytics keystore, do the following:

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

**Note:** This script must be run out of the `$OPSA_HOME` directory.

2. Select the **Configure SSL** option.
3. Select the **Export certificate from OPSA server keystore** option.
4. The script prompts you for the alias name, listing the existing set of aliases from the Operations Bridge Analytics keystore. Enter the alias name to be deleted from the list.
5. The script prompts you for the file path to which the certificate should be exported. Enter the path to export the certificate.

## How to change an OBA keystore file

To change an Operations Bridge Analytics 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 man page) for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

2. Select the **Configure SSL** option.
3. Select the **Change OPSA keystore file** option.
4. The script prompts you with a set of prerequisite actions to take before proceeding. Enter yes after you complete these prerequisites.
5. Enter the absolute path of the keystore file.

## How to change an OBA truststore file

To change an Operations Bridge Analytics 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 man page) for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

2. Select the **Configure SSL** option.

3. Select the **Change OPSA truststore file** option.
4. The script prompts you with a set of prerequisite actions to take before proceeding. Enter yes after you complete these prerequisites.
5. Enter the absolute path of the truststore file.

## Two-Way SSL for accessing ArcSight Logger

Complete the following steps to configure two-way SSL authentication with ArcSight Logger:

1. Create an SSL truststore on the OBA Application Server with ArcSight Logger's server certificate:
  - a. Copy the self-signed or CA certificate from ArcSight Logger. You will find the self-signed certificate in the following location:  
`<Install_Dir>//userdata/platform/ssl.crt/server.crt`
  - b. Run the `opsa-server-manager.sh` script as the root user.

**Note:** Running the `opsa-server-manager.sh` script could result in a message similar to the following:

```
com.hp.opsa.server.admin.ssl.config.OPSCertStoreException:
please run this procedure manually with root credentials:
```

**If you see this message, there will be no residual effect. The remedy is to complete only one of the following actions:**

- Run the `opsa-server-manager.sh` script as root and complete the steps shown below.
- Do not run the `opsa-server-manager.sh` script. Instead, run the following command manually using root credentials:  

```
sudo keytool -import -trustcacerts -alias CN=HPQ Issuing Certification
Authority 2016-1, DC=americas, DC=cpqcorp, DC=net -keystore
/opt/HP/opsa/jdk/jre/lib/security/cacerts -file /home/opsa/HPQ Issuing
Certification Authority 2016-1.pem -storepass changeit
```

- i. Log on as the `opsaadmin` user.
- ii. Choose **Option 2** to configure SSL.
- iii. Choose **Option 4** to import the trusted certificate into the OpsA truststore.

- iv. Enter the file name of the certificate you want to import; then press **Enter**.
  - v. Repeat the prior steps for additional certificate files you want to import.
  - vi. Exit the `opsa-server-manager.sh` script.
2. Create a self-signed certificate and a keystore using OpenSSL for the OBA Application Server:
    - 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 ArcSight Logger to enable client authentication:
    - a. Copy the OBA Application Server's self-signed certificate from the following location:
 

```
$OPSA_HOME/conf/opsa.crt
```

 to this location on the ArcSight Logger server:
 

```
<Install_Dir>/current/local/apache/conf/ssl.crt
```
    - b. Edit ArcSight Logger's web server configuration file:
 

```
<Install_Dir>/current/local/apache/conf/httpd.conf
```
    - c. If the following lines do not exist in the file, add them, then save your work:
 

```
SSLVerifyClient require
SSLVerifyDepth 0
SSLCACertificateFile <Install_
```

- ```
Dir>/current/local/apache/conf/ssl.crt/opsa.crt
```
- d. Run the following command as the root user to restart ArcSight Logger's web server:
 

```
<Install_Dir>/current/arcsight/service/apache restart
```
  4. Configure the OBA Application Server's configuration file:
    - a. Edit the following file: `$OPSA_HOME/conf/opsa-config.properties`
    - b. Add the following line, then 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="$JAVA_OPTS -Djavax.net.ssl.trustStore=/opt/HP/opsa/conf/ssl/opsa-truststore.jks"
JAVA_OPTS="$JAVA_OPTS -Djavax.net.ssl.trustStorePassword=<password of trust store>"
JAVA_OPTS="$JAVA_OPTS -Djavax.net.ssl.keyStore=/opt/HP/opsa/conf/ssl/opsa-keystore.jks"
JAVA_OPTS="$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
```

## SSL for communication between Vertica and OBA

The information in this section explains how to manage SSL communications between Operations Bridge Analytics and the Vertica (Operations Bridge Analytics) database.



## Enabling SSL Communications between the OBA Application Server and Vertica

Complete the following steps from the server that contains the Vertica database to enable SSL communications between the OBA Application Server and the Vertica (Operations Bridge Analytics) database:

- "1 — Create the certificates" below
- "2 — Configure Vertica" on the next page
- "3 — Configure the OBA application server" on page 83
- "4 — Configure the OBA collectors" on page 85

### 1 — Create the certificates

To create the certificates, complete one of the following options on the Vertica node:

- **Option 1:** Self-signed certificate

Execute the following commands to generate a self-signed certificate:

- a. Generate the CA files `servercakey.pem` and the trusted root certificate `root.crt`.

```
openssl genrsa -out servercakey.pem 2048
openssl req -new -x509 -key servercakey.pem -out root.crt
```

- b. Create the server private key and the certificate signing request:

```
openssl genrsa -out server.key 2048
openssl req -new -key server.key -out server_reqout.txt
```

- c. Create the self-signed server certificate by using the signing request:

```
openssl x509 -req -in server_reqout.txt -days 3650 \
-sha1 -CAcreateserial -CA root.crt \
-CAkey servercakey.pem -out server.crt
```

- **Option 2:** Certificate Authority (CA) signed certificate

Execute the following commands to generate a CA signed certificate:

- a. Create the server private key and the certificate signing request:

```
openssl genrsa -out server.key 2048
```

```
openssl req -new -key server.key -out server_reqout.txt
```

- b. Get the signed certificate and the trusted root certificate from your Certificate Authority (CA).

## 2 — Configure Vertica

On the Vertica node, you should have three files:

- `server.key`: The Vertica server private key
- `server.crt`: The signed Vertica server certificate
- `root.crt`: The trusted root certificate

**Note:** All files must have the file permissions '700'.

Configure Vertica as follows:

1. Copy the `root.crt` to `$VSQL_HOME` or, if `$VSQL_HOME` is not set, to the directory `/home/dbadmin`:

```
cp root.crt /home/dbadmin/
chown dbadmin:verticadba /home/dbadmin/root.crt
chmod 700 /home/dbadmin/root.crt
```

2. Distribute the certificate files:

- a. Execute the `adminTools` as `dbadmin` user.
- b. In the Main Menu, select **Configuration Menu > Distribute Config Files > SSKeys > opsadb**. The directories containing the certificate files are displayed.
- c. In another console, copy the listed files into the listed directories. For example:
 

```
cp root.crt server.crt server.key \
/opt/vertica/opsa_data/opsadb/v_opsadb_node0002_catalog/

chown dbadmin:verticadba \
/opt/vertica/opsa_data/opsadb/v_opsadb_node0002_catalog/root.crt

chown dbadmin:verticadba \
/opt/vertica/opsa_data/opsadb/v_opsadb_node0002_catalog/server.crt

chown dbadmin:verticadba \
/opt/vertica/opsa_data/opsadb/v_opsadb_node0002_catalog/server.key
```
- d. In the admin tool where the certificate files are listed, click **OK**, and close the admin tool.

3. Enable SSL in the OBA database as `dbadmin` user. Use `vsq1` to alter the `opsadb` database:

```
ALTER DATABASE opsadb SET EnableSSL = 1;
```

Make sure that the SSL Mutual Mode is not enabled. This mode might have been enabled automatically during the distribution of the certificate files. To disable SSL Mutual Mode, clear the SSL CA:

```
ALTER DATABASE opsadb CLEAR SSLCA;
```

Note that this command may return an error when the parameter was not set. You can safely ignore this.

4. Copy the root certificate to the OBA application server and the OBA collectors:

```
scp root.crt opsa@oba.system.net:/home/opsa/
scp root.crt opsa@obac.system.net:/home/opsa/
```

5. Restart the database by using the admin tool.

### 3 — Configure the OBA application server

Complete the following steps on the OBA application server to enable SSL:

1. Convert the trusted root certificate to a format that is understood by Java:

```
openssl x509 -in root.crt -out root.crt.der -outform der
```

2. Add the certificate to the truststore:

```
keytool -importcert \
  -file ~/root.crt.der \
  -alias opsa_server \
  -keystore /opt/HP/opsa/conf/ssl/opsa-truststore.jks \
  -storepass keystore_neutron_analytics_bigdata_opsa_2013 \
  -storetype jks
```

3. As root user, add the certificate to the OBA-JRE truststore:

```
sudo keytool -importcert \
  -file ~/root.crt.der \
  -alias opsa_server \
  -keystore /opt/HP/opsa/jdk/jre/lib/security/cacerts \
  -storepass changeit
```

4. Edit the \$OPSA\_HOME/conf/opsa-config.properties file.

Make sure that `vertica.ssl.enabled` is set to `true`:

```
vertica.ssl.enabled=true
```

**Note:** If the setting does not exist in that file, add it.

5. Edit the \$OPSA\_HOME/jboss/standalone/configuration/standalone.xml file:

Search for <connection-url>jdbc:vertica: and add a new <connection-property> below the <connection-url>...</connection-url> line:

```
<connection-property name="ssl">true</connection-property>
```

The file should then look like this:

```
...
<connection-url>jdbc:vertica://FQDN of Vertica Server:5433/opsadb</connection-
url>
<connection-property name="ssl">true</connection-property>
<driver>vertica</driver>
...
```

6. 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 and locate the section containing the <system-properties>
  - b. Uncomment the two settings javax.net.ssl.trustStorePassword and javax.net.ssl.trustStore

The <system-properties> section should then look like this:

```
<system-properties>
<property name="org.apache.coyote.http11.Http11Protocol.MAX_HEADER_SIZE"
value="1048576"/>
<property name="javax.net.ssl.trustStorePassword"
value="<truststore_password>"/>
<property name="javax.net.ssl.trustStore"
value="/opt/HP/opsa/conf/ssl/opsa-truststore.jks"/>
<property name="org.apache.coyote.http11.Http11Protocol.SERVER"
value="SECRET"/>
<property name="org.apache.coyote.http11.Http11Protocol.COMPRESSION"
value="on"/>
<property name="org.apache.coyote.http11.Http11Protocol.COMPRESSION_MIME_
TYPES"
value="text/javascript,text/css,text/html,text/xml,text/json,application/xml
,application/</system-properties>
```

If the default password of the OBA truststore was not changed, the <truststore\_password> is `keystore_neutron_analytics_bigdata_opsa_2013`.

7. Restart the OBA application server:

```
opsa restart
```

## 4 — Configure the OBA collectors

Do the following on all OBA collector nodes:

1. Convert the trusted root certificate to a format that is understood by Java:

```
openssl x509 -in root.crt -out root.crt.der -outform der
```

2. Add the certificate to the truststore:

```
keytool -importcert \  
-file ~/root.crt.der \  
-alias opsa_server \  
-keystore /opt/HP/opsa/conf/ssl/opsa-truststore.jks \  
-storepass keystore_neutron_analytics_bigdata_opsa_2013 \  
-storetype jks
```

- a. Edit the `$OPSA_HOME/conf/opsa-config.properties` file.

Make sure that `vertica.ssl.enabled` is set to `true`:

```
vertica.ssl.enabled=true
```

**Note:** If the setting does not exist in that file, add it.

3. Restart the OBA collector hosts:

```
opsa restart
```

## Disabling SSL Communications between the OBA Application Server and Vertica

Complete the following steps from the server that contains the Vertica database to disable SSL communications between the OBA Application Server and the Vertica (Operations Bridge Analytics) database:

1. Enable SSL by altering the database using `vsql`:

```
ALTER DATABASE opsadb SET EnableSSL = 0;
```

2. Restart the database by using the admin tool.

## Enabling SSL Communications between the Operations Bridge Analytics Collector Host and Vertica

Complete the following steps on the OBA Collector host to enable SSL between the OBA Collector host and the Vertica (Operations Bridge Analytics) database:

1. Complete steps 1-8 in ["Enabling SSL Communications between the OBA Application Server and Vertica" on page 81](#) 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 OBA Collector host and the Vertica (Operations Bridge Analytics) database is now enabled.

## Disabling SSL Communications between the Operations Bridge Analytics Collector Host and Vertica

Complete the following steps on the OBA Collector host to disable SSL between the OBA Collector host and the Vertica (Operations Bridge Analytics) database:

1. Complete the steps shown in ["Disabling SSL Communications between the OBA Application Server and Vertica" on the previous page](#) 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 OBA Collector host and the Vertica (Operations Bridge Analytics) database is now disabled.

## Adjusting Operations Bridge Analytics for RC4 Cipher Security Changes

Do the following on OBA Application Server and Collector hosts if you configured Operations Bridge Analytics to work over a secure channel (https):

1. Edit the `$OPSA_HOME/jboss/standalone/configuration/standalone.xml` file.
2. Locate the following section in the `standalone.xml` file:

```
<!--connector name="https" protocol="HTTP/1.1" scheme="https" secure="true"
socket-binding="https">
<ssl ca-certificate-file="/opt/HP/opsa/conf/ssl/opsa-truststore.jks"
certificate-key-file="/opt/HP/opsa/conf/ssl/opsa-keystore.jks" key-alias="opsa_
server" name="ssl" password="${VAULT:ks::pwd:${Password Key}}"
protocol="TLSv1,TLSv1.1,TLSv1.2" verify-client="false"/>
```

Uncomment this block as show below:

```
<connector name="https" protocol="HTTP/1.1" scheme="https" secure="true"
socket-binding="https">
<ssl ca-certificate-file="/opt/HP/opsa/conf/ssl/opsa-truststore.jks"
certificate-key-file="/opt/HP/opsa/conf/ssl/opsa-keystore.jks" key-alias="opsa_
server" name="ssl" password="${VAULT:ks::pwd:${Password Key}}"
protocol="TLSv1,TLSv1.1,TLSv1.2" verify-client="false"/>
```

3. Save your work.
4. 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`

**Note:** After running an `opsa-server restart` command, information logged in the `/opt/HP/opsa/log/opsa-server.log` file is removed. To view the files that retain this information across restarts, see the log files located in the `/opt/HP/opsa/jboss/standalone/log` directory. Look for log file names that begin with `server.log`.

5. Make the following modifications to the `java.security` file.

**Note:** HPE recommends that you back up the existing `java.java.security` file before making these modifications.

- a. Navigate to the `/opt/HP/opsa/jdk/jre/lib/security` directory.
- b. Using a text editor, open the `java.security` file that resides in the directory to which you just navigated.
- c. Locate the `jdk.tls.disabledAlgorithms` line located towards the bottom of this file. Replace the text in that line with the following text. If the line does not exist, add the following line:

`"jdk.tls.disabledAlgorithms=MD5, SSLv3, RC4, DSA"`

- d. Save your work.

## SSL for the SMTP Server used for OBA Alerts

The information in this section explains how to manage SSL communications to your SMTP server.

1. Verify that OBA application servers are already communicating using SSL.
2. Copy the SMTP's root server certificate to the OBA application servers and give the file full permissions.
3. Do the following to import the SMTP's root server certificate into the Operations Bridge Analytics truststore:
  - a. Run the `opsa-server-manager.sh` script.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

- b. Log on to the OBA Application Server as the `opsaadmin` user.



- c. Choose **Option 2** to configure SSL.
- d. Choose **Option 4** to import the trusted certificate into the OBA truststore.
- e. Enter the file name of the certificate you want to import; then press **Enter**.
- f. Repeat the prior steps for additional certificate files you want to import.
- g. Exit the `opsa-server-manager.sh` script.



# HTTP and HTTPS

This section provides information about configuring HTTP and HTTPS for the OBA collector host.

This section includes:

- ["Configure the HTTP and HTTPS port for the OBA collector host" below](#)
- ["Configure the HTTP and HTTPS user name and password for the OBA collector host" on the next page](#)

## Configure the HTTP and HTTPS port for the OBA collector host

The OBA Collector host comes with a pre-configured HTTP and HTTPS port of 9443. If you run into any conflicts with port 9443, the value can be changed.

To change the HTTPS port to which the OBA Collector host 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 man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

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 OBA Collector host on the OBA Application Server 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 man page) for more information.

After you complete this step, future communication to this OBA Collector host uses the new HTTPS port.

## Configure the HTTP and HTTPS user name and password for the OBA collector host

The OBA Collector host comes with a pre-configured HTTPS user name, **opsa**, having an identical password, **opsa**. HPE recommends 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 man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

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 OBA Collector host on the OBA Application Server 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 man page) for more information.

After you complete this step, future access to this OBA Collector host uses the new username and password values.

# Single Sign-On

This section provides information about how to configure, enable, and disable Lightweight Single Sign-On (LW-SSO) for OBA.

This section includes the following topics:

- ["Configure and enable single sign-on to access OBA" below](#)
- ["Disable SSO access to OBA" on page 95](#)

## Configure and enable single sign-on to access OBA

These instructions provide a practical example of configuring and enabling LWSSO between Operations Bridge Analytics and BSM or OMi. Use this practical example to help you configure LWSSO between Operations Bridge Analytics and other applications you plan to use.

Enabling Single Sign-on (LWSSO) in Operations Bridge Analytics permits users to launch the Operations Analytics console from a BSM or OMi event browser without needing to log on again. LWSSO is not enabled by default.

When using Single Sign-on, consider the following:

- Both systems must be configured for http or both systems must be configured for https. A mixture of these two protocols is not supported.
- Single Sign-on does not work if you use the OBA Application Server's IP address or short hostname. When using Single Sign-on, you must use the fully-qualified domain name of the OBA Application Server in the URL.

The below procedure must be performed on all OBA application servers.

**Note:** For this example, the user accounts for the BSM or OMi server and the OBA Application Server must match for these instructions to work correctly. The user name is case sensitive, so the user name used in each application must be identical.

1. Before enabling LWSSO to the OBA Application Server, complete the following steps to create a user in JBoss **Management Realm**:

- a. As the `opsa` user, 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. As the **opsa** user, run the `$OPSA_HOME/bin/opsa-server-manager.sh` script. See the *opsa-server-manager.sh* reference page (or the Linux man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

3. Select option **3: Configure LWSSO**.
4. Select option **1: Configure LWSSO parameters**.
5. When prompted with **Enter the Token Creation Key (initString) [xxxxxxx]**, enter the `initString` key. For example, if you are configuring LWSSO for Operations Bridge Analytics and BSM or OMi, the value must match the `initString` configured in BSM or OMi.

**Note:** To view the `initString` configured in BSM or OMi, log on to BSM or OMi and navigate to:

BSM: **BSM > Admin > Platform > Users and Permissions > Authentication Management**

OMi: **Administration > Users > Authentication Management**

It is important to use the exact `initString` configured in BSM or OMi for this example. It is also important to use the exact `initString` with other applications you plan to use with Operations Bridge Analytics.

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 domain of the OBA Application Server.
8. If the BSM or OMi system is in a different domain, enter the trusted domain names when prompted with **Enter trusted domains separated by comma**. If they are in the same domain, this field can be left blank. Use the following form:

`mytrusteddomain1.com, mytrusteddomain2.com`

**Note:** You must include the domain for the BSM or OMi server, considering the BSM or OMi

example being shown in these steps. This is even more important if the domain is not in the same domain in which the OBA Application Server resides.

9. If BSM or OMi and OBA are in different domains, add them to the OMi Single Sign-On Configuration.
  - a. On the OMi server go to **Administration > Users > Authentication Management**. On a BSM server, go to **BSM > Admin > Platform > Users and Permissions > Authentication Management**. In the Single Sign-On Configuration section click **Edit**.
  - b. Select Lightweight, then select Domain. Enter the domain for BSM or OMi.
  - c. Repeat steps a and b for OBA to create a second entry for the OBA domain.
10. After the `opsa-server-manager.sh` script finishes configuring LWSSO, it displays a **Configured LWSSO successfully** message, and gives you three options. 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.
11. Exit the tool and restart the OBA processes as the root user:

```
sudo /opt/HP/opsa/bin/opsa restart
```

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

After completing the steps in this section, and configuring the correct URL on BSM or OMi, you can launch the Operations Analytics console from a BSM or 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.

## Disable SSO access to OBA

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 man page), for more information.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

2. Select the **Configure LWSSO** option.
3. Select the option **Enable/Disable LWSSO** to disable LWSSO.
4. Exit the tool and restart the OBA processes as the root user:

```
/opt/HP/opsa/bin/opsa restart
```

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



## Configure LDAP authentication

OBA supports Lightweight Directory Access Protocol (LDAP) for user authentication. This section explains how to configure OBA to connect to an LDAP server to validate OBA users. Only a Super Admin user, `opsaadmin`, can configure OBA to authenticate users through an LDAP server.

If group mapping is configured, OBA will automatically create users based on the group members in the LDAP groups. If group mapping is not configured, the tenant admin must manually create LDAP authenticated users in OBA.

When configuring this LDAP authentication, you can also enable SSL for LDAP server authentication. To do this, complete the steps shown in ["Configuring SSL for LDAP server authentication" on page 100](#) before continuing.

You can either use the `opsa-ldap-configuration-manager.sh` command line tool or the OBA user interface to configure LDAP authentication in OBA:

- ["Configure LDAP authentication for OBA with the user interface" below](#)
- ["Configure LDAP authentication for OBA using the command line" on the next page](#)

### Configure LDAP authentication for OBA with the user interface

You can either configure LDAP authentication with group mapping, whereby users are created automatically in the OBA user store, or without group mapping, in which case you must manually add LDAP authenticated users. Choose one of the following:

- ["Configure LDAP authentication with group mapping with the user interface" below](#)
- ["Configure LDAP authentication without group mapping with the user interface" on the next page](#)

### Configure LDAP authentication with group mapping with the user interface

To configure LDAP authentication with group mapping via the user interface, do the following:

1. Log in to OBA as the `opsaadmin` user.
2. Go to **Settings > LDAP Servers**.
3. Click **Add** to add a new LDAP server.

4. To add the group mapping, log in to OBA as the opsatenantadmin user.
5. Go to **Settings > LDAP Group Mapping**.
6. Specify the LDAP Admin and User groups. Members of LDAP Admin groups will automatically be created as tenant admin users. Members of LDAP User groups will be created as basic users in OBA.

## Configure LDAP authentication without group mapping with the user interface

To configure LDAP authentication without group mapping via the user interface, do the following:

1. Log in to OBA as the opsaadmin user.
2. Go to **Settings > LDAP Servers**.
3. Click **Add** to add a new LDAP server.
4. Log in to OBA as the opsatenantadmin user.
5. Go to **Settings > User Manager**.
6. Click **Add User** to add your LDAP users. Make sure that **LDAP authenticated user** is selected.

## Configure LDAP authentication for OBA using the command line

You can either configure LDAP authentication with group mapping, whereby users are created automatically in the OBA user store, or without group mapping, in which case you must manually add LDAP authenticated users. Choose one of the following:

- ["Configure LDAP authentication with group mapping using the command line" below](#)
- ["Configure LDAP authentication without group mapping using the command line" on the next page](#)

## Configure LDAP authentication with group mapping using the command line

To configure LDAP authentication with group mapping via the command line, do the following:

1. Run the following command to save the LDAP server configuration information to Operations Bridge Analytics:

```
$OPSA_HOME/bin/opsa-ldap-configuration-manager.sh add --username opsaadmin --
password <opsa_superadmin_password> --ldapusername <ldap_username> --
ldappassword <ldap_password> --ldaphostname <ldap_hostname>
--ldapbasedn <ldap_basedn> --ldapport <port> --userdn <userdn> --ssl <true |
false> -groupbasedn <ldap_group_basedn> --groupAttribute <group_attribute> --
userAttribute <user_attribute>
```

**Note:** Only use the ssl option if you completed the steps shown in ["Configuring SSL for LDAP server authentication" on the next page](#)

The add option is used to add the LDAP server configuration information to Operations Bridge Analytics. All of the Operations Bridge Analytics users are authenticated by communicating to this LDAP server based on the additional configuration input. For example, notice the ldap-basedn and userdn attributes used in this example.

If you do not specify the optional LDAP integration username and LDAP integration user password during this LDAP configuration, anonymous binding must be enabled on the LDAP Servers.

To configure grouping mapping, you must set the following attributes: groupbasedn, groupAttribute, and userAttribute.

2. Run the following command to check that the LDAP information you added to Operations Bridge Analytics is accurate:

```
$OPSA_HOME/bin/opsa-ldap-configuration-manager.sh list --username opsaadmin --
password <opsa_superadmin_password>
```

3. Run the following command to set up the LDAP group mapping:

```
$OPSA_HOME/bin/opsa-ldap-group-mapping-manager.sh add --username tenantadmin --
password <opsa_tenantadmin_password> -ldapUserGroup <ldap_user_group> --
ldapAdminGroup <ldap_admin_group>
```

Members of LDAP Admin groups will automatically be created as tenant admin users. Members of LDAP User groups will be created as basic users in OBA.

4. To view the added mapping information, run the following command:

```
$OPSA_HOME/bin/opsa-ldap-group-mapping-manager.sh view --username tenantadmin -
-password <opsa_tenantadmin_password>
```

## Configure LDAP authentication without group mapping using the command line

To configure LDAP authentication without group mapping via the command line, do the following:

1. Run the following command to save the LDAP server configuration information to Operations Bridge Analytics:

```
$OPSA_HOME/bin/opsa-ldap-configuration-manager.sh add --username opsaadmin --
password <opsa_superadmin_password> --ldapusername <ldap_username> --
ldappassword <ldap_password> --ldaphostname <ldap_hostname>
--ldapbasedn <ldap_basedn> --ldapport <port> --userdn <userdn> --ssl <true |
false>
```

**Note:** Only use the ssl option if you completed the steps shown in ["Configuring SSL for LDAP server authentication" below](#)

The add option is used to add the LDAP server configuration information to Operations Bridge Analytics.

If you do not specify the optional LDAP integration username and LDAP integration user password during this LDAP configuration, anonymous binding must be enabled on the LDAP Servers.

2. Run the following command to check that the LDAP information you added to Operations Bridge Analytics is accurate:

```
$OPSA_HOME/bin/opsa-ldap-configuration-manager.sh list view --username
opsaadmin --password <opsa_superadmin_password>
```

3. Manually create users with the opsa-user-manager.sh script, making sure to specify the -ldapAuthenticated option:

```
$OPSA_HOME/bin/opsa-user-manager.sh -add -loginuser [<super_admin_user> |
<tenant_admin_user>] -loginPassword [<super_admin_password> | <tenant_admin_
password>] -newUser <username> -ldapAuthenticated -tenant <tenant>
```

Alternatively, LDAP users can be added using a text file with the names of the users. To use this option, use the -batchUpload command as follows:

```
opsa-user-manager.sh -batchUpload [file path] -loginUser tenant_admin_user -
loginPassword <tenant_admin_password>
```

The text file should contain the usernames listed in different rows.

## Configuring SSL for LDAP server authentication

When configuring this LDAP authentication, you can also enable SSL for LDAP server authentication. You can configure LDAP server authentication using one of two methods:

You must complete the instructions in this section before configuring LDAP server authentication.

To configure SSL for LDAP server authentication, do the following:

1. Copy the LDAP's root server certificate to the OBA application servers and give the file full permissions.
2. Run the `opsa-server-manager.sh` script.

**Note:** This script must be run out of the `$OPSA_HOME` directory.

- a. Log on as the `opsaadmin` user.
- b. Choose **Option 2** to configure SSL.
- c. Choose **Option 4** to import the trusted certificate into the OBA truststore.
- d. Enter the file name of the certificate you want to import; then press **Enter**.
- e. Repeat the prior steps for additional certificate files you want to import.
- f. Exit the `opsa-server-manager.sh` script.
- g. Restart OBA processes as the root user:

```
/opt/HP/opsa/bin/opsa restart
```

Now that you enabled SSL for LDAP server authentication, continue configuring LDAP server authentication using one of the methods shown at the beginning of this section. You can now select the option to enable SSL for LDAP server authentication.

# PKI

SSL Client Certificate authentication using Public Key Infrastructure (PKI) enables users to log on to the Operations Analytics console with a client-side X.509 certificate. The following sections provide information about how to configure, disable, and edit user authentication by using PKI.

- ["Configure user authentication using PKI to access OBA" below](#)
- ["Disable user authentication using PKI to access OBA" on page 105](#)
- ["Edit user authentication using PKI to access OBA" on page 105](#)

## Configure user authentication using PKI to access OBA

SSL Client Certificate authentication using Public Key Infrastructure (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 OBA Application Server 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 OBA Application Server, complete the following steps to create a user in JBoss **Management Realm**:
  - 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 man page), for more

information.

**Note:** This script must be run out of the \$OPSA\_HOME directory.

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 OBA Application Server keystore.
  - **CA Signed Certificate:** Select the **Import CA certificate to OPSA server keystore** option to import a CA signed certificate to the OBA Application Server 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 Operations Bridge Analytics to use.
9. When prompted for **Check for certificate revocation [yes/no]**, enter `yes` for Operations Bridge Analytics 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 Operations Bridge Analytics 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` script 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 OBA Application Server.

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

After completing the above steps, Operations Bridge Analytics users can access the Operations Analytics console using HTTP or HTTPS as follows:

See the `opsa-server-manager.sh` reference page (or the Linux man page) for more information.

1. If an Operations Bridge Analytics user enters an HTTP URL, Operations Bridge Analytics automatically redirects the URL to HTTPS, and shows a **Login with digital certificate** button.
2. After clicking the **Login with digital certificate** button, Operations Bridge Analytics presents its digital certificate, and the browser verifies it against its truststore.
3. After verifying the Operations Bridge Analytics certificate, Operations Bridge Analytics prompts the user to select the client certificate. On selecting the client certificate, Operations Bridge Analytics 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 Operations Bridge Analytics home page.



## Disable user authentication using PKI to access OBA

To disable PKI authentication, do the following:

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux man page), for more information.  
**Note:** This script must be run out of the `$OPSA_HOME` directory.
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 OBA Application Server.

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

After completing the above steps, Operations Bridge Analytics presents its users with a user name and password page to access the Operations Analytics console.

## Edit user authentication using PKI to access OBA

To modify PKI authentication, do the following:

1. Run the `$OPSA_HOME/bin/opsa-server-manager.sh` script and log on with **super-admin** credentials. See the *opsa-server-manager.sh* reference page (or the Linux man page), for more information.  
**Note:** This script must be run out of the `$OPSA_HOME` directory.
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 ["Configure user authentication using PKI to access OBA" on page 102](#).
5. Select the **Go back to main menu** option; then select the **Restart OPSA server** option to restart the OBA Application Server.

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

## Resetting user passwords

By default, users are prompted to select new passwords every 182 days. The number of days between resets can be modified by an administrator.

**Note:** If you are using LDAP to authenticate Operations Bridge Analytics users, do not use the information in this section when resetting passwords for LDAP authenticated users.

To modify the password reset time:

1. Go to `/opt/HP/opsa/conf/opsa-config.properties`
2. Modify the `password.expiration.period.days` property to the desired value.

## Change the password of a collector host

After registering an OBA Collector host you might need to change its password because of your security policy or for other reasons. You might have Source Data from Source Types that you do not want to lose.

To safely change the password of a registered OBA Collector host, do the following:

1. Open the RTSM JMX console by using the following URL:  
`http://<fully qualified domain name of the Collector Host>:29900/mbean?objectname=com.hp.opsa.collector.http.server%3Aname%3DCollectorRestServer`
2. From the `CollectorRestServer` interface invoke `changeCollectorPassword`.
3. Type the new password twice in the parameter form fields.

## Changing the port used by the OBA console

There might be a need to change the port used by the Operations Analytics console to comply with local security policies. Note, however, that ports below 1024 are privileged ports and Linux prevents applications from using these ports.

To change the port used by the OBA console, do the following:

1. As the `opsa` user, edit the `$JBOSS_HOME/standalone/configuration/standalone.xml` file.
2. Search for the **standard-sockets** stanza and change the **http** port, **https** port, or both to the ports you want to use as shown in the following example (the items to search for are in bold font):

```
<socket-binding-group name="standard-sockets" default-interface="public"
port-offset="${jboss.socket.binding.port-offset:0}">
  <socket-binding name="management-native" interface="management"
port="${jboss.management.native.port:19999}"/>
  <socket-binding name="management-http" interface="management"
port="${jboss.management.http.port:9990}"/>
  <socket-binding name="management-https" interface="management"
port="${jboss.management.https.port:9991}"/>
  <socket-binding name="ajp" port="8009"/>
  <socket-binding name="http" port="8080"/>
</socket-binding-group>
```

```
<socket-binding name="https" port="8443"/>
<socket-binding name="remoting" port="4447"/>
<socket-binding name="txn-recovery-environment" port="4712"/>
<socket-binding name="txn-status-manager" port="4713"/>
<outbound-socket-binding name="mail-smtp">
  <remote-destination host="localhost" port="25"/>
</outbound-socket-binding>
</socket-binding-group>
```

3. Save your work.
4. Run the `$OPSA_HOME/bin/opsa-server restart` command to restart the OBA Application Server.

**Note:** If you have multiple tenants, HPE recommends using different OBA Collector hosts for each tenant. Doing so ensures data separation for each tenant.

# 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 Administration Guide (Operations Bridge Analytics 3.03)**

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 [ovdoc-asm@hpe.com](mailto:ovdoc-asm@hpe.com).

We appreciate your feedback!