

---

# HP OSS Analytics Foundation

## Version 1.1.1



### **Installation, Configuration, and Administration Guide**

**Edition: 1.0**

**For Linux, RHEL 6.5**

**October 2015**

## Legal notices

### Warranty

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

### License requirement and U.S. Government legend

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

### Copyright notices

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

### Trademark notices

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

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

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

HP Vertica™, the HP Vertica Analytics Platform™ are trademarks of Hewlett-Packard

Firefox® is a registered trademark of the Mozilla Foundation.

Google Chrome® is a trademark of Google Inc.

UNIX® is a registered trademark of The Open Group.

Red Hat® is a registered trademark of the Red Hat Company.

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

JBoss®, Wildfly and PicketLink are registered trademarks of RedHat Inc.

# Contents

<b>Preface .....</b>	<b>7</b>
<b>Chapter 1.....</b>	<b>9</b>
<b>Product overview .....</b>	<b>9</b>
1.1 HP OSS Analytics Server introduction.....	9
1.2 HP OSS Analytics Server architecture .....	11
<b>Chapter 2.....</b>	<b>14</b>
<b>Installation prerequisites .....</b>	<b>14</b>
2.1 Hardware and operating systems prerequisites .....	14
2.2 Software prerequisites .....	14
2.2.1 Java.....	14
2.2.2 Database.....	15
2.2.3 Vertica client.....	15
2.2.4 Timezone settings .....	15
2.3 Installation package.....	16
2.4 Code signing .....	16
2.4.1 Verifying an HP product signature using GPG .....	16
<b>Chapter 3.....</b>	<b>17</b>
<b>Product installation .....</b>	<b>17</b>
3.1 Creating the HP OSS Analytics Server Linux user .....	17
3.2 Installing the RPM kit .....	17
3.3 Creating the ossa database user and schema .....	17
3.4 Configuring the installation .....	18
3.5 Finalizing the installation .....	20
3.6 Validating the installation.....	20
<b>Chapter 4.....</b>	<b>22</b>
<b>Product configuration .....</b>	<b>22</b>
4.1 Configuring a dedicated Vertica resource pool.....	22
4.2 Configuring HP OSS Analytics Foundation as a Red Hat Linux service .....	22
4.3 Security .....	22
4.3.1 Securing the server .....	22
4.3.2 Activating SSL.....	22
4.3.3 Cryptographic algorithms .....	24
4.3.4 Enabling auditing.....	25
4.3.5 Authentication / Single Sign-On.....	25
4.3.6 Role based access control .....	25
4.3.7 Encryption .....	25
4.3.8 DEBUG traces logging.....	26
<b>Chapter 5.....</b>	<b>27</b>
<b>Product administration .....</b>	<b>27</b>

5.1	Starting the HP OSS Analytics Server .....	27
5.2	Stopping the HP OSS Analytics Server .....	27
5.3	Loading metadata .....	27
5.4	Loading a batch job .....	28
5.5	Loading batchs descriptions contained in a directory .....	29
5.6	Running a batch job.....	29
5.7	OSS Analytics Foundation administration console.....	29
5.7.1	Batch monitoring.....	30
5.7.2	Batch scheduling administration .....	31
<b>Chapter 6.....</b>		<b>32</b>
<b>Product troubleshooting .....</b>		<b>32</b>
6.1	Logging.....	32
6.2	Common errors .....	34
6.2.1	Deployment error about OssaDS .....	34
<b>Chapter 7.....</b>		<b>35</b>
<b>Uninstalling the HP OSS Analytics Server .....</b>		<b>35</b>
<b>Annex 1 Setting up the PicketLink identity provider .....</b>		<b>36</b>
	Prerequisite .....	36
	Configuring JBoss Wildfly.....	36
	Updating the PicketLink libraries .....	36
	Creating the security domain .....	36
	Deploying the identity provider .....	37
	Security.....	37
	Enabling signature support.....	37
	Enabling encryption support.....	38

# Figures

Figure 1 – HP OSS Analytics Foundation Server .....	11
Figure 2 – HP OSS Analytics REST Server architecture .....	12
Figure 3 – HP OSS Analytics Batch Server architecture .....	13
Figure 4 – HP OSSA Foundation administration console – Batch Monitoring.....	30
Figure 5 – HP OSSA Foundation administration console – Batch Administration .....	31
Figure 6 – HP OSSA Foundation administration console – Batch Scheduling .....	31

# Tables

Table 1 - Software versions .....	7
Table 2 – Minimum hardware requirements for HP OSS Analytics Server on <i>Linux</i> .....	14
Table 3 – ossa.cfg parameters description.....	19
Table 4 – HP OSSA SSL configuration parameters description .....	24

# Preface

This guide describes how to install, configure, administrate, and troubleshoot the HP OSS Analytics Foundation software component.

Software component name: HP OSS Analytics Foundation

Software component version: 1.1.1

Software kit version: V1.1.1

## Intended audience

- This installation and configuration guide is for anyone who is responsible for installing, uninstalling, configuring, administrating, or troubleshooting the HP OSS Analytics Foundation.
- The readers are assumed to understand Linux shell concepts.

## Software versions

The terms Unix and Linux are used as a generic reference to the operating system, unless otherwise specified.

The software versions referred to in this document are as follows:

**Table 1 - Software versions**

<b>Product version</b>	<b>Supported operating systems</b>
<i>HP OSS Analytics Foundation</i> version 1.1.1	<ul style="list-style-type: none"><li>• Red Hat Enterprise Linux Server release RHEL 6.5</li></ul>
<i>Wildfly</i> version 8.2.1 (formerly known as <i>JBoss</i> )	<ul style="list-style-type: none"><li>• Red Hat Enterprise Linux Server release RHEL 6.5</li></ul>
<i>HP Vertica</i> version 7.1	<ul style="list-style-type: none"><li>• Red Hat Enterprise Linux Server release RHEL 6.5</li></ul>

## Typographical conventions

Courier font:

- Source code and examples of file contents
- Commands that you enter on the screen
- Path names
- Keyboard key names

*Italic text:*

- File names, programs, and parameters
- The names of other documents referenced in this manual

**Bold text:**

- To introduce new terms and to emphasize important words

## Associated documents

- HP OSS Analytics Foundation Release Notes

## Support

Visit the HP Software Support Online website at <https://softwaresupport.hp.com/> for contact information, and for details about HP software products, services, and support.

The software support area of the website includes the following:

- Downloadable documentation
- Troubleshooting information
- Patches and updates
- Problem reporting
- Training information
- Support program information



## Product overview

### 1.1 HP OSS Analytics Server introduction

The HP OSS Analytics Foundation is part of the HP OSS Analytics program.

It provides a unified solution for external applications to access data and to perform analysis on data stored in data warehouses.

The HP OSS Analytics Foundation is composed of an OSS Analytics application server having two main functionalities:

- OSS Analytics REST server
- OSS Analytics Batch server

---

#### Note

The HP OSS Analytics Foundation v1.1 supports access to the *HP Vertica* database.

---

#### OSS Analytics REST server

The OSS Analytics REST server acts as an operational interface for the HP OSS Analytics solutions requiring analyzing data stored in data warehouses.

The character generic of the OSS Analytics REST server comes from the usage of “metadata”. Metadata provides an abstraction of the underlying database and allows the OSSA REST server to retrieve and compute data for any domain. The metadata describes the data warehouse schema as a set of associations of Facts to Dimensions, and describes the type of aggregations that are possible. The possible aggregation types are:

- Pre-calculated
- On-the-fly

Once the metadata file is loaded into the OSS Analytics REST server, the server is able to reply to Dimensions-and-Facts client requests in order to perform multidimensional analysis.

---

#### Note

HP OSS Analytics Foundation does not provide any metadata instantiation. See the HP OSS Analytics solutions offering for details.

---

#### OSS Analytics Batch server

The OSS Analytics Batch services aim at defining, scheduling, monitoring batch jobs for your OSS Analytics solution.

What is a “batch job” in the context of OSS Analytics?

This is a process that generally handle and transform large amount of data, and for which you want an automatic and periodical execution.

The OSS Analytics Batch framework allows you to define your batch jobs based on some generic pre-defined but customizable tasks. Those tasks allow you to perform: loading data to Vertica, SQL processing, summarization on data, or report execution and mailing.

For details about those tasks please refer to the *HP OSS Analytics Foundation Integration guide*.

Then, the OSS Analytics Batch services allow you to deploy, schedule and monitor your batch jobs.

---

**Note**

---

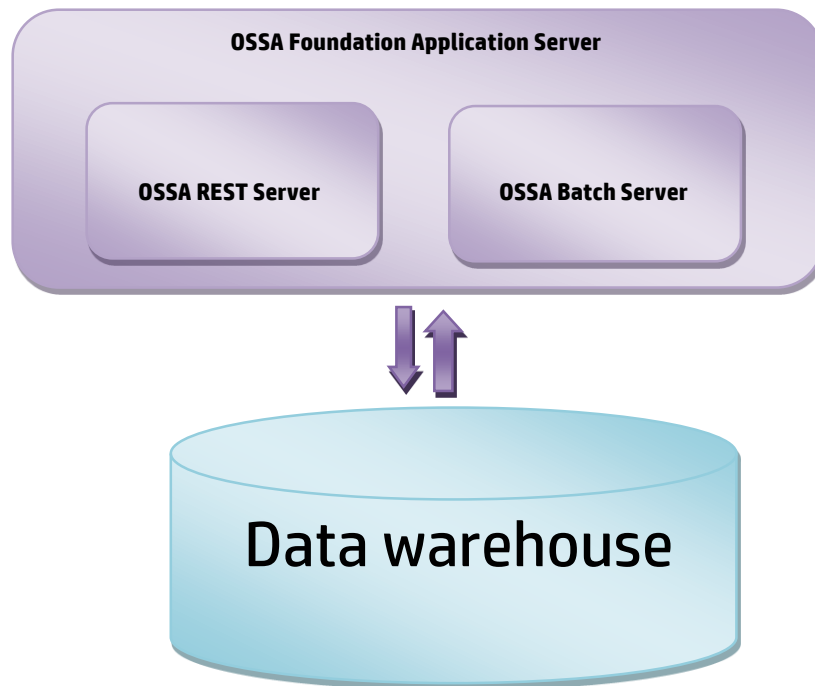
HP OSS Analytics Foundation does not provide any batch job instantiation. See the HP OSS Analytics solutions offering for details.

---

## 1.2 HP OSS Analytics Server architecture

The HP OSS Analytics Foundation is composed of an OSS Analytics application server containing:

- OSS Analytics REST server
- OSS Analytics Batch server



**Figure 1 – HP OSS Analytics Foundation Server**

## OSS Analytics REST server architecture

The HP OSS Analytics Foundation provides a HTTP/HTTPS REST API for external applications to access and compute the required information stored in a data warehouse.

The main operations done by the HP OSS Analytics Foundation REST server are:

- Analyze the HTTP request
- Generate the corresponding SQL query, by using the domain specific metadata
- Execute the SQL query on the data warehouse
- Retrieving the set of data
- Sending the HTTP response

The HP OSS Analytics Foundation REST server consists of:

- *REST server* and query generator, contained in a JBoss server  
(Note that JBoss is embedded within the software kit)
- Repository loader tool

The Repository loader tool allows to load metadata, which is then stored in the database, in a dedicated schema.

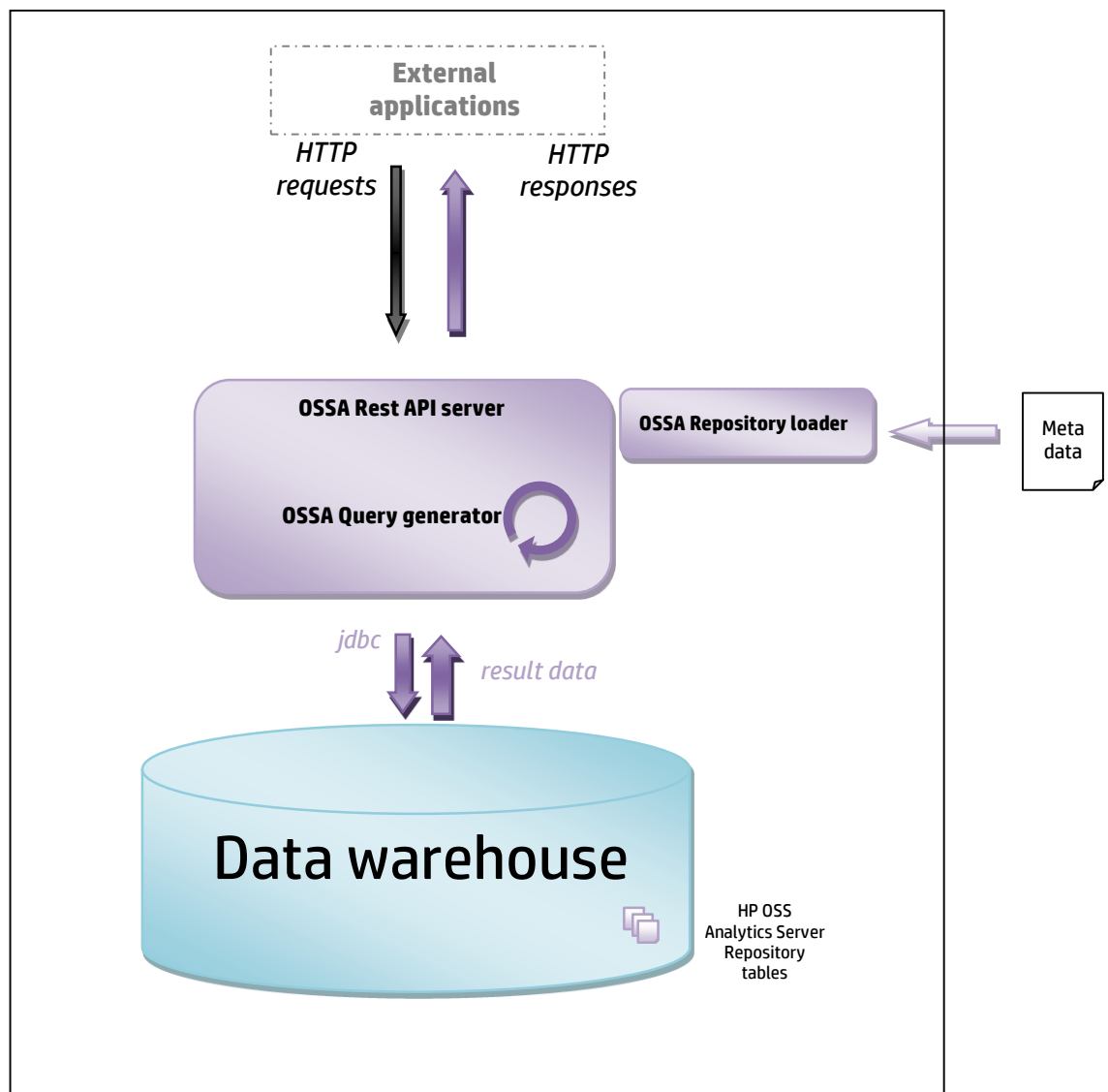


Figure 2 – HP OSS Analytics REST Server architecture

### OSS Analytics Batch server architecture

The HP OSS Analytics Foundation provides a Batch job server relying on the JEE Batch API (Batch Applications for the Java Platform, JSR-352).

The main operations done by the HP OSS Analytics Foundation Batch server are:

- Schedule batch jobs whose schedule descriptions are loaded into the repository
- Execute the batch jobs whose processing descriptions are also loaded into the repository

The definitions and executions of the batch jobs rely on the OSS Analytics Batch Library which provides several kind of data processing:

Load data to Vertica, SQL execution, summarization of data, schedule of reports, mailing.

This set of components relies on the JEE Batch API.

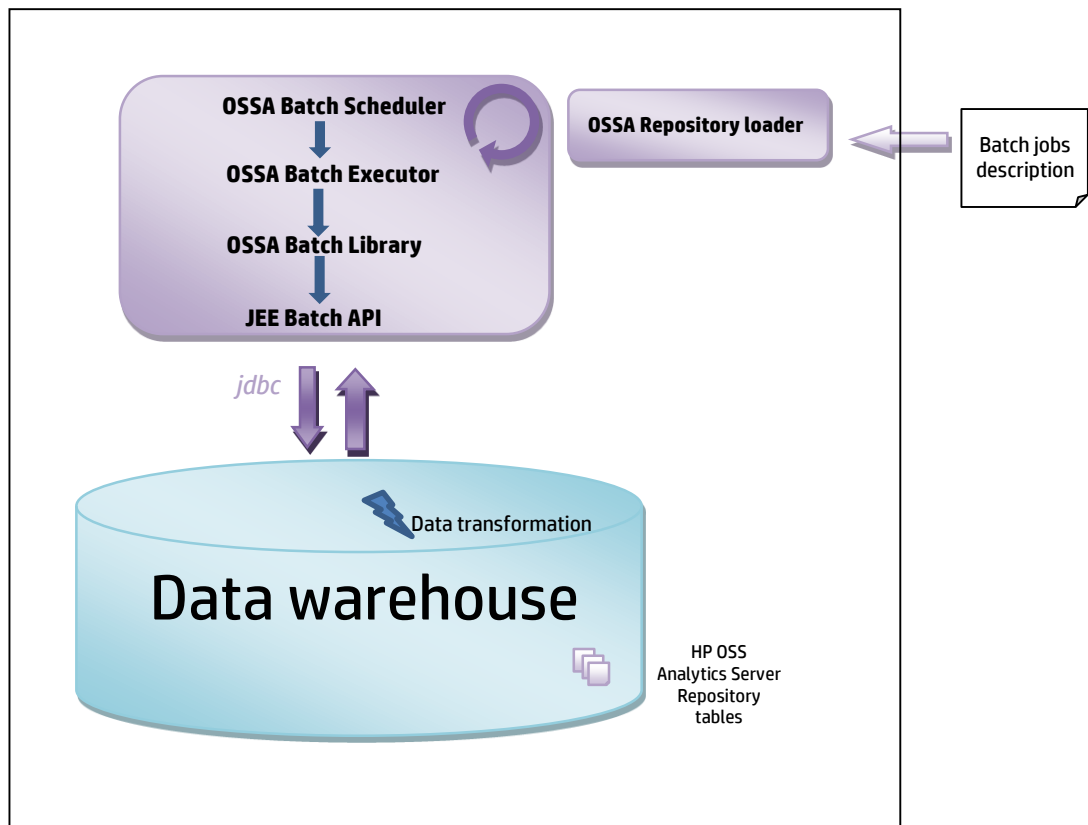


Figure 3 – HP OSS Analytics Batch Server architecture

## Installation prerequisites

### 2.1 Hardware and operating systems prerequisites

The HP OSS Analytics Foundation is supported only on Red Hat Enterprise Linux v6.x, x86-64.

Before installing HP OSS Analytics Foundation, verify that your system meets the following minimum requirements.

**Table 2 – Minimum hardware requirements for HP OSS Analytics Server on Linux**

Hardware	Minimum requirement
CPU	1 CPU 2.5 GHz 4 cores
RAM	8 GB
Hard disk size	50 GB
Network	100 MB Ethernet

#### Note

The listed minimum requirements are for HP OSS Analytics Foundation only and not for any other server of the HP OSS Analytics solution.

Contact the product manager for further details on the sizing requirements.

### 2.2 Software prerequisites

#### 2.2.1 Java

HP OSS Analytics Foundation requires *java1.8.0-openjdk*.

Generally, Red Hat Enterprise Linux Server comes with OpenJDK Java VM, but check if it is installed.

As *root* user, run:

```
# rpm -qa | grep 1.8.0-openjdk
```

If *java1.8.0-openjdk* is not installed, install it.

As *root* user, run:

```
# yum update java-1.8.0-openjdk
```

## 2.2.2 Database

HP OSS Analytics Foundation retrieves domain-specific data from a Vertica data warehouse. Thus, HP OSS Analytics Server requires the HP Vertica client to be installed in order to access the data.

Note that HP OSS Analytics Foundation stores also its internal repository tables in this database.

### **HP Vertica is a prerequisite for HP OSS Analytics Foundation.**

HP OSS Analytics Foundation supports HP Vertica 7.1.x database, version 7.1.0 and later.

If your domain specific data warehouse is not yet installed, see the HP Vertica Installation Guide (<http://www.vertica.com/documentation/hp-vertica-7-1-x-documentation>) for information on how to install your Vertica database on a dedicated server.

Contact your Vertica database administrator on setup and configuration information for your environment.

#### **Note**

---

The number of concurrent users that can be served in a suitable response time by HP OSS Analytics Server depends on the ability of the Vertica cluster to serve the simultaneous requests. It is recommended to build a Vertica platform aligned with the Vertica hardware guide and the system settings described in the Vertica installation and administration guides.

---

## 2.2.3 Vertica client

Version 7.1.0, or a later version of the 7.1 series of Vertica client has to be installed on the HP OSS Analytics Foundation system.

Align the Vertica client with the version of Vertica server that the client connects to.

Check if Vertica client is already installed. As *root user*, run:

```
# rpm -qa | grep vertica
```

If not installed, download **Vertica client 7.1.x** from the HP Vertica website and install it.

## 2.2.4 Timezone settings

### **WARNING !**

---

While configuring your platform please ensure that the **local timezone** is set for the **operating system** and for the **java virtual machine** and for the **Vertica database**.

---

## 2.3 Installation package

The HP OSS Analytics Foundation software component is delivered as a Red Hat Enterprise Linux RPM package:

```
ossa-server-1.1.1-MP.noarch.rpm
```

## 2.4 Code signing

The code signing procedure allows you to assess the integrity of the delivered product before installing it, by verifying the signature of the software packages.

HP recommends using signature verification on its products. Customers have the choice of running this verification, or not, according to their IT policies.

Pick the .sig signature file shipped with the product and enter the following GPG command:

```
# gpg --verify <product.sig> <product>
```

Example:

```
gpg --verify xxxx.rpm.sig xxxx.rpm
## Note: Look for the comments shown below in the command
output
Good signature from "Hewlett-Packard Company (HP Code
signing Service)"
```

### 2.4.1 Verifying an HP product signature using GPG

1. Check if the GNU Privacy Guard (GnuPG or GPG) is installed on the system. If not, install GPG.
2. Configure GPG to accept HP signature:
  - a. Log in as root to your system.
  - b. Get the HP public key from:

<https://h20392.www2.hp.com/portal/swdepot/displayProductInfo.do?productNumber=HPLinuxCodeSigning>

- c. Save it as *hpPublicKey.pub*.

The HP public key file is stored in the root's home directory.

- d. Follow the instruction in the "Verification using GPG" section of the URL from which the HP public key was retrieved.



## Product installation

### 3.1 Creating the HP OSS Analytics Server Linux user

You need to have root credentials for installing the package; but, the installed files are owned by the standard user, and no processes must run under the root account.

For security reasons, no Linux user is created automatically during the installation process. Therefore, the required user must be created manually prior to the installation.

1. To create the 'ossa' HP OSS Analytics Foundation Linux user, run the following commands as *root* user:

```
# useradd -g users -b /home -m -s /bin/bash \  
-c 'HP OSS Analytics user' ossa
```

2. Modify the password of the ossa user:

```
# passwd ossa
```

### 3.2 Installing the RPM kit

Install the HP OSS Analytics Foundation package on your Linux system.

As *root* user, enter:

```
# rpm -ivh ossa-server-1.1.1-MP.noarch.rpm
```

The default installation location of the RPM package is */opt/ossa*.

### 3.3 Creating the ossa database user and schema

The HP OSS Analytics Foundation repository stores some parameters and metadata in the database. So, it is required to create a dedicated *ossa* database user and a dedicated *ossa* schema.

The *ossa* database schema has to be located in the same database as the data warehouse that is accessed by HP OSS Analytics Foundation.

1. If your data warehouse is not yet created, create the database following the instructions of the HP OSS Analytics solutions you have chosen.

See the HP Vertica installation guide for details.

2. Ensure that the HP Vertica *vsql* client software is in the *ossa* Linux user *PATH* environment variable.
3. Create the *ossa* database user within the database created previously.

Note that if your database is not named 'OSSA',  
in */opt/ossa/ddl/create\_user\_ossa.sql* replace

```
DATABASE OSSA with DATABASE <yourDatabaseName>
```

For creating the *ossa* DB user, enter as *ossa* Linux user:

```
$ cd /opt/ossa/ddl
$ vsql -d <DBname> -h <DBIPAddress> \
  -U <DBAdministratorUserName> \
  -w <DBAdministratorPassword> -f create_user_ossa.sql
```

4. Create the *ossa* schema for this *ossa* database user.  
Enter as *ossa* Linux user:

```
$ vsql -d <DBname> -h <DBIPAddress> -U ossa -w ossapwd \
  -v ossa_user=ossa -f create_schema_ossa.sql
```

5. Verify that the schema was created successfully by running the following command as *ossa* Linux user:

```
$ echo "select id from package" | vsql -d <DBname> -h
<DBIPAddress> -U ossa -w ossapwd
```

Output example:

```
id
----
(0 rows)
```

You are now ready to configure the HP OSS Analytics Foundation.

## 3.4 Configuring the installation

All remaining steps must be run with the 'ossa' unix user.

1. As *ossa* user, edit the */opt/ossa/ossa.cfg* installation configuration file.
2. Modify the parameters according to your environment.

The expected values for the parameters are the following:

**Table 3 – ossa.cfg parameters description**

Parameter	Expected value
<i>JBOSS_CONFIG</i>	The name of the JBoss configuration file that will be used. Keep the default value ' <i>standalone.xml</i> ' for standard usage. (could be modified in future versions)
<i>JBOSS_HOST</i>	The address to which JBoss binds. Keep the default values for standard usage. By default, it uses the result of ' <code>hostname -i</code> ' which is the IP address of the host.
<i>JBOSS_HTTP_PORT</i>	The HTTP port of reference for JBoss web server. Keep the default value " <i>8080</i> " for standard usage.
<i>JBOSS_PORT_OFFSET</i>	The value to add to the default port number ( <i>JBOSS_HTTP_PORT</i> ) for defining a different port on which JBoss runs. Ex: value 100 makes JBoss web server running on port 8180. The default value is 0, meaning that JBoss runs on default port 8080.
<i>JBOSS_JAVA_OPTS</i>	The Java options to pass to the JBoss server. Keep the default values for standard usage.
<i>JBOSS_SERVER_OPTS</i>	(optional) The JBoss server options you want to pass.
<i>OSSA_DB_TYPE_01</i>	The data warehouse database type. For HP OSS Analytics Foundation version 1.1, only ' <i>vertica</i> ' is supported.
<i>OSSA_DB_HOST_01</i>	The IP address of the database server hosting the data warehouse.
<i>OSSA_DB_NAME_01</i>	The name of the database hosting the data warehouse
<i>OSSA_DB_PORT_01</i>	The port of the database
<i>OSSA_DB_USER_01</i>	The name of the dedicated database ossa user previously created
<i>OSSA_DB_PASSWORD_01</i>	The password for the dedicated ossa user previously created
<i>OSSA_DS_MAX_POOL_SIZE_01</i>	The maximum number of connections created in the OSSA Server pool for accessing the data warehouse.
<i>OSSA_DS_NAME</i>	The name of the OSSA datasource within JBoss configuration. Keep the default values for standard usage.
<i>OSSA_SSL_ENABLED</i>	Y or N. The default value is N, meaning that SSL is not enabled. You do not have to modify the other <i>OSSA_SSL_XXX</i> parameters if you do not enable SSL. For enabling and configuring SSL correctly, see section 4.3.1

Parameter	Expected value
<i>OSSA_MAIL_SERVER</i>	(optional) The address of the mail server you want to declare in JBoss configuration
<i>OSSA_MAIL_PORT</i>	(optional) The port of the mail server you want to declare in JBoss configuration

The *OSSA\_DB\_xxx\_01* parameters allow to define the connection to the data warehouse through an internal *OSSA\_DB\_URL\_01* variable. The format of the variable is:

```
jdbc:${OSSA_DB_TYPE_01}://${OSSA_DB_HOST_01}:${OSSA_DB_PORT_01}/${OSSA_DB_NAME_01}
```

If you have some specificities to define for the *OSSA\_DB\_URL\_01*, declare its value in the *ossa.cfg* file by adding the line:

```
OSSA_DB_URL_01= <your specific DB connection url>
```

Once you have checked that all the listed parameters have correct values, use the defined work environment by sourcing the *ossa\_env.sh* file:

```
$ source ${OSSA_HOME}/bin/ossa_env.sh
```

## 3.5 Finalizing the installation

Run the *ossa-config* tool in order to automate the configuration of the server based on the parameters you defined previously:

```
$ sh ${OSSA_HOME}/bin/ossa_config.sh
```

The HP OSS Analytics Server is automatically started.

If you want to activate the OSS Analytics Batch services please load also run:

```
$ sh ${OSSA_HOME}/bin/ossa_config_batch.sh
```

The OSS Analytics Foundation installation is now finalized.

## 3.6 Validating the installation

1. Open an internet browser and check the following URL to confirm that the JBoss server is running :

```
http://<OSSA_server>:8080/
```

### Note

---

<OSSA\_server> is the IP address of the host on which you installed HP OSS Analytics Foundation.

8080 is the default port used for the server.

You may have modified this port by setting an offset at configuration time (section 3.4, parameter *JBOSS\_PORT\_OFFSET*).

---

```
http://<OSSA_server>:8080/ossa/packages/
```

This checks if HP OSS Analytics Foundation (Rest API server) is running.

The output gives you the list of metadata packages loaded into the server.

If no metadata packages are loaded, the result will simply be: [ ]

### Note

---

HP OSS Analytics Foundation does not provide any metadata instantiations, so after a fresh installation, the list of metadata is empty.

---

You are now ready to load your domain-specific metadata.

## Product configuration

### 4.1 Configuring a dedicated Vertica resource pool

The HP OSS Analytics Server might not be the only software component accessing the Vertica database. In order to separate the different accesses to the database and to configure them differently, you can define a dedicated Vertica resource pool for OSSA Server.

1. As Vertica database administrator (*dbadmin*), create a resource pool dedicated to the *ossa* database user.

```
$ vsql -d <DBname> -h <DBIPAddress> -U
<DBAdministratorUserName> -w <DBAdministratorPassword>

OSSA=> create resource pool pool_OSSA;
OSSA=> grant usage on resource pool pool_OSSA to OSSA;
OSSA=> alter user OSSA resource pool pool_OSSA;
OSSA=> \q
```

### 4.2 Configuring HP OSS Analytics Foundation as a Red Hat Linux service

1. To configure HP OSS Analytics Foundation as a RedHat Linux service in order to have it automatically started at boot time, and stopped at halt time, as *root* user, enter:

```
# ln -s /opt/ossa/bin/ossa-server.sh /etc/init.d/ossa-
server
# chkconfig --add ossa-server
```

## 4.3 Security

### 4.3.1 Securing the server

By default, secured communication across the components of the server is not enabled, as it requires first the administrator to install the certificate of the platform for the SSL authentication. It is strongly recommended to activate SSL as described in section 4.3.2, to protect private data that might be published by the deployed specific domain metadata.

### 4.3.2 Activating SSL

This process is optional but the HP OSS Analytics Foundation can be configured with server authentication and secured communication, with the use of HTTPS.

## Note

---

This feature deals with the authentication of the HP OSS Analytics Foundation for the web clients.

It does not aim at authenticating web clients. For authenticating web clients, your user interface application has to rely on an independent identity provider, specific to your company that manages the users.

---

1. As *ossa* user, edit the */opt/ossa/ossa.cfg* installation configuration file.

2. Modify the parameters according to your environment.

For test or demonstration purposes, you can configure the HP OSS Analytics Foundation with an example of SSL certificate, which is automatically generated during configuration.

- a. During the configuration, modify the *OSSA\_SSL\_ENABLED* parameter in *ossa.cfg*, to *Y*.
- b. Keep the default values of the other *OSSA\_SSL\_XXX* parameters.

In the application server, a self-signed certificate is automatically generated with a fake organizational identity and location.

The generated certificate is an example that can be used only for demonstration purposes.

On production systems, you have to include your company SSL certificate (Java keystore file) and define all the *OSSA\_SSL\_XXX* parameters.

Check the expected values thoroughly for the parameters:

**Table 4 – HP OSSA SSL configuration parameters description**

Parameter	Expected value
<i>OSSA_SSL_ENABLED</i>	Y or N. Set it to Y for enabling SSL.
<i>OSSA_SSL_KEYTOOL_PATH</i>	The path to the <i>keytool</i> utility. <i>keytool</i> is the cryptographic key and certificate management utility which is installed with <i>java-1.8.0-openjdk</i> .
<i>OSSA_SSL_JAVA_KEYSTORE_FILE</i>	The location of your company SSL certificate; a Java <i>keystore</i> file containing keys and the certificate.
<i>OSSA_SSL_KEYSTORE_ENTRY_ALIAS</i>	The logical name of the entry within the <i>keystore</i> file, where the certificate and the private key are stored.
<i>OSSA_SSL_KEYSTORE_PASSWD</i>	The password of the Java <i>keystore</i> file.

3. Use the work environment with the configured parameters, by sourcing the *ossa\_env.sh* file:

```
$ source ${OSSA_HOME}/bin/ossa_env.sh
```

4. Finalize the configuration of the HP OSS Analytics Foundation (see section 3.5)

With those settings, for accessing the HP OSS Analytics Foundation, you must use the HTTPS protocol, and use the corresponding port number. The default port number is 8443, if you did not modify the value of the *JBOSS\_PORT\_OFFSET* parameter.

### 4.3.3 Cryptographic algorithms

Using SSL, the certificates are managed through the regular configuration of Java SE 7 JCE keystores.

If you use the automatically generated, self-signed example certificate, the encryption is RSA, and the key length is 2048.

By using your company SSL certificate, the encryption algorithm and key length can differ.



### 4.3.4 Enabling auditing

Security event audit logging is disabled by default, because it may introduce a performance impact on servers.

This feature has to be configured to be available on-demand only.

1. To enable it, in the `${JBOSS_HOME}/standalone/configuration/standalone.xml` file, set the following values:

```
<audit-log>
...
  <logger enabled="true" log-boot="true" log-
read-only="false">
...
</audit-log>
```

2. Restart the server.

The audit log is available at:

`${JBOSS_HOME}/standalone/data/audit-log.log`

#### Warning

---

These logs must be accessible only to platform administrator, because they may contain sensitive privacy information, for example the IP address.

---

### 4.3.5 Authentication / Single Sign-On

The HP OSSA Server does not provide a client authentication mechanism. If your external user interface application enables integration with identity providers through the SAML V2.0 protocol, and you want the application server from HP OSSA Server to host a *PicketLink* (open source) identity provider, follow the steps listed in Annex 1 at the end of this document.

### 4.3.6 Role based access control

The HP OSS Analytics Foundation features a role based access control (RBAC).

The definitions of the different roles are defined within each specific domain metadata. The definitions determine which roles can access which data. The definition and use of roles are optional.

The HP OSS Analytics Foundation analyzes the user roles sent in the request URLs and compares them to the ones specified within the metadata for the requested data. The HP OSS Analytics Foundation then determines whether the data can be accessed or not.

### 4.3.7 Encryption

The Vertica JDC driver does not encrypt the database connection password. If the platform deployment exposes the dedicated private network, the IP connection between the HP OSSA server, and the underlying Vertica database, it is strongly recommended to configure IPsec for that transport, to secure the connection.

### 4.3.8 DEBUG traces logging

If troubleshooting requires to enable the DEBUG traces logging, some private, and confidential data recorded in the database can appear in the log files. Access to these files must be limited to the authorized users. Debug trace logging is to be managed with a system login account.

## Product administration

### 5.1 Starting the HP OSS Analytics Server

In the terminal where you sourced *ossa\_env.sh*, run the following command as *ossa* user:

```
$ jbossstart
```

### 5.2 Stopping the HP OSS Analytics Server

In the terminal where you sourced *ossa\_env.sh*, run the following command as *ossa* user:

```
$ jbossstop
```

### 5.3 Loading metadata

1. Check the HP OSS Analytics solution you have chosen in order to retrieve your domain specific metadata. This metadata contains information about:
  - Description of the data to analyze
  - Database location and tables and columns that contains the data
  - Type of aggregations that can be performed on the data

This metadata file must be loaded to OSS Analytics server with the use of the repository loader.

2. Make sure that:
  - OSS Analytics server is running
  - Metadata specific database schema is loaded
3. In the terminal where you sourced *ossa\_env.sh*, as *ossa* user, load your metadata:

```
$ ${OSSA_HOME}/bin/ossa-repo.sh loadMetadata \  
  <full_path_to_metadata_file>
```

Output:

```
loadMetadata <metadata file>  
OK
```

4. Then, you can start using the HP OSS Analytics Foundation for your specific domain, through your user interface application. Depending on the HP OSS Analytics solution you are deploying you could use the HP Unified OSS Console v2.x with its HP OSS Analytics add-on.

To load the metadata and the associated HP OSS Console v2 views and workspaces JSON definition files, in the terminal where you sourced *ossa\_env.sh*, as *ossa* user, execute:

```
$ ${OSSA_HOME}/bin/ossa-repo.sh loadMetadataViewsWks \  
  <full_path_of_metadataFile.xml> \  
  <full_path_of_UOC_ViewsFile.json> \  
  <full_path_of_UOC_WorkspacesFile.json>
```

## 5.4 Loading a batch job

1. Check the HP OSS Analytics solution you have chosen in order to retrieve your domain specific transformations.  
The *<batchJobName.xml>* file contains the applicative processing of the batch job.  
The *<BATCH\_batchJobName.json>* file contains the execution description of the batch job: job parameters, administrative state (lock or unlocked), batch schedule.
2. Make sure that:
  - OSS Analytics server is running
  - OSS Analytics server has been configured with the Batch Job processing functionality
3. In the terminal where you sourced *ossa\_env.sh*, as *ossa* user, load the description of the processing of the job.

```
$ ${OSSA_HOME}/bin/ossa-repo.sh loadParam <packageName> \  
  <jobProcessingName> <BatchJob.xml>
```

Then, load the execution description of the job.

```
$ ${OSSA_HOME}/bin/ossa-repo.sh loadParam <packageName> \  
  BATCH_<jobDescriptionName> <BATCH_batchJob.json>
```

### Notes - Warnings

- 
- It is mandatory to prefix the *<jobDescriptionName>* with '**BATCH\_**'
  - Within the *<BATCH\_batchJob.json>* file, the *jobXmlPath* value must be identical to the *<jobProcessingName>* that you have put in the first command.
  - *<BatchJob.xml>* and *<BATCH\_batchJob.json>* must be full path to file.
- 

If you have loaded an unlocked batch job, **the batch job is automatically scheduled and will start at next schedule time.**

## 5.5 Loading batch descriptions contained in a directory

This command allows you to load several batch jobs which applicative processing descriptions and execution descriptions are contained in a directory.

In the terminal where you sourced *ossa\_env.sh*, as *ossa* user, execute:

```
$ ${OSSA_HOME}/bin/ossa-repo.sh loadDirectory <packageName> \  
    <full_path_of_directory>
```

This loads into the OSSA repository for the *<package\_name>*, the content of the directory (recursively scanning directories and sub directories):

- "param\_name" being the relative path of each file (relative to the *<full\_path\_of\_directory>* passed as parameter)
- "param\_value" being the content of each file

In that way, a hierarchy of .json and .xml files contained in a directory can be uploaded with a one shot command.

(the same recommendations, as previous section in term of naming of files and content of files must be followed)

## 5.6 Running a batch job

In case you want to launch a one-time execution of your batch job (for testing purpose for example), you have two choices:

- o either you can run the job synchronously, meaning that the command gives you back the control only once the job is completed:

```
$ ${OSSA_HOME}/bin/ossa-run-batch-sync.sh <jobName> \  
    <jobParameters.json>
```

- o or you can run the job asynchronously, meaning that the control is given back to you directly after you enter the command , without waiting for the completion of the batch job:

```
$ ${OSSA_HOME}/bin/ossa-run-batch.sh <jobName> \  
    <jobParameters.json>
```

## 5.7 OSS Analytics Foundation administration console



The HP OSS Analytics Foundation server provides a web console for administration purpose. The main usage of this console is to monitor the execution of the batch jobs.

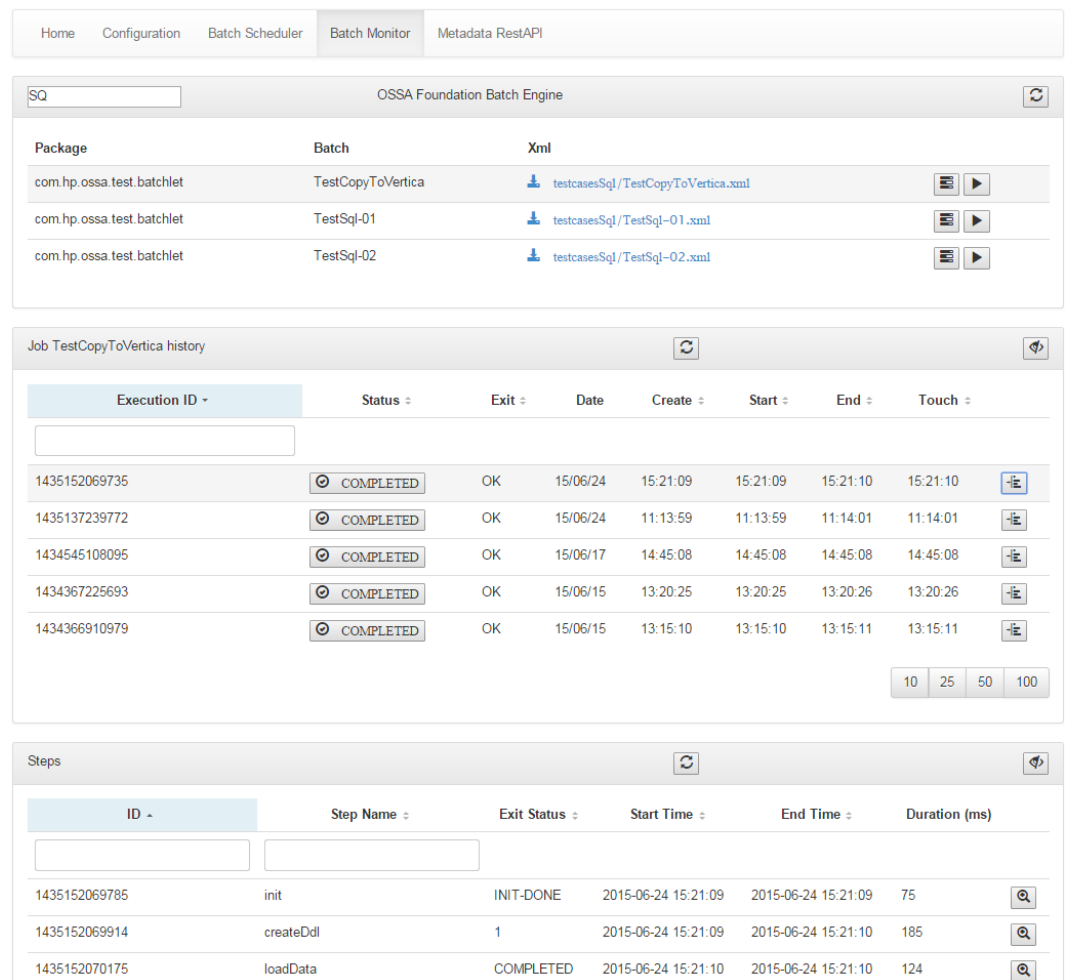
This administration console is available at the following url:

```
http://<OSSA_server>:8080/
```

## 5.7.1 Batch monitoring

When clicking on the **Batch Monitor** tab, the list of batch jobs is displayed.

- The **Xml** column allows you to click on each batch job xml in order to display its processing description.
- When clicking on the button  at the end of the line, you have access to the history of the batch jobs executions.
  - o If you want some more details about a job execution, you can click on the button  at the end of the line: this then lists the steps executions of this batch job.




The screenshot displays the HP OSSA Foundation administration console's Batch Monitor interface. It features a navigation bar with tabs for Home, Configuration, Batch Scheduler, Batch Monitor, and Metadata RestAPI. Below the navigation bar is a search box containing 'SQ' and the title 'OSSA Foundation Batch Engine'. The main content area is divided into three sections:







- Batch Jobs Table:** A table with columns for Package, Batch, and Xml. It lists three batch jobs: 'TestCopyToVertica', 'TestSql-01', and 'TestSql-02'. Each row includes a list icon and a play button icon.
- Job TestCopyToVertica history:** A table showing the execution history for the 'TestCopyToVertica' job. The columns are Execution ID, Status, Exit, Date, Create, Start, End, and Touch. All listed executions are in a 'COMPLETED' status with an 'OK' exit code.
- Steps:** A table showing the individual steps of a job execution. The columns are ID, Step Name, Exit Status, Start Time, End Time, and Duration (ms). The steps listed are 'init', 'createDdl', and 'loadData', all of which are in a 'COMPLETED' status.

**Figure 4 – HP OSSA Foundation administration console – Batch Monitoring**

## 5.7.2 Batch scheduling administration

The Batch Scheduler tab allows you to display and administrate batch scheduling. You can see the administrative state of a batch job: Locked or Unlocked. You can modify it.

You also have access to the scheduling information by pushing the  button.

OSSA Foundation Scheduler											
Package	Batch	Status		Scheduling							
				Day Of Week	Day Of Month	Month	Year	Hour	Minute		
com.hp.ossa.test.batchlet	testcasesSql/BATCH_TestcaseTestCopyToVertica.json	Locked		*	*	*	*	*	*	*/1	
com.hp.ossa.test.batchlet	testcasesSql/BATCH_Testcase_testSql-01.json	Locked		*	*	*	*	*	*	*/1	
com.hp.ossa.test.batchlet	testcasesSql/BATCH_Testcase_testSql-02.json	Locked		*	*	*	*	*	*	*/1	

**Figure 5 – HP OSSA Foundation administration console – Batch Administration**

Day of week: \*  
0 to 7 (both 0 and 7 refer to Sunday). For example: dayOfWeek="3".  
Sun, Mon, Tue, Wed, Thu, Fri, Sat. For example: dayOfWeek="Mon".

Day of Month: \*  
1 to 31. For example: dayOfMonth="15".  
-7 to -1 (a negative number means the nth day or days before the end of the month). For example: dayOfMonth="-3".  
Last. For example: dayOfMonth="Last".  
[1st, 2nd, 3rd, 4th, 5th, Last] [Sun, Mon, Tue, Wed, Thu, Fri, Sat]. For example: dayOfMonth="2nd Fri".

Month: \*  
1 to 12. For example: month="7".  
Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec. For example: month="July".

Year: \*  
A four-digit calendar year. For example: year="2011".

Hour: \*  
0 to 23. For example: hour="13".

Minute: \*/1  
0 to 59. For example: minute="15".

com.hp.ossa.test.batchlet  
testcasesSql/BATCH\_TestcaseTestCopyToVertica.json :

```
{
  "jobXmlPath": "testcasesSql/TestCopyToVertica.xml",
  "jobParameters": {},
  "adminState": "Locked",
  "batchSchedule": {
    "minute": "*/1",
    "dayOfWeek": "**",
    "dayOfMonth": "**",
    "month": "**",
    "year": "**",
    "hour": "**"
  }
}
```

See [JZEE Timer Service documentation](#) for more details.

Cancel Save

**Figure 6 – HP OSSA Foundation administration console – Batch Scheduling**

## Product troubleshooting

### 6.1 Logging

The log files related to HP OSS Analytics Foundation are stored at the following location:

```
/opt/ossa/wildfly-8.2.1.Final/standalone/log/
```

1. *server.log* for the JBoss logs
2. *ossa\_server.log* for the HP OSS Analytics Foundation server log files

The current log level used for each component can be displayed.

1. As *ossa* user, enter the following command in the terminal where you sourced *ossa\_env.sh*, as *ossa* user:

```
$ ossa_server_get_log
```

Output example:



```

[standalone@xx.xx.xx.xx:9990 /]
/subsystem=logging/logger=com.hp.ossa.common:read-
attribute(name=level)
{
    "outcome" => "success",
    "result" => "INFO"
}
[standalone@xx.xx.xx.xx:9990 /]
/subsystem=logging/logger=com.hp.ossa.repo:read-
attribute(name=level)
{
    "outcome" => "success",
    "result" => "INFO"
}
[standalone@xx.xx.xx.xx:9990 /]
/subsystem=logging/logger=com.hp.ossa.querygen:read-
attribute(name=level)
{
    "outcome" => "success",
    "result" => "INFO"
}
[standalone@xx.xx.xx.xx:9990 /]
/subsystem=logging/logger=com.hp.ossa.restapi:read-
attribute(name=level)
{
    "outcome" => "success",
    "result" => "INFO"
}

```

The *INFO* level is the default setting.

Note that the *DEBUG* level allows you to see, for example, all the requests sent to the server from the different client IP addresses, and the result of each request.

2. For troubleshooting purposes, you can decide to set a different log level for all the loggers by executing the command:

```
$ ossa_server_set_log <level>
```

Where <level> can take the following values, in descending order of detail:

TRACE, DEBUG, INFO, WARN, ERROR, FATAL.

#### Note

---

Check the disk space consumption when setting levels below INFO level.

---

3. (Optional) To set a different level of log on specific loggers, use this command:

```
$ jbosssetlog <logger> <level>
```

## 6.2 Common errors

### 6.2.1 Deployment error about OssaDS

When starting HP OSS Analytics Foundation, if you encounter the following error in server log files, it means that your HP OSS Analytics Foundation data source cannot be reached.

```
ERROR [org.jboss.as.controller.management-operation]
(DeploymentScanner-threads - 1) JBAS014613: Operation
("deploy") failed - address: ([("deployment" => "ossa-
restapi-1.1.0.ear")]) - failure description:
{"JBAS014771: Services with missing/unavailable
dependencies" => [
  "jboss.naming.context.java.comp.ossa-
restapi.querygen.QueryGen.env.\"com.hp.ossa.querygen.Quer
yGen\".ds is missing
[jboss.naming.context.java.jboss.datasources.OssaDS]",
  "jboss.naming.context.java.comp.ossa-
restapi.repo.RepoBean.env.\"com.hp.ossa.repo.RepoBean\".d
s is missing
[jboss.naming.context.java.jboss.datasources.OssaDS]"
]}
```

1. To troubleshoot the problem, as *ossa* user, source your environment:

```
$ source ${OSSA_HOME}/bin/ossa_env.sh
```

2. Check if you can connect to your Vertica database using the Vertica client:

```
$ vsql -d ${OSSA_DB_NAME_01} -h ${OSSA_DB_HOST_01}
-p ${OSSA_DB_PORT_01} -U ${OSSA_DB_USER_01}
-w ${OSSA_DB_PASSWORD_01}
```

If you can connect to the server, it means that there was an environment problem at the time the HP OSS Analytics Foundation was started.

3. Before using `jbossstop` or `jbossstart`, make sure that as *ossa* Linux user, you source the following:

```
${OSSA_HOME}/bin/ossa_env.sh.
```

If you cannot connect to the server, it means that:

- Your Vertica database is not started. Contact your Vertica database administrator, and restart the HP OSS Analytics Foundation.
- Configuration parameters you have set in the *ossa.cfg* are incorrect. Check the correctness of the *OSSA\_DB\_XXX* parameters carefully. Once they are corrected, follow those steps:

1. Stop HP OSS Analytics Foundation.
2. Remove the `/opt/ossa/wildfly-8.2.1.Final` directory.
3. Re-run the configuration, as described in section 3.4,

# Uninstalling the HP OSS Analytics Server

1. In the terminal where you sourced *ossa\_env.sh*, as *ossa* user, stop the HP OSS Analytics Foundation:

```
$ jbossstop
```

2. As root user, uninstall the HP OSS Analytics Foundation Linux package:

```
# rpm -ev ossa-server-1.1.1-MP.noarch.rpm
```

3. (Optional) If you have configured HP OSS Analytics Server as a Red Hat Linux service, as root user, enter:

```
# chkconfig --del ossa-server  
# rm /etc/init.d/ossa-server
```

### Note

---

The uninstallation does not remove the *ossa.cfg* file, which contains your HP OSS Analytics Foundation configuration.

If you plan to install a new version of the HP OSS Analytics Foundation, keep this file at its current location. During the next installation, the file is not overwritten and it is used as it is now. In that way, the same configuration is used for your next installation.

---

# Annex 1

## Setting up the PicketLink identity provider

### Prerequisite

The Apache Ant tool is required to set up this identity provider.

### Configuring JBoss Wildfly

#### Updating the PicketLink libraries

1. Update the PicketLink libraries.

The Identity Provider requires PicketLink 2.7.0, or later, which is not the version embedded within the HP OSSA Server Jboss. Use the installer provided by [picketlink.org](http://picketlink.org).

2. Stop the Jboss server if it is running, and make sure you have read and write access to your JBoss server installation.
3. Download the installer from:  
<http://downloads.jboss.org/picketlink/2/latest/picketlink-installer-2.7.0.CR1.zip>
4. Extract the files from the archive and open the *picketlink-installer-2.7.0.CR1* extracted folder. The content of the folder is:
  - *Config folder*
  - *Tmp folder*
  - *Build.xml file*
  - *Installer.properties file*
5. Open a command line interface, and navigate to the *picketlink-installer-2.7.0.CR1* folder.
6. Run the *ant* command to launch the installation script.
7. Answer the JBoss question “Which JBoss Application Server are you using?”-
8. Enter the path to your JBoss server installation (*\_\${JBOSS\_HOME}*)

For example (*opt/ossa/wildfly-8.2.1.Final*).

Your JBoss server is updated with PicketLink 2.7.0.

#### Creating the security domain

It is assumed that you run the server in standalone mode and use the *standalone.xml* supplied with the distribution.

The JBoss server must be stopped, and the *\_\${JBOSS\_HOME}/standalone/configuration/standalone.xml* server configuration file must be backed up. The backup file can be used to restore the server to its original configuration.

1. Start the JBoss server
2. Navigate to the *root* directory of the identity provider files with a command line interface.
3. Run the following command, replacing *JBOSS\_CONTROLLER* with a management endpoint of your server: By default, it is *\$JBOSS\_HOST:9990*.

```
${JBOSS_HOME}/bin/jboss-cli.sh --connect  
controller=<JBOSS_CONTROLLER> --file=configure-security-  
domain.cli
```

Output example:

```
The batch executed successfully  
{  
"outcome" => "success",  
}
```

4. Restart the JBoss server

## Deploying the identity provider

1. Run the JBoss server, if it is not running yet.
2. Enter the following command, replacing *JBOSS\_CONTROLLER* with a management endpoint of your server. By default, it is *\$JBOSS\_HOST:9990*.

```
${JBOSS_HOME}/bin/jboss-cli.sh -connect \  
--controller=${JBOSS_HOST}:`expr 9990 + ${JBOSS_PORT_OFFSET}` \  
--command="deploy /opt/ossa/idp/hp-picketlink-federation-saml-idp-  
basic-1.0-SNAPSHOT.war --force"
```

The identity provider is now deployed to the JBoss server and is accessible from the address *http://<OSSA\_server>:8080/idp*

## Security

These steps describe how to enable optional security features, such as signature and encryption of SAML assertions. These optional features are based on customizations of the *picketlink.xml* identity provider configuration file.

For more information, see:

<https://docs.jboss.org/author/display/PLINK/Identity+Provider+Configuration>

## Enabling signature support

1. From the root directory of the identity provider files, open the *picketlink.xml* file in the */conf/wildfly/WEB-INF* folder
2. In the *PicketLinkIDP* tag, add the following attributes
  - a. *SupportsSignatures="true"*
  - b. *CanonicalizationMethod="http://www.w3.org/2001/10/xml-exc14n#"*.

This attribute is required if the default canonicalization method is not supported by your external UI application. For the default canonicalization method, see <http://www.w3.org/2001/10/xml-exc14n#WithComments>

3. Under the *Handlers* element, add:
  - a. A child handler:

```
<Handler
class="org.picketlink.identity.federation.web.ha
ndlers.saml2.SAML2SignatureGenerationHandler"
/>
```

b. A child handler:

```
<Handler
class="org.picketlink.identity.federation.web.ha
ndlers.saml2.SAML2SignatureValidationHandler" />
```

These handlers enable the processing of signed assertions.

4. Under the *PicketLinkIDP* tag, configure a *KeyProvider* child element.

```
<KeyProvider
  ClassName="org.picketlink.identity.federation.core.impl.KeyStoreKeyManager">
  <Auth Key="KeyStoreURL" Value="/keystore.jks" />
  <Auth Key="KeyStorePass" Value="password" />
  <Auth Key="SigningKeyPass" Value="password" />
  <Auth Key="SigningKeyAlias" Value="idpcert" />
  <ValidatingAlias Key="localhost" Value="localhost" />
  <ValidatingAlias Key="127.0.0.1" Value="localhost" />
</KeyProvider>
```

Figure: *KeyProvider* element sample

The *KeyProvider* element specifies configurations about the Java KeyStore that is used to sign SAML assertions:

- **KeyStoreURL**: The location of a Java KeyStore
- **KeyStorePass**: The password of the KeyStore
- **SigningKeyAlias** : The alias of the certificate to be used to sign SAML assertions
- **SigningKeyPass** : The password of the certificate referenced by the **SigningKeyAlias**
- **ValidatingAlias**: This element allows to verify the signatures of the SAML assertions. The **Key** attribute defines the alias of the certificate to be used.

This name must match one of the trusted domains of the identity provider. The *value* attribute defines the password of the trusted domain.

## Enabling encryption support

1. From the root directory of the Identity Provider files, open the *picketlink.xml* file in the */conf/wildfly/WEB-INF* folder s.
2. In the *PicketLinkIDP* tag, add the attribute *Encrypt="true"*
3. Under the **Handlers** element, add:

a. child handler

```
<Handler
class="org.picketlink.identity.federation.web.ha
ndlers.saml2.SAML2EncryptionHandler" />
```

b. child handler

```
<Handler  
class="org.picketlink.identity.federation.web.ha  
ndlers.saml2.SAML2SignatureValidationHandler" />
```

These handlers enable the processing of encrypted assertions.

**Note**

---

Do not use the *SAML2EncryptionHandler* with the *SAML2SignatureGenerationHandler* at the same time, otherwise SAML messages are signed several times. In such a case, use only the *SAML2EncryptionHandler*.

---

4. Note: Configure a *KeyProvider* element, as described in the previous section.