Oracle Financial Services Compliance Studio

Installation Guide

Release 8.1.2.5.0

July 2024

F48800-01

ORACLE
Financial Services

ORACLE

OFS Compliance Studio Installation Guide

and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Document Control

Table 1 lists the document control of this guide.

Table 1: Document Control

Version Number	Revision Date	Change Log
8.1.2.5.11	July 2024	Added patch number (v8.1.2.5.11) in the Download the Installer Kit section.
8.1.2.5.10	May 2024	Added patch numbers in the Download the Installer Kit section.
8.1.2.5.8	May 2024	Added patch number (v8.1.2.5.8) in the Download the Installer Kit section. Added the Setup Compliance Studio Instance for Cloning the Filesystem section.
8.1.2.5.7	April 2024	Added patch number (v8.1.2.5.7) in the Download the Installer Kit section.
8.1.2.5.6	March 2024	Added patch numbers in the Download the Installer Kit section. Updated step 1 in the Configure the resources.xml for Graph Schema section.
8.1.2.5.1	October 2023	Added the new "GRPADMIN" and "GRPUSR" User groups in the Appendix E – Create Users, Groups, and Mappings section.
8.1.2.5.1	September 2023	Removed the "Apache Log4j Security Alert CVE-2021-44228 Patch Details" section. Added bug ID for 8.1.2.5.1 patch release in the Download the Installer Kit section.
8.1.2.5.0	August 2023	Added the logstash link in the Table 7. Updated post-installation grants for BD and ECM graphs in the Graph Schema section. Added Synonyms and Stopword files in the Add Synonyms and Stopword files in OpenSearch section.
8.1.2.5.0	July 2023	Added the latest bug ID for v8.1.2.5.0 release in the Download the Installer Kit section. Added new grants (Post-installation grants for BD/ECM Graph Schema) in the Create Graph Schema and Grant Permission section. Added post upgrade steps for v8.1.2.5.0 in the Post Upgrade Steps in case ER Batches are Executed before Upgrade section. Updated script details in the Importing OOB Graph Definition and related Metadata section. Added a new step related to ER Bulk similarity Job gets failure in the Frequently Asked Questions in Compliance Studio section.

Table 1: Document Control

Version Number	Revision Date	Change Log
8.1.2.5.0	July 2023	Added new parameter in the config.sh file, “CLUSTER_TYPE” as "OS". Updated Database Server as “Enterprise Edition” in the Table 7. Added graph schema information in the Create the Tablespace and Graph Schema. The support for Legacy ETL is deprecated in the current release and the related note is added in the required sections. Removed the “Additional Upgrade Steps” section. Removed install open SSL compatibility in the Pre-installation section.

Table of Contents

1 Preface

This section provides the Oracle Financial Services (OFS) Compliance Studio Installation Guide information.

Topics:

1.1 Audience

OFS Compliance Studio Installation Guide is intended for System Engineers who are responsible for installing and maintaining the application.

This document assumes that you have experience in installing Enterprise components and basic knowledge about the following:

- UNIX commands
- Database concepts
- Big Data concepts

1.2 Related Documents

You can strive to keep this and all other related documents updated regularly; visit the OHC Documentation Library to download the latest version available there. The list of related documents is provided here.

- Oracle Financial Services Compliance Studio Administration and Configuration Guide
- Oracle Financial Services Compliance Studio User Guide
- Oracle Financial Services Compliance Studio Matching Guide
- Oracle Financial Services Compliance Studio Data Model Guide
- Oracle Financial Services Compliance Studio Release Notes
- Oracle Financial Services Compliance Studio Use Case Guide

1.3 Conventions

Table 2 lists text conventions are used in this document.

Table 2: Document Conventions

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.

Table 2: Document Conventions

Convention	Meaning
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, file names, text on the screen, or text you enter.
Hyperlink	Hyperlink type indicates the links to external websites and internal document links to sections.

1.4 Abbreviations

Table 3 lists the abbreviations used in this document.

Table 3: Abbreviations

Abbreviation	Meaning
OFS	Oracle Financial Services
Compliance Studio	Oracle Financial Services Compliance Studio
OFSAA	Oracle Financial Services Analytical Application
BD	Behavior Detection
FCDM	Financial Crime Data Model
ICIJ	International Consortium of Investigative Journalists
IDCS	Oracle Identity Cloud Service
ECM	Enterprise Case Management
SSO	Single Sign-On
SSH	Secure Shell

2 Introduction

OFS Compliance Studio is an advanced analytics application that supercharges anti-financial crime programs for better customer due diligence, transaction monitoring, and investigations by leveraging the latest innovations in artificial intelligence, open-source technologies, and data management. It combines Oracle's Parallel Graph Analytics (PGX), Machine Learning for AML, Entity Resolution, and notebook-based code development. It enables Contextual Investigations in one platform with complete and robust model management and governance functionality.

This chapter provides the information required to understand the installation of the OFS Compliance Studio application.

This release (v8.1.2.5.0) of Compliance Studio can be used for the following:

- To install a new instance of Compliance Studio as follows:
- Compliance Studio with OFSAA (Oracle Financial Services Analytical Application). Here, OFSAA is with Behavior Detection (BD) or Enterprise Case Management (ECM).
- Compliance Studio without OFSAA

To upgrade an existing instance of Compliance Studio as follows:

- Upgrade Compliance Studio from v8.1.2.0.1 onwards to Compliance Studio v8.1.2.5.0.

  Or

- Upgrade Compliance Studio from v8.1.2.4.0 onwards to Compliance Studio v8.1.2.5.0.

Topics:

2.1 Installation Checklist when Studio is installed with OFSAA

To complete the installation process, you must perform the steps listed in the Table 4 checklist. Use this checklist to verify whether these steps are completed or not. Click the reference link to go to the topic.

Table 4: Installation Checklist

Sl. No.	Activity	Mandatory	Description
Pre-installation Steps			
1	Install all the prerequisite Hardware and Software Requirements.	Yes	-
2	Setup the environmental settings (System Configuration).	Yes	-

Table 4: Installation Checklist

3	Download the Big Data Files	No	It is required for graph analytics and leverages fragmented data or as a datasource for models. **NOTE:** **This prerequisite is deprecated for Legacy ETL feature.**
4	Configure the OpenSearch Component	No	It is required for graph analytics and leverage fragmented data or for matching service and Entity Resolution
5	Configure the Interpreter Settings	Yes	-
6	Create the Studio Schema	Yes	-
7	Assign Grants for the Studio Schema	Yes	-
8	Setup Password Stores with Oracle Wallet	Yes	-
9	Create the Credential Keystore	No	It is required for graph analytics and leverages fragmented data or as a datasource for models
10	Download the Installer Kit	Yes	-
Installation Steps			
1	Extract the Installer Kit	Yes	-
2	Place Files in the Installation Directories	Yes	-
3	Generate the Public and Private Keys	Yes	-
4	Generate API token for CS API User	Yes	-
5	Place the Key Store File for Secure Batch Service	Yes	-
6	Add the Studio Service (SSL) to PGX Configuration	Yes	-
7	Configure the Extract Transfer and Load (ETL) Process	No	It is required for graph analytics and leveraging fragmented data. **NOTE:** **This step is not applicable as Legacy ETL feature is deprecated.**
8	Configure the config.sh File	Yes	-
9	Run the Compliance Studio Installer	Yes	-
10	Configure the PGX Service	Yes	-

Table 4: Installation Checklist

Post-Installation Steps			
1	Verify the Installation	Yes	-
2	Start the PGX Service	Yes	-
3	Access the Compliance Studio Application	Yes	-
4	Perform the OFSAA Configuration for Batch Execution	No	It is required if leverage OFSAA's scheduling and executing capability.
5	Configure and Run Published Notebooks	No	It is required if leveraging OFSAA's batch execution.

2.2 Installation Checklist when Studio is installed without OFSAA

To complete the installation process, you must perform the steps listed in the Table 5 checklist. Use this checklist to verify whether these steps are completed or not. Click the reference link to go to the topic.

Table 5: Installation Checklist

Sl. No.	Activity	Mandatory	Details
Pre-installation Steps			
1	Install all the prerequisite Hardware and Software Requirements.	Yes	-
2	Setup the environmental settings (System Configuration).	Yes	-
3	Configure the Interpreter Settings	Yes	-
4	Create the Studio Schema	Yes	-
5	See the Configure the resources.xml for Multiple ER Schemas section for more details.	Yes	-
6	Setup Password Stores with Oracle Wallet	Yes	-
7	Create the Credential Keystore	Yes	-
8	Download the Installer Kit	Yes	-
Installation Steps			
1	Extract the Installer Kit	Yes	-
2	Place Files in the Installation Directories	Yes	-
3	Generate the Public and Private Keys	Yes	-
4	Generate API token for CS API User	Yes	-
5	Place the Key Store File for Secure Batch Service	Yes	-

Table 5: Installation Checklist

6	Configure the config.sh File	Yes	-
7	Run the Compliance Studio Installer	Yes	-
Post-Installation Steps			
1	Verify the Installation	Yes	-
2	Access the Compliance Studio Application	Yes	-

3 Pre-installation

This chapter provides information about the tasks that must be performed before installing Compliance Studio. To install Compliance Studio with OFSAA, ensure the Behavior Detection (BD) or the Enterprise Case Management (ECM) application pack is installed.

The following patches are required only when integrating with old versions for ECM:

- On top of ECM 8.0.8.0.0, apply the following ECM patch for ML-ECM integrations.

  8.0.8.0.28 (BUG: **31497997**)
- On top of ECM 8.0.8.1.0, apply the following ECM patch for ML-ECM integrations.

  8.0.8.1.4 (BUG: **33395125**)

> **NOTE** From ECM 8.1.1.0.0 and later versions, the above patches are not required for ML-ECM integrations.

- On top of ECM 8.1.2.0.0, apply the following ECM patch for ECM-IH integration.

  8.1.2.0.8 (BUG: **34337520**)
- On top of ECM 8.1.2.4.0, apply the following ECM patch for ECM-IH integration.

  8.1.2.4.5 (BUG: **35456951**)

Verify the **AIF_USER_TS** and **AIF_USER_TEMP_TS** are available in the BD production database server. If not, then you need to create a tablespace. For information about how to create a tablespace, see Create the Tablespace section.

After creating a tablespace, you need to provide a quota on the tablespace AIF_USER_TS and AIF_USER_TEMP_TS.

For example:

```
ALTER USER <BD ATOMIC SCHEMA USER> QUOTA <size in megabyte> ON AIF_USER_TS;

ALTER USER <BD ATOMIC SCHEMA USER> QUOTA <size in megabyte> ON
AIF_USER_TEMP_TS;
```

Topics:

3.1 Download the Installer Kit

To download the software as a .zip folder, download the following installer bug IDs for the respective release in sequential order from the My Oracle Support (MOS) as mentioned in the Table 6.

Table 6: Patch Details

Bug ID / Patch Number	Patch Version
35566319	v8.1.2.5.0
35681752	v8.1.2.5.1

Table 6: Patch Details

Bug ID / Patch Number	Patch Version
35736640	v8.1.2.5.2
35991888	v8.1.2.5.3
36129632	v8.1.2.5.4
36169385	v8.1.2.5.5
36234696	v8.1.2.5.6
36368461	v8.1.2.5.7
36531915	v8.1.2.5.8
36658364	v8.1.2.5.9
36499323	v8.1.2.5.10
36819165	v8.1.2.5.11

3.2 Extract the Installer Kit

After downloading the .zip folder, follow these steps to extract the folder contents:

1. Extract the contents of the installer archive file in the download directory using the following command:

   ```
   unzip -a <Compliance_Studio_Installer_Archive_File>.zip
   ```

   The Compliance Studio installer file is extracted, and the `OFS_COMPLIANCE_STUDIO` directory is obtained and is referred to as `<COMPLIANCE_STUDIO_INSTALLATION_PATH>`.

   WARNING Do not rename the application installer directory name after extraction from the archive.

2. Navigate to the download directory where the installer archive is extracted, and assign execute permission to the installer directory using the following command:

   ```
   chmod –R 0755 OFS_COMPLIANCE_STUDIO
   ```

3.3 Hardware and Software Requirements

The following hardware and software are required for this version of Compliance Studio. The installation environment or setup must have these requirements for an application to run smoothly and efficiently.

Topics:

- System Configuration
- Port Numbers for Application
- Prerequisite Environmental Settings
- Download the Big Data Files
- Validation Checklist

- Configure the OpenSearch Component
- Configure Logstash
- Configure the Interpreter Settings
- Create the Hive Schema
- Create the Tablespace
- Create the Studio Schema
- Assign Grants for the Studio Schema
- Create the Sandbox Schema
- Assign Grants for the Sandbox Schema
- Graph Schema
- Entity Resolution

Table 7 lists the Hardware and Software Requirements:

Table 7: Hardware and Software Requirements

Hardware or Software Category	Component Version
Browser	Chrome
Java Version	JDK 11.0.18
Processing Server	• RHEL 7.6+ and 8+
Database Server	• Oracle Database Release 19c (19.3+)Enterprise Edition • Oracle Machine Learning for R (OML4R) (formerly ORE) 1.5.1 with Open source R or Oracle R Distribution 3.6.1
PGX (Graph) Server	• RHEL 7.4+ • Minimum gcc library v4.8.2
OpenSearch Version	2.3.0
Logstash Version	7.16.3 Logstash should be downloaded from the link.
Oracle Instant Client	instantclient-basic-linux.x64-19.8.0.0.0 **NOTE**: The version should be the same as the Database version, and this should be present in the processing server.
Big Data **NOTE:** **This is the prerequisite for legacy ETL which is deprecated in this release and will be removed in the next release. For Spark interpreter, this prerequisite may be required.** You can use either **Cloudera** or open-source **Apache** for a big data cluster. If you are using a Graph pipeline, skip this section. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.	

Table 7: Hardware and Software Requirements

Hadoop and Spark	Kerberos authentication must be enabled for Big Data. • Apache Hadoop Version 3.0.0 • Apache Hive Version 2.1.1 • Apache Spark Version 2.4.0 • Apache Sqoop Version 1.4.7 • The **.profile** file must be present with the SPARK_HOME and PYTHON_HOME parameters already set. **NOTE**: **This is the prerequisite for legacy ETL which is deprecated in this release and will be removed in the next release. For Spark interpreter, this prerequisite may be required.** The product is certified for Apache-Hadoop, and any vendor-specific Hadoop distributions have to confirm compliance with Apache-Hadoop standards, and if not, the vendor the customer chooses to work with for Hadoop should ensure compliance to Apache-Hadoop standards. Any issue raised on vendor-specific distributions will be fixed only if the issue is reproducible on Apache-Hadoop, Apache-Hive, and Apache-Spark.
Hive Connectors	`Hive JDBC Connectors V 2.5.15` **NOTE**: **This is the prerequisite for legacy ETL which is deprecated in this release and will be removed in the next release. For Spark interpreter, this prerequisite may be required.**
Apache	• `Kerberos 1.19.1` • `Hadoop Version 3.0.0` • `Hive Version 3.1.2` • `Spark Version 2.4.8 (with Hadoop)` • `Sqoop Version 1.4.7` **NOTE**: • **This is the prerequisite for legacy ETL which is deprecated in this release and will be removed in the next release. For Spark interpreter, this prerequisite may be required.** • The **.profile** file must be present with the SPARK_HOME and PYTHON_HOME parameters already set. • Kerberos authentication must be enabled for the above services and ensure these services are Apache standards.

Table 7: Hardware and Software Requirements

Apache	• The product is certified for Apache-Hadoop, and any vendor-specific Hadoop distributions must confirm compliance with Apache-Hadoop standards. If not, the vendor, the customer, who chooses to work with Hadoop should comply with the Apache-Hadoop standards. Any issue raised on vendor-specific distributions will be fixed only if the issue is reproducible on Apache-Hadoop, Apache-Hive, and Apache-Spark.
Hadoop Security Protocol	• `Kerberos 5` • `Apache Sentry-2.1.0` **NOTE**: **This is the prerequisite for legacy ETL which is deprecated in this release and will be removed in the next release. For Spark interpreter, this prerequisite may be required.**

3.3.1 System Configuration

1. Log in to the server as a root user.
2. Navigate to UNIX file path `/etc/security/limits.conf` to edit the file.
3. Add the following values at the end of the file for Compliance Studio:

   ```
   <Username> hard nproc 65536
   <Username> soft nproc 65536
   ```

   For example,

   ```
   compliancestudio hard nproc 65536
   compliancestudio soft nproc 65536
   ```

3.3.2 Port Numbers for Application

You can see default port numbers for services in the **install.sh** files in `<Compliance Studio Installation Path>/bin`. To change the port number for service(s) in **install.sh** and in respective service(s) directories, see Appendix A - Change Port Numbers for the Applicable Services section.

3.3.3 Prerequisite Environmental Settings

The following prerequisite environmental settings must be set before beginning the installation of Compliance Studio. These settings are the configuration that a system must have for an application to run smoothly and efficiently.

Table 8 lists the Prerequisite Environmental Settings:

Table 8: Prerequisite Environmental Settings

Category	Expected Value
Java Settings	`PATH` in the `.profile/.bash_profile` file must be set to include the Java Runtime Environment (JDK 11) absolute path. **Supported version**: jdk 11.0.18 **NOTE**: Ensure the absolute path to `JDK/bin` is set at the beginning of the PATH variable. For example: `PATH=/scratch/fccstudio/jdk1.8.0_261/bin:$PATH` Ensure no SYMBOLIC links to Java installation are set in the PATH variable.

PGX Server	The following packages must be installed or present in the server where the PGX service is installed:
	`krb5-libs`
	`krb5-workstation`
	`procps-ng`
	`nc`
	Execute the following command to install the packages as mentioned above:
	`yum install -y krb5-libs krb5-workstation procps-ng nc`
	NOTE:
	• **This is the prerequisite for legacy ETL which is deprecated in this release and will be removed in the next release. For Spark interpreter, this prerequisite may be required.**
	• If you are using a Graph pipeline, skip this configuration/kerberos packages. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

Oracle Database Settings	**Oracle Processing Server** `ORACLE_HOME` must be set in the .profile file pointing to the appropriate Oracle DB Client installation. `PATH` in the `.profile` file must be set to include the appropriate `$ORACLE_HOME/bin` directory.
Download Directory	Indicates the directory where the product installer zip file is downloaded or copied. The user permission must be set to 755 for this directory.
Installation Directory	Indicates the directory where the product installer zip file is extracted, and the installation files are placed. The user permission must be set to 755 for this directory. **NOTE:** The Installation and the Download Directory can be the same if the product installer zip file is not copied separately to another directory.

Table 8: Prerequisite Environmental Settings

Category	Expected Value
OS Locale	Linux: `en_US.utf8` Execute the following command to check the locale: `locale -a | grep -i 'en_US.utf'` The locale is displayed.
Oracle Instant client	Install oracle instant client in the server where compliance Studio is installed and provide the configuration `LD_LIBRARY_PATH` in `config.sh`

3.3.4 Download the Big Data Files

Download the following configuration files from the Big Data server or contact the Big Data Administrator:

NOTE

- **This section is deprecated in the current release and will be removed in the future release.**
- These files must be kept ready and provided in the following file structure used during Compliance Studio installation.
- If you are using a Graph pipeline, skip this section. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

Table 9 lists the required file structure:

Table 9: Required File Structure

File Category	File Names
Hadoop Cluster	• `core-site.xml` • `hive-env.sh` • `hive-site.xml` • `hadoop-env.sh` • `hdfs-site.xml` • `mapred-site.xml` • `yarn-site.xml` • `redaction-rules.json` • `log4j.properties` • `ssl-client.xml` • `topology.map` • `topology.py`
Kerberos Files	• `krb5.conf` • keytab file name as mentioned in the `config.sh` file.

Table 9: Required File Structure

Additional Jars	• `hive-exec-*.jar`. • `HiveJDBC4.jar`. • `hive-metastore-*.jar`. • `hive-service-*.jar`. **NOTE**: • The version of the jars is client or user-specific. These jars can be obtained from the existing jars of the Cloudera installation. • The `HiveJDBC4.jar` file is not available in the Cloudera installation setup. You must download the same from the Cloudera website. This is applicable only for Cloudera Cluster. • For additional jars, see the Appendix C – Additional Jars – PGX and Appendix D – Additional Jars – Batch Service.
OS-Hadoop Jars	`opensearch-spark-20_2.11-2.3.0.jar` To download the `opensearch-spark-20_2.11-2.3.0.jar` file, follow these steps: 1. Download the ZIP file from OpenSearch 2.3.0 2. Extract the downloaded file. 3. Navigate to the dist directory and download the `opensearch-spark-20_2.11-2.3.0.jar` **NOTE:** The version should be the same as the OpenSearch version.

3.3.5 Validation Checklist

The Validation Checklist section provides you with the parameters that you can validate to avoid installation issues. This section explains the validation and actions that can be taken for some of the common parameters that are used in the `config.sh` file for the installation. The parameters that can be validated are as follows:

> **NOTE** If you are using a Graph pipeline, skip Cloudera parameters. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

Table 10 lists the required file structure:

Table 10: Required File Structure

Parameters	Validation
External Service (OFSAA_SERVICE URL)	The OFSAA_Service URL can be validated by clicking the URL for verification.
DB Details for Studio Schema	You can log in to SQL developer and verify the DB Details for Studio Schema.
Compliance Studio Schema Wallet Details	You can verify the Wallet details by reviewing the steps in Verify the Connectivity of the Wallet.

Table 10: Required File Structure

Atomic Wallet Detail	You can verify the Wallet details by reviewing the steps in Setup Password Stores with Oracle Wallet.
Cloudera	You can verify the Cloudera details and validate them by reviewing the steps in Create the Credential Keystore. **NOTE:** **This is the parameter for legacy ETL which is deprecated in this release and will be removed in the next release. For Spark interpreter, this parameter may be required.**
Cloudera (SSH Connection)	Run the command `ssh <hostname of the Cloudera machine>`. You must run this command from the host where the Studio is installed. **NOTE:** **This is the parameter for legacy ETL which is deprecated in this release and will be removed in the next release. For Spark interpreter, this parameter may be required.**
Cloudera (Keytab)	Run the command `kinit -V <KERBEROS_PRINCIPAL> -k -t <KEYTAB_FILEPATH>` to verify the keytab. **NOTE:** **This is the parameter for legacy ETL which is deprecated in this release and will be removed in the next release. For Spark interpreter, this parameter may be required.**

3.3.6 Configure the OpenSearch Component

To configure the OpenSearch component, follow these steps:

NOTE

- Ensure that a minimum of 4GB free RAM space is available for OpenSearch. If RAM is low, the shards of the OpenSearch fail, and the correct result is not fetched.
- You must manually clean the cache if facing a performance issue.
- **Prerequisites**
  - Download the analysis-icu and analysis-phonetic plugins. You can download the plugins from the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/opensearch` directory.
  - Java version should be 11 and above.

1. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/opensearch` directory.
2. Untar the OpenSearch by executing the below command:

```
tar -xvzf opensearch-${<version>}.tar.gz
```

3. Install the following plugins:

```
<COMPLIANCE_STUDIO_INSTALLATION_PATH>/opensearch/opensearch-<version>/
bin/opensearch-plugin install file:////<PATH>/analysis-icu-<version>.zip
```

```
<COMPLIANCE_STUDIO_INSTALLATION_PATH>/opensearch/opensearch-<version>/
bin/opensearch-plugin install file:////<PATH>/analysis-phonetic-
<version>.zip
```

Where PATH specifies location of the plugins.

> **NOTE** You can also install OpenSearch and the plugins on a different machine other than where the Compliance Studio is installed.

4. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/opensearch/opensearch-<version>/config` directory.
5. Configure the `opensearch.yml` with the following variables:

Table 11 lists the parameters of `opensearch.yml` file:

Table 11: opensearch.yml File

Interaction Variable Name	Significance
cluster.name	Indicates the name of the cluster.
node.name	Indicates the name given for the node.
path.data	Indicates the directory where you want to store the data.
path.logs	Indicates the directory where you want to store the logs.
network.host	Indicates the hostname of the machine where you want to install the OpenSearch service.
http.port	Indicates the port number where the OpenSearch service is installed.
discovery.seed_hosts	(Optional) Indicates the hostnames of the nodes of the cluster.
cluster.initial_cluster_manager_nodes	(Optional) Indicates the number given to the nodes of the cluster.

6. Configure the `jvm.options` file as follows:

Table 12 lists Interaction variable names for Configure jvm.options File.

Table 12: Configure jvm.options File

Interaction Variable Name	Significance
-Xms4g	• Set the value for these parameters. • The maximum value set can be up to 50% of the RAM size of the machine. • Recommended value: Less than 32GB.
-Xmx4g	

7. After configuration changes, navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/opensearch/opensearch-<version>/bin` directory.
8. Execute the following commands to configure OpenSearch:

   ```
   nohup ./opensearch &
   ```

   This command is used to start the OpenSearch.

   ```
   tail -f nohup.out
   ```

   This command is used to check the OpenSearch logs.

3.3.6.1 Enable SSL Configuration and Authentication

To enable SSL and Authentication for OpenSearch, configuration is required at both OpenSearch and Compliance Studio.

3.3.6.1.1 OpenSearch Configuration

To configure OpenSearch, follow these steps:

1. Download the opensearch-security plugin zip file.

   For information about how to configure OpenSearch, see the OpenSearch documentation.

3.3.6.1.2 Compliance Studio Configuration

To configure Compliance Studio, follow these steps:

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory.
2. Change the following property in the config.sh file:

   OPEN_SEARCH_USERNAME=admin
3. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/ficdb /bin` directory and encrypt the password (`./FCCM_Studio_Base64Encoder.sh --admin`) using FCCMBASEENCODER64.
4. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/opensearch/opensearch-<version>/config` directory.
5. Execute the following command for generating `admin.p12` file:

   ```
   openssl pkcs12 -export -out admin.p12 -inkey <path to/admin-key.pem> -in <path to/admin.pem>
   ```
6. Execute the following command for generating `ca.crt` file:

   ```
   openssl x509 -outform der -in <path to/admin.pem> -out ca.crt
   ```
7. Copy the `admin.p12` file and place in the following directories:

   `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/load-to-open-search/conf`

   `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/matching-service/conf`
8. Copy the `ca.crt` file and place in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/logstash/config` directory.
9. Configure the following parameters under **OpenSearch Cluster details** in the config.sh file:

   OPEN_SEARCH_ENCRYPTED_PASSWORD='##ENCRYPTED_PASSWORD##'

   OPEN_SEARCH_HTTPS_ENABLED=true

OPEN_SEARCH_TRUSTSTORE_FILE_NAME=admin.p12

OPEN_SEARCH_TRUSTSTORE_PASSWORD=password

> **NOTE** To generate an encrypted password, see the Appendix F - Generate an Encrypted Password for OPenSearch section.

10. Install the Compliance Studio.

3.3.6.2 Cleanup for OpenSearch Indexes

To clean up the OpenSearch indexes, run the following command:

```
curl -XDELETE http://<FULLY QUALIFIED HOSTNAME OF STUDIO SERVER>:<PORT
of Load To Open Search Service>/load-to-open-search/idx/deleteIndex/
<INDEX NAME>
```

For example,

```
curl -XDELETE http://testserver.in.oracle.com:7053/load-to-open-search/
idx/deleteIndex/test_index
```

3.3.7 Configure Logstash

1. Download Logstash tar file from the link.
2. Untar the tar file in one of the Server locations where you are installing Compliance Studio.
3. Create a folder “Logstash” under CS install path.
4. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/Logstash`.
5. Untar the contents of the tar file.
6. Provide this folder path for the parameter “Logstash_Home” in `config.sh` file. The Compliance Studio installer will automatically configure the Logstash properties where necessary.

> **NOTE** The ca.crt file should be copied from the open search server into the `Logstash_Home/config` path when https is enabled in OpenSearch.

3.3.8 Configure the Interpreter Settings

Before installing Compliance Studio, you must configure the interpreter settings for the following interpreters.

> **NOTE** Ensure to configure the settings only for the interpreters that you require.

Table 13 lists the Pre-installation Interpreter Settings:

Table 13: Pre-installation Interpreter Settings

Interpreter	Prerequisite Settings
`jdbc`	For the required configuration, see **jdbc Interpreter** section in the OFS Compliance Studio Administration and Configuration Guide.
`md`	No additional configuration is required.

Table 13: Pre-installation Interpreter Settings

`pgql`	No additional configuration is required.
`pgx-algorithm`	No additional configuration is required.
`pgx-java`	No additional configuration is required.
`pgx-python`	No additional configuration is required. You can point to any other python virtual environment.
`pyspark`	For the required configuration, see Configure the PySpark Interpreter.
`spark`	For the required configuration, see Configure the Spark Interpreter.
`fcc-python`	No additional configuration is required. **NOTE:** **From 8126, virtual environments will no longer be shipped with Compliance Studio and instead conda environment functionality will be used.** **These python interpreters (fcc-python, fcc-python-ml4aml, and fcc-python-sane) will be deprecated and removed in the future release but will remain in case of upgrades.**

3.3.8.1 Configure the Spark Interpreter

3.3.8.1.1 Prerequisites for using the Spark Interpreter

To configure Spark Interpreter, you must download the desired spark distribution from Spark's official website.

For example, spark-2.4.0-bin-hadoop2.7.tgz from the website.

Configure the Spark Interpreter can be used in several situations as follows:

- Connecting to remote spark cluster
  - With/without Kerberos
  - Custom Hadoop client configuration
  - Custom libraries
- Spark in local mode.

In case you want to connect to a remote spark cluster, then obtain the following files:

- Hadoop or Hive client configuration as per your use case
- Kerberos files (if applicable)
  - `krb5.conf`
  - `keytab file`

3.3.8.1.2 Setting up spark-interpreter

The spark interpreter requires spark distribution to start. If you do not intend to use a spark-interpreter, disable the interpreter by performing the following steps:

> **NOTE** If you do not intend to use a spark interpreter, disable the interpreter. You can perform the following steps.

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory.
2. Open the `config.sh` file and set `export SPARK_ENABLED=`**`false`**.

The default configuration of Spark is configured for yarn-client deployment mode with Kerberos enabled remote spark cluster. For local mode, skip this section and follow below.

3.3.8.1.3 Spark Interpreter with remote spark cluster

The Spark Interpreter with remote spark cluster can be performed for the following:

- Configuration with Kerberos enabled remote spark cluster:
  - **krb5.conf**
  - ***.keytab** (For example, **fccstudio.keytab**)
- Configuration with Kerberos disabled remote spark cluster

3.3.8.1.3.1 Configuration with Kerberos enabled remote spark cluster

1. Move the obtained Kerberos files to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/batchservice/user/conf` directory.

   > **NOTE**
   > - These are the same Kerberos files used for ETL.
   > - If extralibs directory does not exists in this path `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmgstudio/interpreter-server/spark-interpreter-<version>/extralibs`, then create it.

2. Place the `spark-<version>-bin-hadoop<version>` files to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/interpreter-server/spark-interpreter-<version>/extralibs` directory.

   For example, `spark-2.4.0-bin-hadoop2.7`

3. Create a folder and name as "**conf**" in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/interpreter-server/spark-interpreter-<version>/extralibs` directory.
4. Place the Hadoop or Hive client configuration files to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/interpreter-server/spark-interpreter-<version>/extralibs/conf` directory.

   > **NOTE** Do not remove the `spark-env.sh` file.

5. Create spark-default.conf and update the spark configurations accordingly. See the Sample spark-default.conf Configuration File section for more information.

6. Update `spark.yarn.dist.files` and `spark.executorEnv.PYTHONPATH`.

> **NOTE** The path must be the same as the path given for the downloaded spark distribution. For example: path for `spark-2.4.0-bin-hadoop2.7spark` distribution.

7. Update the `spark.driver.host`.
8. Update the `spark.yarn.keytab`.
9. Update the `spark.yarn.principal`.
10. If required, you can add an additional spark configuration.

3.3.8.1.3.2 Configuration with Kerberos disabled remote spark cluster:

1. Place the Hadoop or Hive client configuration files to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/interpreter-server/spark-interpreter-<version>/extralibs/conf` directory.

> **NOTE** Do not remove the `spark-env.sh` file.

2. Create `spark-default.conf` and update the spark configurations accordingly. See the Sample spark-default.conf Configuration File section for more information.
3. Update `spark.yarn.dist.files` and `spark.executorEnv.PYTHONPATH`.

> **NOTE** The path must be the same as the path given for the downloaded spark distribution. For example: path for spark-2.4.0-bin-hadoop2.7 spark distribution.

4. Update the `spark.driver.host`.
5. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/interpreter-server/spark-interpreter-<version> /extralibs/conf` directory.
6. Open `spark-default.conf` file and update spark.driver.defaultJavaOptions by removing:

```
"-Dsun.security.krb5.debug=false -
Djavax.security.auth.useSubjectCredsOnly=false -
Djava.security.krb5.conf=<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/
batchservice/user/conf/krb5.conf"
```

3.3.8.1.4 Spark Interpreter in local mode

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/interpreter-server/spark-interpreter-<version> /extralibs/conf` directory.
2. Create `spark-default.conf` and update the spark configurations accordingly. See the Sample spark-default.conf Configuration File section for more information.
3. Open `spark-default.conf` file and update spark.driver.defaultJavaOptions by removing:

```
"-Dsun.security.krb5.debug=false -
Djavax.security.auth.useSubjectCredsOnly=false -
```

```
Djava.security.krb5.conf=<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/
batchservice/user/conf/krb5.conf"
```

4. Set `spark.master as local[*]` in interpreter configuration file.

3.3.8.1.5 Configuration

The Spark interpreter configuration can be divided into the following categories:

- Configuration related to deployment

  These properties can be set either in the Spark libraries, for example, the `spark-defaults.conf` file, or through the system environment variable, **SPARK_CONF**.

  For example, `SPARK_CONF="--conf spark.driver.memory=2g"`.

  > **NOTE** These properties cannot be changed when the Spark interpreter is running.

- Configuration related to Spark runtime control.

  These properties can be set from the Interpreters page of the Compliance Studio application UI. This includes properties such as a `spark.executor.memory`.

  > **NOTE** The properties related to the driver cannot be set during runtime and are considered deployment configurations. The properties related to the executors can be set during runtime. Hence, the latter option of runtime control configuration is preferred.

A list of possible properties is available in the Spark Official Documentation. All the properties prefixed with the term "zeppelin" listed in the Zeppelin Spark Configuration Document can also be set from the Interpreters page of the Compliance Studio application UI.

3.3.8.2 Configure the PySpark Interpreter

Compliance Studio uses PySpark 2.4.0. Before you begin the configurations, check the prerequisites depending on your operation mode.

3.3.8.2.1 Prerequisites

The PySpark interpreter has the same prerequisites as that as the Spark Interpreter. For more information, see Configure the Spark Interpreter. Also, all Spark components must be configured to use the same Python version.

3.3.8.2.2 Configuration

The PySpark interpreter can be configured through the Spark interpreter, with the only exception being the Python version used. By default, the Python version is set to 3 that can be changed either in the interpreter JSON files before the startup or from the **Interpreters** page of the Compliance Studio application UI during runtime by changing the following properties:

To change the value of the `spark.pyspark.python` property before installing the Compliance Studio, follow these steps:

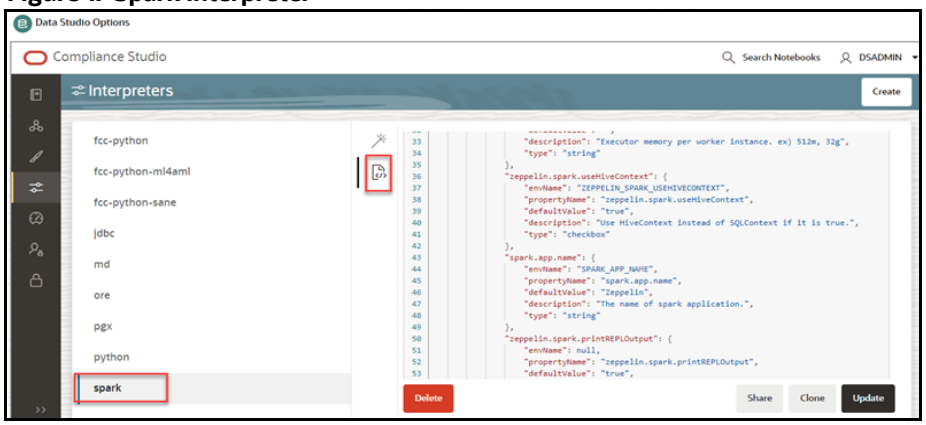
1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/server/builtin/interpreters/spark.json` directory.
2. Update the value in `spark.pyspark.python` property of the `spark.json` file.

To change the value of the `spark.pyspark.python` property after installing the Compliance Studio, follow these steps:

1. Login to the Compliance Studio application.
2. Launch the **CS Production** Workspace.
3. Hover the mouse over the **Data Studio Options** widget and Click **Interpreters**.

   By default, the Interpreters page lists all the available interpreters on the LHS.

Figure 1: Spark Interpreter



4. Click **spark** interpreter on the LHS and then click **Plain Configuration** on the RHS.
5. Update the value in the `spark.pyspark.python` property and click **Update**.

In the **Spark Interpreter Settings** page of the Compliance Studio application UI (or `spark.json` file), change the value of the `spark.pyspark.python` property to the Python executable path that is to be used by the Spark executors.

In the **PySpark Interpreter Settings** page of the Compliance Studio application UI (or `pyspark.json` file), change the value of the `zeppelin.pyspark.python` property to the Python executable path that is to be used by the Spark driver.

3.3.8.2.3 Use the Python Virtual Environments with PySpark

To ensure that the two Python versions match, in case your components run on different machines, you must use the Python virtual environments with PySpark.

To use Python Virtual Environments with PySpark, follow these steps:

1. Create a Virtual Environment with Conda
2. Update the Interpreter Properties

3.3.8.2.3.1 Create a Virtual Environment with Conda

> **NOTE** You can also use **virtualenv** to create your virtual environment instead of **conda**.

To create a virtual environment with Conda, follow these steps:

1. Ensure that you have conda and conda-Pack installed.
2. Create your virtual environment using the following command:

```
conda create -y -n <environment-name> python=<python-version>
<additional-packages>
```

> **NOTE** The <environment-name> can be chosen freely and subsequently has to be substituted in further commands.

3. Activate your virtual environment using the following command:

   ```
   conda activate <environment-name>
   ```

4. Execute the following to obtain the path to your virtual environment:

   ```
   which python
   ```

   The obtained result is referred to as `<environment-abs-path>`.

5. Compress your virtual environment using the following command:

   ```
   conda pack -n <environment-name> -o <environment-abs-path>/<environment-
   name>.tar.gz
   ```

3.3.8.2.3.2 Update the Interpreter Properties

The interpreter properties can either be configured in the interpreter JSON files or from the Interpreters page of the Compliance Studio application UI after starting the Compliance Studio application.

- In the **Spark Interpreter Settings** page of the Compliance Studio application UI (or `spark.json`), change the following:
  - Change the value of the `spark.yarn.dist.archives` property to `<environment-abs-path>/<environment-name>.tar.gz#<environment-name>`
  - Change the value of the `spark.pyspark.python property to ./<environment-name>/bin/python`
- In the **PySpark Interpreter Settings** page of the Compliance Studio application UI (or `pyspark.json`), change the value of the `zeppelin.pyspark.python` parameter to `<environment-abs-path>/bin/python`.

3.3.9 Create the Hive Schema

> **NOTE**
> - **This section is deprecated in the current release and will be removed in the future release.**
> - If you are using a Graph pipeline, skip this section. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

To create a hive schema, perform the following steps:

1. Login to the server where **cloudera/hive** is installed.
2. Open a hive session in the command prompt.

   ```
   hive
   ```

3. Create a new hive schema using the following script:

   ```
   create database <hive schema name>;
   ```

4. Use the hive schema that is created by the following command:

   ```
   Use <hive schema name>
   ```

   A new hive schema is created.

3.3.10 Create the Tablespace

To create a tablespace in the Oracle Database using the script as described in the Table 14.

Table 14: Create Tablespace

User	Script
AIF_USER_TEMP_TS	CREATE TABLESPACE AIF_USER_TEMP_TS DATAFILE '<Datafile Path>' SIZE <size in byte> REUSE AUTOEXTEND ON NEXT <size in megabyte> MAXSIZE UNLIMITED;
AIF_USER_TS	CREATE TABLESPACE AIF_USER_TS DATAFILE '<Datafile Path>' SIZE <size in byte> REUSE AUTOEXTEND ON NEXT <size in megabyte> MAXSIZE UNLIMITED;
<CS_USER_TS>	CREATE TABLESPACE <CS_USER_TS> DATAFILE '<Datafile Path>' SIZE <size in byte> REUSE AUTOEXTEND ON NEXT <size in megabyte> MAXSIZE UNLIMITED;
<GRAPH_SCHEMA_TS>	CREATE TABLESPACE <GRAPH_SCHEMA_TS> DATAFILE '<Datafile Path>' SIZE <size in byte> REUSE AUTOEXTEND ON NEXT <size in megabyte> MAXSIZE UNLIMITED;

NOTE The tablespace size should be defined based on the size of the data.

3.3.11 Create the Studio Schema

To create a studio schema, create a new Oracle Database schema user using the following script:

```
CREATE USER <STUDIO SCHEMA USER> IDENTIFIED BY <PASSWORD> DEFAULT
TABLESPACE <STUDIO TABLESPACE>;

ALTER USER <STUDIO SCHEMA USER> QUOTA 2000M ON <STUDIO TABLESPACE>;

ALTER USER <STUDIO SCHEMA USER> QUOTA <SIZE IN MEGABYTE> ON AIF_USER_TS;
```

For example;

```
ALTER USER CS812_USER QUOTA 500M ON AIF_USER_TS;
```

> **NOTE** The tablespace and quota sizes should be defined based on the size of the data.

A new Oracle Database schema (Studio schema) is created.

3.3.12 Assign Grants for the Studio Schema

Grant the following permissions to the newly created Oracle Database studio schema:

- GRANT CREATE SESSION TO <STUDIO SCHEMA USER>;
- GRANT CREATE TABLE TO <STUDIO SCHEMA USER>;
- GRANT CREATE VIEW TO <STUDIO SCHEMA USER>;
- GRANT CREATE ANY TRIGGER TO <STUDIO SCHEMA USER>;
- GRANT CREATE ANY PROCEDURE TO <STUDIO SCHEMA USER>;
- GRANT CREATE SEQUENCE TO <STUDIO SCHEMA USER>;
- GRANT EXECUTE ON DBMS_RLS TO <STUDIO SCHEMA USER>;
- GRANT EXECUTE ON SYS.DBMS_SESSION TO <STUDIO SCHEMA USER>;
- GRANT CREATE SYNONYM TO <STUDIO SCHEMA USER>;
- GRANT EXECUTE ON DBMS_REDEFINITION TO <STUDIO SCHEMA USER>;
- GRANT REDEFINE ANY TABLE TO <STUDIO SCHEMA USER>;
- GRANT CREATE MATERIALIZED VIEW TO <STUDIO SCHEMA USER>;
- GRANT SELECT ON SYS.V_$PARAMETER TO <STUDIO SCHEMA USER>;
- GRANT SELECT ON SYS.DBA_FREE_SPACE TO <STUDIO SCHEMA USER>;
- GRANT SELECT ON SYS.DBA_TABLES TO <STUDIO SCHEMA USER>;
- GRANT SELECT ON SYS.DBA_TAB_COLUMNS TO <STUDIO SCHEMA USER>;
- GRANT CREATE RULE TO <STUDIO SCHEMA USER>;
- GRANT DROP ANY TRIGGER TO <STUDIO SCHEMA USER>;
- GRANT SELECT ON SYS.DBA_RECYCLEBIN TO <STUDIO SCHEMA USER>;
- GRANT CREATE JOB TO <STUDIO SCHEMA USER>;
- GRANT EXECUTE ON DBMS_LOCK TO <STUDIO SCHEMA USER>;
- GRANT EXECUTE ON DBMS_STATS TO <STUDIO SCHEMA USER>;
- GRANT ANALYZE ANY TO <STUDIO SCHEMA USER>;
- GRANT CREATE TYPE TO <STUDIO SCHEMA USER>;

- `GRANT EXECUTE ON CTXSYS.CTX_DDL TO <STUDIO SCHEMA USER>;`

> **NOTE** The following grants should be revoked after the successful installation of Compliance Studio:
>
> - `REVOKE SELECT ON SYS.DBA_RECYCLEBIN FROM <STUDIO SCHEMA USER>;`
> - `REVOKE SELECT ON SYS.DBA_FREE_SPACE FROM <STUDIO SCHEMA USER>;`

3.3.13 Create the Sandbox Schema

> **NOTE** This section is applicable for ML4AML features except ASC.

To create a sandbox schema, create a new Oracle Database sandbox schema user using the following script:

```
create user <SANDBOX SCHEMA USER>
IDENTIFIED BY <PASSWORD>
default tablespace AIF_USER_TS
temporary tablespace TEMP
profile DEFAULT
quota unlimited on AIF_USER_TS
quota unlimited on AIF_USER_TEMP_TS
```

> **NOTE**
>
> - The sandbox will always be on a different database other than the production schema.
> - After creating a user for the sandbox schema, you must create a sandbox workspace. To create a sandbox workspace, see **Creating a Sandbox Workspace** section in the OFS Compliance Studio Administration and Configuration Guide.
> - The tablespace and quota sizes should be defined based on the size of the data.

A new Oracle Database schema (Sandbox schema) is created.

3.3.14 Assign Grants for the Sandbox Schema

Grant the following permissions to the newly created Oracle Database sandbox schema:

- `GRANT CONNECT TO <SANDBOX SCHEMA USER>;`
- `GRANT CREATE SESSION TO <SANDBOX SCHEMA USER>;`
- `GRANT CREATE PROCEDURE TO <SANDBOX SCHEMA USER>;`
- `GRANT CREATE SEQUENCE TO <SANDBOX SCHEMA USER>;`
- `GRANT CREATE TABLE TO <SANDBOX SCHEMA USER>;`

- GRANT CREATE TRIGGER TO <SANDBOX SCHEMA USER>;
- GRANT CREATE VIEW TO <SANDBOX SCHEMA USER>;
- GRANT CREATE MATERIALIZED VIEW TO <SANDBOX SCHEMA USER>;
- GRANT CREATE SYNONYM TO <SANDBOX SCHEMA USER>;
- GRANT CREATE RULE TO <SANDBOX SCHEMA USER>;
- GRANT CREATE ANY TRIGGER TO <SANDBOX SCHEMA USER>;
- GRANT DROP ANY TRIGGER TO <SANDBOX SCHEMA USER>;
- GRANT CREATE ANY TYPE TO <SANDBOX SCHEMA USER>;

3.3.15 Graph Schema

3.3.15.1 Create Graph Schema and Grant Permission

To create a Graph schema, create a new Oracle Database schema user using the following script:

```
CREATE USER <GRAPH SCHEMA USER> IDENTIFIED BY <PASSWORD> DEFAULT TABLESPACE
<GRAPH_SCHEMA_TS>;

ALTER USER <GRAPH SCHEMA USER> QUOTA 2000M ON <GRAPH_SCHEMA_TS>;
```

For example;

```
ALTER USER GRAPH_SCHEMA_USER QUOTA 500M ON GRAPH_SCHEMA_TS;
```

> **NOTE** The tablespace and quota sizes should be defined based on the size of the data.

A new Oracle Database schema (Graph schema) will be created.

To assign grants, run the following:

- **Pre-installation Grants for both BD and ECM Graphs**

  Change the <GRAPH_SCHEMA> to the underlying schema of the data source of the graph pipeline.

  - GRANT ANALYZE ANY TO <GRAPH_SCHEMA>;
  - GRANT CREATE SESSION TO <GRAPH_SCHEMA>;
  - GRANT CREATE TABLE TO <GRAPH_SCHEMA>;
  - GRANT CREATE VIEW TO <GRAPH_SCHEMA>;
  - GRANT CREATE ANY PROCEDURE TO <GRAPH_SCHEMA>;
  - GRANT CREATE SEQUENCE TO <GRAPH_SCHEMA>;
  - GRANT CREATE JOB TO <GRAPH_SCHEMA>;
  - GRANT CREATE MATERIALIZED VIEW TO <GRAPH_SCHEMA>;
  - GRANT EXECUTE ON DBMS_SCHEDULER TO <GRAPH_SCHEMA>;
  - GRANT EXECUTE ON DBMS_COMPARISON TO <GRAPH_SCHEMA>;
  - GRANT EXECUTE ON DBMS_RLS TO <GRAPH_SCHEMA>;
  - GRANT EXECUTE ON SYS.DBMS_SESSION TO <GRAPH_SCHEMA>;

- GRANT EXECUTE ON DBMS_REDEFINITION TO <GRAPH_SCHEMA>;
- GRANT REDEFINE ANY TABLE TO <GRAPH_SCHEMA>;
- GRANT SELECT ON SYS.V_$PARAMETER TO <GRAPH_SCHEMA>;
- GRANT EXECUTE ON DBMS_STATS TO <GRAPH_SCHEMA>;
- GRANT EXECUTE ON DBMS_ISCHED TO <GRAPH_SCHEMA>;
- GRANT EXECUTE ON DBMS_PARALLEL_EXECUTE TO <GRAPH_SCHEMA>;

- **Pre-installation Grants for BD Graph**

  Change the <BD_ATOMIC_SCHEMA> to the underlying schema of the data source of the BD graph pipeline.

  NOTE The following grants are applicable for the Out-of-the-box graph pipeline only. If the user has to execute the custom graph, the same permissions have to be provided for the input tables referred in Custom Graph Pipeline.

  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.ACCT TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.CUST_ACCT TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.ACCT_BAL_POSN_SMRY TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.CUST TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.BACK_OFFICE_TRXN TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.EMP TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.CUST_CUST TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.KDD_CAL TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.KDD_REVIEW TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.EMP_ACCT TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.ACCT_SMRY_MNTH TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.ACCT_ADDR TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.EXTERNAL_ENTITY TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.CUST_EMAIL_ADDR TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.CUST_PHON TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.CUST_ADDR TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.CASH_TRXN TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.LINK_STAGE TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.WIRE_TRXN TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.MI_TRXN TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.INSTN_MASTER TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <BD_ATOMIC_SCHEMA>.EXTERNAL_ENTITY_ADDR TO <GRAPH_SCHEMA>;

- GRANT SELECT ON <BD_ATOMIC_SCHEMA>.DERIVED_ADDRESS TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <BD_ATOMIC_SCHEMA>.CLIENT_BANK TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <BD_ATOMIC_SCHEMA>.EXTERNAL_ENTITY_LINK TO <GRAPH_SCHEMA>;
- GRANT ANALYZE ANY TO <GRAPH_SCHEMA>;

- **Pre-installation Grants for ECM Graph**

  Change the <ECM_ATOMIC_SCHEMA> to the underlying schema of the data source of the ECM graph pipeline.

  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_ACCT TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_CUST_ACCT TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_ACCT_BAL_POSN_SMRY TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_CUST TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_BACK_OFFICE_TRXN TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_EMP TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_CUST_CUST TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_EMP_ACCT TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_ACCT_SMRY_MNTH TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_ACCT_ADDR TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_EXTERNAL_ENTITY TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_CUST_EMAIL_ADDR TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_CUST_PHON TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_CUST_ADDR TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_CASH_TRXN TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_WIRE_TRXN TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_MI_TRXN TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_INSTN_MASTER TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_EXTERNAL_ENTITY_ADDR TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_DERIVED_ADDRESS TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_CLIENT_BANK TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_EXTERNAL_ENTITY_LINK TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.KDD_CASES TO <GRAPH_SCHEMA>;

- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.KDD_CASE_ACCOUNTS TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_SCENARIO_MASTER TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_EVENTS TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_EVENT_DETAILS TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_EVENT_ENTITY_MAP TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_PRECASE_CASE_MAP TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.KDD_CASE_CUSTOMERS TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.KDD_CASE_EXTERNAL_ENTITY TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.KDD_CASE_INSTN_MASTER TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_CORRELATION_EVENT_MAP TO <GRAPH_SCHEMA>;
- GRANT EXECUTE ON DBMS_SCHEDULER TO <ECM_ATOMIC_SCHEMA>;
- GRANT EXECUTE ON DBMS_ISCHED TO <ECM_ATOMIC_SCHEMA>;
- GRANT EXECUTE ON DBMS_PARALLEL_EXECUTE TO <ECM_ATOMIC_SCHEMA>;
- GRANT CREATE JOB TO <ECM_ATOMIC_SCHEMA>;

- **Post-installation Grants for BD Graph**

> **NOTE** The following grants should be added after completed steps mentioned in the Importing OOB Graph Definition and related Metadata section.

- GRANT EXECUTE ON <BD_ATOMIC_SCHEMA>.P_FCC_CS_BD_EXTERNAL_ENTITY TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <BD_ATOMIC_SCHEMA>.FCC_CS_BD_EXTERNAL_ENTITY TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <BD_ATOMIC_SCHEMA>.FCC_CS_BD_DERIVED_GROUP TO <GRAPH_SCHEMA>;

- **Post-installation Grants for ECM Graph**

> **NOTE** The following grants should be added after completed steps mentioned in the Importing OOB Graph Definition and related Metadata section.

- GRANT EXECUTE ON <ECM_ATOMIC_SCHEMA>.P_FCC_CS_CM_EXTERNAL_ENTITY TO <GRAPH_SCHEMA>;
- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_CS_CM_EXTERNAL_ENTITY TO <GRAPH_SCHEMA>;

- GRANT SELECT ON <ECM_ATOMIC_SCHEMA>.FCC_CS_CM_DERIVED_GROUP TO <GRAPH_SCHEMA>;

- **Post-installation Grants for both BD and ECM Graphs**
  - GRANT SELECT, INSERT, UPDATE, DELETE ON <STUDIO_SCHEMA>.FCC_GRAPH_M_TRXN_VIEWS TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <STUDIO_SCHEMA>.FCC_M_TABLES TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <STUDIO_SCHEMA>.FCC_M_COLUMNS TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <STUDIO_SCHEMA>.FCC_M_ATTRIBUTE TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <STUDIO_SCHEMA>.FCC_M_ATTRIBUTE_COLUMN_MAP TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <STUDIO_SCHEMA>.FCC_M_COLUMNS_DETAILS TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <STUDIO_SCHEMA>.FCC_M_MAP TO <GRAPH_SCHEMA>;
  - GRANT SELECT ON <STUDIO_SCHEMA>.MMG_GRAPH_SCHEMA TO <GRAPH_SCHEMA>;

3.3.15.2 Create Wallet for Graph Schema

See **step 4** in the Setup the Password Stores for Database User Accounts section.

> **NOTE**
> - Graph schema must be in the same database where Compliance Studio Schema is exists.
> - You can refer only one Graph schema in Compliance Studio.

3.3.15.3 Configure Resource XML

See the Configure the resources.xml for Multiple ER Schemas section for more details.

3.3.16 Entity Resolution

3.3.16.1 Create Entity Resolution Schema and Grant Permission

> **NOTE** ER schema is nothing but FSDF schema and creation of ER schema can be skipped if the FSDF schema is already available. The grants mentioned in this section are required.

To create ER schema, create a new Oracle Database schema user using the following script:

```
CREATE USER <ER SCHEMA USERNAME> IDENTIFIED BY <PASSWORD>;
```

A new Oracle Database schema (ER schema) will be created.

To assign grants, see the Assign Grants for the Studio Schema section.

3.3.16.2 Create a wallet for ER/FSDF schema

See **step 4** in the Setup the Password Stores for Database User Accounts section.

> **NOTE**
> - ER schema can be in the same database where CS is installed or a different database.
> - You can create multiple ER schemas.

3.3.16.3 Configure Resource XML

See the Configure the resources.xml for Multiple ER Schemas section for more details.

3.3.16.4 Configure ER schema Profile

Set the SESSIONS_PER_USER limit to UNLIMITED for ER Schema by executing the below steps:

1. Get the ER schema profile by executing the below query:

   ```
   select profile from dba_users where username ='<ER SCHEMA USERNAME>';
   ```

2. Change the profile which is obtained from the step 1 by executing the below query:

   ```
   ALTER PROFILE <profile> LIMIT SESSIONS_PER_USER UNLIMITED;
   ```

3.3.16.5 Post Upgrade Steps in case ER Batches are Executed before Upgrade

To upgrade the ER/FSDF Schema if ER batches are executed before upgrade, follow these steps:

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/ficdb/upgrade/8.1.2.5.0/UpgradeFSDFSchema`.

   > **NOTE** The below steps are required only if Entity Resolution Jobs are executed before the v8.1.2.5.0 upgrade.

2. Log in to ER/FSDF Schema.
3. Execute the script `UpdateTempMappingTable.sql`.

   This script will update a new **v_persisted_id** column in the **fcc_er_temp_mapping table** and it is required to take manual action in the Merge & Split UI.
4. Execute the script `CreateSystemGuidTables.sql`.

   This script will create tables for persisting GUIDs in manual action or while running the batches.
5. Execute the script `AlterViewTableIndex.sql`.

   This script will create an index on the view table which is required for search in the Merge & Split UI.
6. Execute the script `UpdateERConfigTable.sql`.

   This script will add new configurations in the ER config table which are required for search in the Merge & Split UI.

3.4 Setup Password Stores with Oracle Wallet

As part of an application installation, administrators must set up password stores for database user accounts using Oracle Wallet. These password stores must be installed on the application database

side. The installer handles much of this process. The administrators must perform some additional steps.

A password store for the application and application server user accounts must also be installed; however, the installer takes care of this entire process.

Topics:

3.4.1 Setup the Password Stores for Database User Accounts

After the database is installed and the default database user accounts are set up, administrators must set up a password store using the Oracle Wallet. This involves assigning an alias for the username and associated password for each database user account. The alias is used later during the application installation. This password store must be created on the system where the application server and database client are installed.

This section describes the steps to create a wallet and the aliases for the database user accounts. For more information on configuring authentication and password stores, see Oracle Database Security Guide.

> **NOTE** In this section, `<wallet_location>` is a placeholder text for illustration purpose. Before running the command, ensure that you have already created the <wallet_location> directory where you want to create and store the wallet.

To create a wallet, follow these steps:

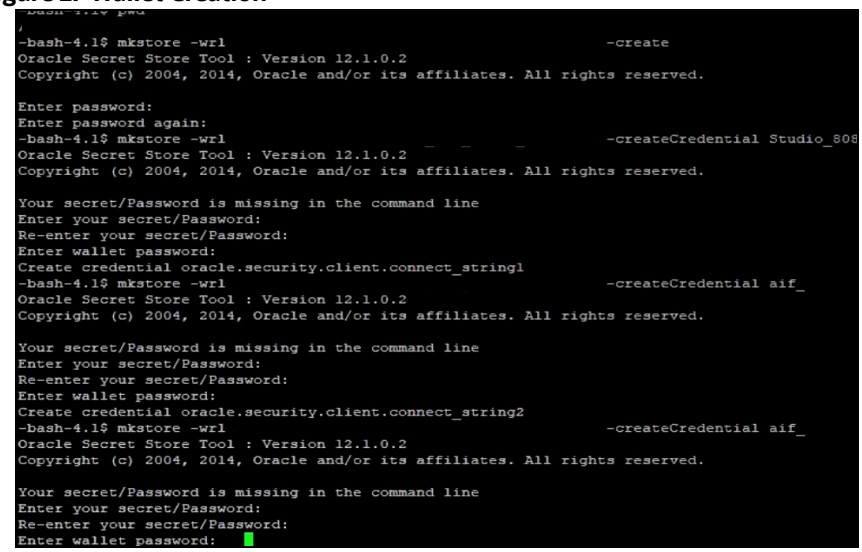
1. Log in to the server as a Linux user.
2. Create a wallet in the <wallet_location> using the following command:

   ```
   mkstore -wrl <wallet_location> -create
   ```

   > **NOTE** The mkstore utility is included in the Oracle Database Client installation.

3. After you run the command, a prompt appears. Enter a password for the Oracle Wallet in the prompt.

Figure 2: Wallet Creation



4. Create the database connection credentials for the studio schema/ER Schema alias using the following command:

   ```
   mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name>
   ```

   After you run the command, a prompt appears. Enter the password associated with the database user account in the prompt. You are prompted to re-enter the password. You are prompted for the wallet password used in Step 1.

5. Create the database connection credentials for the atomic schema alias using the following command:

   ```
   mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name>
   ```

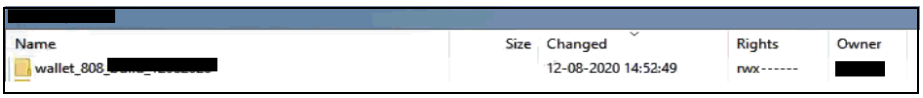
> **NOTE** Creating an atomic schema is not required when installing Compliance Studio without OFSAA.

In this manner, create a wallet and associated database connection credentials for all the database user accounts.

The wallet is created in the <wallet_location> directory with the auto-login feature enabled. This feature enables the database client to access the wallet contents without using the password. For more information, see Oracle Database Security Guide.

After the wallet is created, go to the <wallet_location> directory and click **Refresh**  to view the created wallet folder.

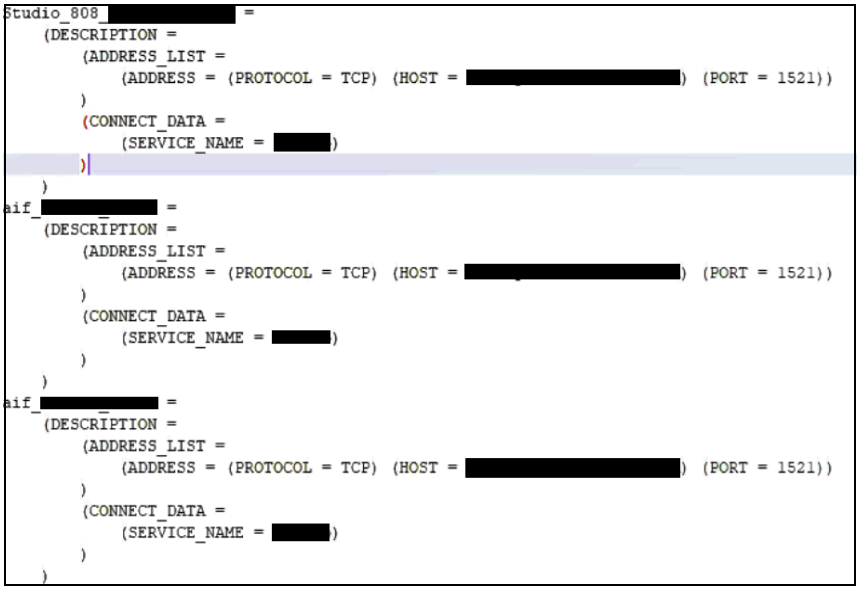
Figure 3: Location of the Created Wallet Folder



The wallet folder contains two files: **ewallet.p12** and **cwallet.sso**.

6. In the <wallet_location> directory, configure the **tnsnames.ora** file to include the entry for each alias name to be set up.

Figure 4: Location of the Created Wallet Folder



NOTE

- You can either update the existing tnsnames.ora file with the above details or create new tnsnames.ora file and add the required entries.
- <alias-name> is a user-defined value.

7. Create a **sqlnet.ora** file in the wallet directory using the following content:

```
WALLET_LOCATION = (SOURCE = (METHOD = FILE) (METHOD_DATA = (DIRECTORY = <Wallet_Location>)) )

SQLNET.WALLET_OVERRIDE=TRUE

SSL_CLIENT_AUTHENTICATION=FALSE
```

3.4.2 Verify the Connectivity of the Wallet

To verify the connectivity of the wallet, follow these steps:

1. Create a sqlnet.ora file in the wallet directory using the following content:

```
WALLET_LOCATION = (SOURCE = (METHOD = FILE) (METHOD_DATA = (DIRECTORY = <Wallet_Location>)) )

SQLNET.WALLET_OVERRIDE=TRUE

SSL_CLIENT_AUTHENTICATION=FALSE
```

2. Test the connectivity using the following command:

NOTE The ORACLE_HOME used with the wallet must be the same version or higher than the wallet created.

```
$ export WALLET_LOCATION=<wallet_location>
```

```
$ export TNS_ADMIN=<tnsnames.ora_location>. If you have created a new
tnsnames.ora file, provide the location of the new file.

$ sqlplus /@<alias_name>
```

The output is similar to:

```
SQL*Plus: Release 11

Connected to:

Oracle Database 12c

To verify if you are connected to the correct user:

SQL> show user

The output is similar to:

USER is "<database-user-name>"
```

3.4.3 Create the Credential Keystore

> **NOTE**
> - **This section is deprecated in the current release and will be removed in the future release.**
> - If you are using a Graph pipeline, skip this section. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

Credential keystore must be created for the Behavior Detection (BD) or Enterprise Case Management (ECM) **Atomic schema** and **Compliance Studio Schema**. To create a credential keystore, follow these steps:

1. Login as HDFS Superuser.
2. Create a credential keystore on HDFS using the following command:

   ```
   hadoop credential create mydb.password.alias -provider jceks://hdfs/
   user/root/oracle.password.jceks
   ```

3. Verify the credential keystore file using the following command:

   ```
   hadoop credential list -provider jceks://hdfs/user/root/
   oracle.password.jceks
   ```

4. Grant Read permission to the keystore file using the following command:

   ```
   hadoop fs -chmod 744 /user/root/oracle.password.jceks
   ```

> **NOTE** Ensure the credential keystore file path and the alias are correctly mentioned in the `config.sh` file.

3.4.3.1 Copying and Adding Files

NOTE

- **This section is deprecated in the current release and will be removed in the future release.**
- If you are using a Graph pipeline, skip this section. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

To copy the jar files, follow these steps:

1. Create the folder in the `GRAPH_FILES_PATH` parameter in a node of the big data cluster.
2. Create a folder called jars inside the folder that is created in the previous step.
3. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/ficdb/etlJars directory` and copy all the jars in this directory into the big data node inside the folder jars.
4. Perform this step if https is enabled for OpenSearch:
   a. Create a new folder with the name as **conf** in the **GRAPH_FILES_PATH** parameter in a node of the big data cluster.
   b. Place the **admin.p12** file in the **conf** folder.

NOTE

To use the OS-Hadoop connector, download the `commons-httpclient-3.0.1.jar` and `opensearch-spark-20_2.11-2.3.0.jar` (depending on which OpenSearch version is used) files and place them in the jars folder.

This is applicable only in the case of ETL for Graph.

3.4.3.2 Create Credential Keystore for OpenSearch

Credential keystore must be created for the OpenSearch if https is enabled for OpenSearch.

NOTE

- **This section is deprecated in the current release and will be removed in the future release.**
- If you are using a Graph pipeline, skip this section. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

To create a credential keystore, follow these steps:

1. Login as HDFS Superuser.
2. Create a credential keystore on HDFS using the following command:

```
hadoop credential create open.password.alias -value <Open search password> \
-provider jceks://hdfs/user/fccstudio/open/open.password.jceks
hadoop credential create open.keystore.password.alias -value password \
-provider jceks://hdfs/user/fccstudio/open/open.password.jceks
Where,
```

   - `open.password.alias` is the OpenSearch password alias name

- `open.keystore.password.alias` is the OpenSearch keystore password alias name
- `<open search password>` is OpenSearch password
- `password` is OpenSearch keystore password
- `hdfs/user/fccstudio/open/open.password.jceks` is the file path of the credential keystore

3. Verify the credential keystore file using the following command:

```
hadoop credential list -provider jceks:// hdfs/user/fccstudio/open/
open.password.jceks.
```

4. Grant Read permission to the keystore file using the following command:

```
hadoop fs -chmod 744 /user/fccstudio/open/open.password.jceks
```

NOTE

- Ensure the credential keystore file path and the alias are correctly mentioned in the `config.sh` file.
- The version of the open search jar should be the same as the version of OpenSearch installed.

4 Installation

Perform the following steps to complete the installation:

4.1 Place Files in the Installation Directories

> **NOTE** If you are using a Graph pipeline, skip **steps 1** and **2**. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

To place the required jars and Kerberos files in the required locations, follow these steps:

1. To place the additional jar files, follow these steps:

   > **NOTE** This step is deprecated in the current release and will be removed in the future **release.**

   a. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/batchservice/user/lib` directory.

   b. Place the following additional jar files:

   - `hive-exec-*.jar. For example, hive-exec-1.1.0.jar.`
   - `HiveJDBC4.jar`
   - `hive-metastore-*.jar. For example, hive-metastore-1.1.0.jar.`

- `hive-service-*.jar. For example, hive-service-1.1.0.jar.`

For additional jars, see the Appendix C – Additional Jars – PGX and Appendix D – Additional Jars – Batch Service sections.

NOTE

- The version of the jars is client or user-specific. These jars can be obtained from the existing jars of the Cloudera installation.
- The `HiveJDBC4.jar` file is not available in the Cloudera setup. You must download the same from the Cloudera website.
- **For upgrade scenario**: The libraries available in the previous version of Compliance Studio can be copied and placed in the new location. To place the libraries, follow step 1.

2. To place the Kerberos files, follow these steps:

NOTE This step is deprecated in the current release and will be removed in the future **release.**

a. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/batchservice/user/conf` directory.

b. Place the following Kerberos files:

- `krb5.conf`
- keytab file name as mentioned in the `config.sh` file.

NOTE **For upgrade scenario**: The files available in the previous version of Compliance Studio can be copied and placed in the new location. To place the files, follow step 2.

3. Perform this step if https is enabled for OpenSearch:

a. Copy `admin.p12` file from `<OpenSearch_Installation_path>//config` directory.

b. Place the `admin.p12` file in `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/matching-service/conf` directory.

c. Place the `admin.p12` file in `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/load-to-open-search/conf` directory.

4.2 Add Synonyms and Stopword files in OpenSearch

NOTE **For upgrade scenario**: The Synonyms and Stopword files can be re-utilized from the previous Compliance Studio version.

To consider the similarity when performing the OpenSearch, you can add the synonyms and stopword files in the OpenSearch.

To add synonyms and stopword files in OpenSearch, perform the following steps:

1. Create a folder in the name of “analysis” in the `<OpenSearch_Installation_path>/config` directory.

2. You can add your synonyms and stopwords to these files and place the files in the analysis folder:
   - Cardinal_ordinal.txt
   - Country.txt
   - Gender.txt
   - Namestop.txt
   - Name_synonym.txt
   - Organisation_strip.txt
   - Organisation_suffix.txt
   - Synonym.txt
   - Title.txt

NOTE
- User can decide to provide any data in the Stopword or Synonym files.
- Each Stopword must be provided in a separate line.
- All related synonyms must be provided in the same line, separated by a comma.
- All the synonyms must be provided in the same line and ensure that there are no repetitions of the synonym. For Example, rob, robi, robie, roby, robbi.

4.3 Place Files in Wallet

To place the files in the wallet in the required locations, follow these steps:

1. To place the files in the wallet, follow these steps:
   a. Navigate to <COMPLIANCE_STUDIO_INSTALLATION_PATH>.
   b. Create a folder 'wallet' and place the following files.
   c. Navigate to <COMPLIANCE_STUDIO_INSTALLATION_PATH>/wallet.
   d. Place the following files, which are being generated from the <wallet_directory> in the section Setup the Password Stores for Database User Accounts:
      - tnsnames.ora
      - ewallet.p12
      - cwallet.sso
      - ewallet.p12.lck
      - cwallet.sso.lck

NOTE This folder path will be referred to as "WALLET_LOCATION" and "TNS_ADMIN_PATH" in config.sh while configuring Compliance Studio. If you want to maintain tnsname.ora in a different folder, then "TNS_ADMIN_PATH" will be that folder location.

2. Place the **sqlnet.ora** file into the wallet and update the path for the current wallet location.

```
WALLET_LOCATION =
    (SOURCE =
        (METHOD = FILE)
        (METHOD_DATA =
            (DIRECTORY = <wallet location>)
        )
    )
SQLNET.WALLET_OVERRIDE = TRUE
```

4.4 Generate the Public and Private Keys

The Public and Private keys are JSON Web Tokens (JWT) that are generated for Authentication from Compliance Studio.

To generate the keys, follow these steps:

> **NOTE** The following steps are mandatory for the first-time Compliance Studio installation.

1. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-home/bin` directory.
2. Run the Shell Script `./key-generator.sh` from the directory.

   The Public and Private Keys are generated and available in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-home/conf` directory.
3. Copy the `private.key` and `public.key` files to the following paths:
   - `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-home/mmg-studio/conf` directory.
   - `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/batchservice/conf` directory.
   - `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/pgx/pgx-server/conf` directory.
   - `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-home/mmg-load-to-graph/graph-service/conf` directory.
   - `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-home/mmg-ui/conf` directory.

4.5 Generate API token for CS API User

To generate the API token, follow these steps:

1. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-home/bin` directory.
2. Run the shell script: `./token-generator.sh CS_API_USER`

   This will generate the API token in the same directory as `token.out`.

   The generated token that is required while configuring `config.sh` file in the path `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin`.

4.6 Generate Compliance Studio Server SSL Configuration Mandatory File

Topics:

4.6.1 Generate Self-signed Certificate

To generate the self-signed certificate, perform the following steps:

1. Run the following jks command in the Studio Server:

   ```
   keytool -J-Dkeystore.pkcs12.legacy -genkey -alias <alias> -keyalg RSA -
   keystore <alias>.jks -dname "CN=<hostname>, OU=OR, O=OR L=OR, ST=OR,
   C=OR" -ext "SAN=IP:<ip address 1>,IP:<ip address 2>"
   ```

   NOTE
   - ip address 2 is optional and hostname is the fully qualified host name.
   - You must use the same password and alias that is provided in the `config.sh` file.

2. Specify the keystore password.
3. When generating the keytool ensure to provide the hostname in the first name. For example:

   Question: What is your first and last name?

   Answer: Provide the fully qualified studio server hostname.

   For example, <hostname>.<domain name>
4. Specify any name for the other questions.
5. Specify the keystore password. The `jks` file is created in the Studio Server.

   NOTE You must use the same password and alias that is provided in the `config.sh` file.

6. Run the following jks command in the Studio Server to generate the `.p12` file using the `.jks` file.;

   ```
   keytool -J-Dkeystore.pkcs12.legacy -importkeystore -srckeystore
   <alias>.jks -destkeystore <alias_name>.p12 -srcalias <alias> -
   srcstoretype jks -deststoretype pkcs12
   ```

7. Specify the keystore password. The `.p12` file is created in the Studio Server.
8. Copy the `.p12` files and place in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-home/mmg-studio/conf` directory.

4.6.2 Generate Signed Certificate

To generate the signed certificate, perform the following steps:

1. Log in to the server as a Linux user.

2. Generate the CSR file that describes the certificate requested and needed by the signing authority.
3. Openssl default configuration does not include subject alternative names by default.
4. SANs should be updated in `cert.conf` file. Additional SANs or IPs can be added through properties such as DNS.2, DNS.3, IP.1, and IP.2 in the [alt_names] section.
5. Once the configuration file is placed, generate the CSR file and associated private key by running the following command:

   ```
   openssl req -new -newkey rsa:2048 -nodes -keyout server.key -out server.csr -config cert.conf
   ```

6. Provide the requested entries, and some entries can be left blank.

NOTE

- You can check the CSR contains SANs by running the following command:

  ```
  openssl req -text -noout -verify -in server.csr
  ```

- This step is optional only.

7. Request certificate from the signing authority. Once the certificate is received, convert the `server.cer` into PEM format if required by running the following command:

   ```
   openssl x509 -in server.cer -out server.pem -outform PEM
   ```

NOTE

- You can check the contents of the certificate to make sure that the SANs are included by running the following command:

  ```
  openssl x509 -in server.pem -text
  ```

- This step is optional only.

8. Create `.p12` keystore.

NOTE

- The -name parameter must match the value of the **STUDIO_SERVER_SSL_ALIAS** variable from the path `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin/config.sh`
- To store the password, run the following command:

  ```
  openssl pkcs12 -export -out studio_server.p12 -inkey server.key -in server.pem -name studio_alias
  ```

- The password must match the value of the **STUDIO_SERVER_SSL_PASSWORD** variable from `<COMPLIANCE_STUDIO_INSTALLATION_PATH >/bin/config.sh`
- To check the keystore, run the following command:

  ```
  openssl pkcs12 -export -out studio_server.p12 -inkey server.key -in server.pem -name studio_alias
  ```

- This step is optional only.

9. Copy the `cp studio_server.p12` file and place in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-home/mmg-studio/conf/studio_server.p12` and `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/conf/studio_server.p12` directories.
10. Restart Compliance Studio. To do this, navigate to the

    `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory and run the `./compliance-studio.sh restart`

4.7 Import the certificate to JDK security

> **NOTE** This section is required for both signed and self-signed certificates.

To import `.p12` and `.jks` files, follow these steps:

1. Execute the following commands:

   ```
   keytool -exportcert -keystore <Path of .p12 file >/<filename>.p12 -storetype PKCS12 -alias <alias> -file <Path where studiop.cer file should be created>/studiop.cer
   ```

   For example,

   ```
   keytool -exportcert -keystore /Home/fccstudio/CS_81201_81240_UP/OFS_COMPLIANCE_STUDIO/studio_server.p12 -storetype PKCS12 -alias studio_server -file /scratch/fccstudio/CS_81201_81240_UP/OFS_COMPLIANCE_STUDIO/studiop.cer
   ```

   ```
   keytool -importcert -keystore <JAVA_HOME>/lib/security/cacerts -storepass changeit -alias studio_server -file <Path of studiop.cer file created from about command>/studiop.cer
   ```

   For example,

   ```
   keytool -importcert -keystore /Home/fccstudio/jdk-11.0.18/lib/security/cacerts -storepass changeit -alias studio_server -file /scratch/fccstudio/CS_81201_81240_UP/OFS_COMPLIANCE_STUDIO/studiop.cer
   ```

   > **NOTE** If you need to delete certificate from the JDK then execute the following command:
   >
   > ```
   > keytool -delete -noprompt -alias studio_server  -keystore "<JAVA_HOME>/lib/security/cacerts" -storepass "changeit"
   > ```
   >
   > This could be helpful if you need to re-import a new certificate in the JDK.

4.8 Place the Key Store File for Secure Batch Service

> **NOTE** **For upgrade scenario**: The files can be re-utilized from the previous version of Compliance Studio. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then `.jks` and `.p12` files have to be generated newly by performing the steps mentioned in the section.

Place the `.jks` and `.p12` files generated from the Generate Compliance Studio Server SSL Configuration Mandatory File section and place them in the path (batch service) `<Compliance_Studio_Installation_Path>/batchservice/conf`.

> **NOTE** The `.jks` file is not required if you have signed p12 certificate. To generate signed certificate, see the Generate Signed Certificate section.

4.9 Add the Studio Service (SSL) to PGX Configuration

Adding the Studio Service (SSL) to PGX Trust Store facilitates you to apply redaction on the graph batch service and connect with PGX.

To add the Studio Service to PGX Trust Store, copy the `.p12` file from `<Compliance_Studio Installation_path>/mmg-studio/conf` directory to the `<PGX Server path>/pgx-server/conf` directory.

After generating the `.p12` file and adding the Studio service to the PGX trust store.

4.10 Configure the Extract Transfer and Load (ETL) Process

> **NOTE**
> - **This section is deprecated in the current release and will be removed in the future release.**
> - If you are using a Graph pipeline, skip this section. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

Extract Transfer and Load (ETL) is the procedure of copying data from one or more sources into a destination system that represents the data differently from the source or in a different context. Data movement and graph loading is performed using ETL.

To configure the Data Movement and Graph Load, copy the applicable `FCCM_Studio_SqoopJob.sh` files from the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/ficdb/bin` directory and add in the `<FIC_HOME of OFSAA_Installed_Path>/ficdb/bin` directory.

For more information, see the **Configure ETL** and **Execute ETL** sections in the OFS Compliance Studio Administration and Configuration Guide.

> **NOTE** Before running the sqoop job, ensure that the correct values are the `server-config.properties` file from the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/batchservice/conf` directory.

4.10.1 Loading Graphs

Loading graphs to Compliance Studio can be based on the following scenarios:

4.10.1.1 Loading sample graph without running ETL

NOTE

- **This section is deprecated in the current release and will be removed in the future release.**
- If you are using a Graph pipeline, skip this section. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

To load the sample graph without running ETL, perform the following steps:

1. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/sample-graph` folder.
2. Unzip the contents of the `sample-graph-8.*.zip` file in the same folder.
3. Copy the entire path of the folder `sample-graph`.
4. Open the `sample-graph-8.*.json` file and paste the copied `<sample-graph folder path>` into the placeholder `<SERVER_PATH>` under the parameter `"uris"`.

NOTE Ensure to replace all the placeholders with the copied path of the folder `sample-graph`.

5. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/pgx/pgx-server/conf` directory and modify the `pgx.conf` file for the highlighted parameters:

```
"preload_graphs": [
    {
      "path": "<sample-graph folder path>/sample-graph.json",
      "name": "GlobalGraphIH",
      "publish": false,
      "publish_with_snapshots": true
    }
  ],
  "pgx_realm": {
    "implementation": "com.oracle.ofss.fccm.studio.pgx.FCCMPgxRealm"
  },
  "file_locations": [
    {
      "name": "hdfs_storage",
      "location": "<sample-graph folder path>"
    }
  ]
```

6. Restart the PGX server.

4.10.1.2 Loading the graph generated from ETL

NOTE

- **This section is deprecated in the current release and will be removed in the future release.**
- If you are using a Graph pipeline, skip this section. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

You can load a graph generated from ETL based on the following:

PGX fails until you have a graph generated from ETL on the HDFS. Once the graph is generated, and then as soon as the PGX server pods restart, the graph is pre-loaded to the Compliance Studio.

- Create a backup of `pgx.conf`. The backup can be used when the graph is generated from ETL.
- At the time of deployment, you must delete the following lines from the `pgx.conf` file.

```
"preload_graphs": [
  {
    "path": "##URL_GLOBAL_GRAPH_CONFIG_JSON##",
    "name": "##PGX_GLOBAL_GRAPH_NAME##",
    "publish": false,
    "publish_with_snapshots": true
  }
],
```

  - The following lines must be delete multiple times.

```
,
        {
          "preloaded_graph": "##PGX_GLOBAL_GRAPH_NAME##",
          "grant": "manage"
        }
```

- Proceed with the Compliance Studio deployment.
- Once the graph is generated, perform the following:
  - Replace the existing `pgx.conf` file with the backed up pgx.conf file
- Restart Compliance Studio.

4.10.1.3 Loading the graph generated from the Graph Pipeline

> **NOTE**
> - **This section is deprecated in the current release and will be removed in the future release.**
> - If you are using a Graph pipeline, skip this section. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

You can load a graph generated from the Graph pipeline based on the following:

1. To configure the PGX service, see the Configure the PGX Service section.
2. In the `config.sh` file set the `LOAD_GRAPH_FROM_HDFS` parameter to load the graph:

   If **True**, both the graph pipeline and legacy ETL will be loaded.

   If **False**, the graph pipeline only will be loaded.
3. Start/restart the PGX service.
4. To refresh the graph, see the **Creating a Population Schedule** section in the OFS Compliance Studio User Guide and perform the steps from **10** to **12.**

   The graph will be loaded.
5. To verify the loaded graph details, kindly create a new notebook and query the graph using pgql.

4.11 Apply Fine-Grained access control and Redaction Changes for Compliance Studio

> **NOTE**
> - **This section is deprecated in the current release and will be removed in the future release.**
> - This is applicable for the graph which is generated by Legacy ETL, i.e., with BigData cluster.

After generating the key store file and adding the batch service to the PGX trust store, you must configure the user mapping for the changes made in the database. For more information about configuring user mapping, see the OFS Compliance Studio Administration and Configuration Guide.

4.12 Configure the config.sh File

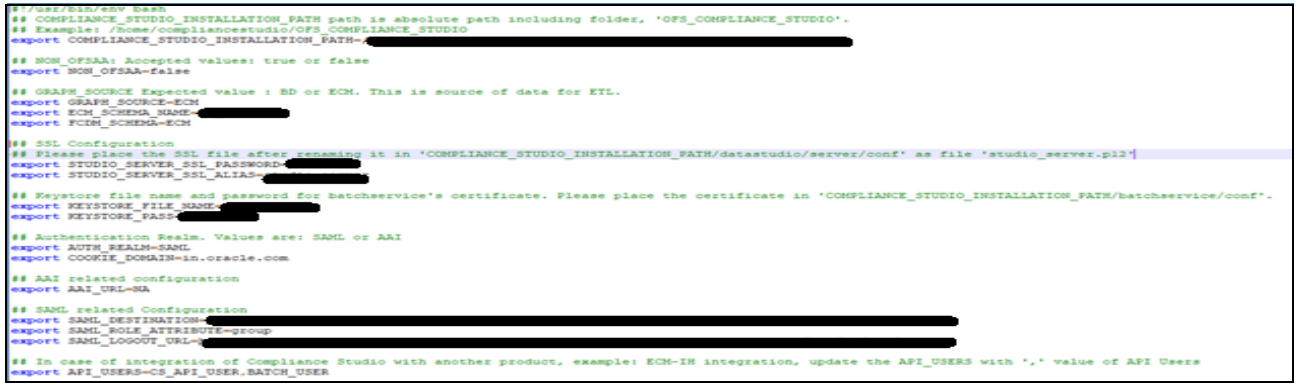
> **NOTE**
> **For upgrade scenario**: The value of highlighted parameters from the `config.sh` file has to be re-utilized from the previous version of Compliance Studio. If you upgrade from 8.1.2.0.* to CS 8.1.2.5.0, the additional parameters can be configured as mentioned in this section.

To configure the `config.sh` file for installing Compliance Studio, follow these steps:

1. Login to the server as a non-root user.
2. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory.
3. Configure the applicable `config.sh` attributes are shown in the following table.

A sample `config.sh` file is shown:

Figure 5: Sample Config.sh File



NOTE

- You must manually set the parameter value in the `config.sh` file. If a value is not applicable, enter NA and ensure that the value is not entered as **NULL**.
- If the parameter STUDIO_DB_SERVICE_NAME has been filled, the parameter STUDIO_DB_SID should be left **blank**, and vice versa.
- If the parameter ATOMIC_DB_SERVICE_NAME has been filled, the parameter ATOMIC_DB_SID should be left **blank**, and vice versa.

Table 15 lists configuration parameters of the `config.sh` file.

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
COMPLIANCE_STUDIO_INSTALLATION_PATH	Indicates the path where the Compliance Studio installer file is extracted. For example, `/scratch/testuser/OFS_COMPLIANCE_STUDIO`.	Yes	Yes. Provide the path where the new installer is extracted. For example, `/scratch/testuser/OFS_COMPLIANCE_STUDIO`.	Yes	Yes
NON_OFSAA	To install Compliance Studio with OFSAA, enter "**false**". To install Compliance Studio without OFSAA, enter "**true**".	Yes Enter the value as **false**.	Yes Enter the value as **false**.	Yes Enter the value as **true**.	Yes Enter the value as **true**.

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
GRAPH_SOURCE					
GRAPH_SOURCE	Indicates the source database for Compliance Studio. The available options are **ECM** and **BD**. **NOTE:** • If Legacy Graph (ETL connector job using Hadoop) is not required, then set the value as **NA**. • Compliance Studio can use either the BD or ECM schema as the source of FCDM data for the graph. • Ensure to enter the value as ECM whenever ECM integration is required with Investigation Hub. • Here, the ECM schema is used as the source of the FCDM data to load the case information into the graph.	Yes The value of this parameter should be provided either **BD** or **ECM**.	Yes The value of this parameter should be provided either **BD** or **ECM**. Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter according to the Graph source, ECM/ BD.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
ECM_SCHEMA_NAME	Indicates the name of the ECM Atomic Schema. **NOTE:** If Legacy Graph (ETL connector job using Hadoop) is not required, then set the value as **NA**.	Yes The value should be name of the ECM Atomic Schema.	Yes The value should be name of the ECM Atomic Schema. Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
FCDM_SCHEMA	This indicates the datasource for the Production workspace. The available options are **ECM** and **BD**.	Yes The value of this parameter should be provided either **BD** or **ECM**.	Yes The value of this parameter should be provided either **BD** or **ECM**. Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
SSL Configuration					
STUDIO_SERVER_SSL_SECRET	Indicates the password for Studio Server P12 that is required for HTTPS configuration.	Yes	Yes. This file has to be generated newly. To create a file, see the Generate Compliance Studio Server SSL Configuration Mandatory File section.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
STUDIO_SERVER_SSL_ALIAS	Indicates the alias name for P12 for the Studio Server.	Yes	Yes. Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly .	Yes	Yes
Authentication Realm					
AUTH_REALM	Realm indicates the functional grouping of database schemas and roles that must be secured for an application. Realms protect data from access through system privileges; realms do not give its owner or participant's additional privileges. Compliance Studio uses realm-based authorization and authentication for its users. For more information, see the Realm Based Authorization section in the OFS Compliance Studio Administration and Configuration Guide. The Compliance Studio application can be accessed using the following realms: **FCCMRealm** Value=AAI **FCCSamlRealm** Value=SAML	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
COOKIE_DOMAIN	The domain of the server where Compliance Studio is installed. Example: in.oracle.com	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
AAI related configuration					
AAI_URL	The Application URL of ECM/BD application. URL: http://<Server Hostname>:<Applic ation URL PORT>/ <Context Path> For example, http:// testserver.in.oracle. com:4000/ BDTEST **NOTE:** This parameter is applicable only if AUTH_REALM is **AAI**.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
SAML The SAML-related parameters are applicable only if SAMLRealm is used in the Realm parameter.	In the case of SAML Realm, the certificate from IDP (**key.cert** file) is required. The certificate that is obtained from the IDP must be renamed to key.cert and placed in the <COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-home/mmg-studio/conf directory. This certificate is used to identify the trust of the SAML response from the Identity Provider. Specify the Role Attribute name from IDP, in which the User Roles are present in the SAML response.				
SAML_DESTINATION	Indicates the SAML IDP URL that the Identity Provider provides after creating the SAML Application. **NOTE:** This parameter is applicable only if AUTH_REALM is **SAML**.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio config.sh file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
SAML_ROLE_ATTRIBUTE	Indicates the SAML client identifier provided by the SAML Administrator for the Role and Attributes information while creating the SAML application for Compliance Studio. **NOTE:** This parameter is applicable only if AUTH_REALM is **SAML**.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
SAML_LOGOUT_URL	Indicates the SAML client identifier provided by the SAML Administrator for the Logout URL information while creating the SAML application for Compliance Studio. **NOTE:** This parameter is applicable only if AUTH_REALM is **SAML**.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
Integrate with other products					
API_USERS	Retain the default settings.	Yes	Yes Retain Default Settings.	Yes	Yes
VALID_ROLES	Retain the default settings.	Yes	Yes Retain Default Settings.	Yes	Yes
BATCH_ROLE	Retain the default settings.	Yes	Yes Retain Default Settings.	Yes	Yes
MMG Service Configurations					
SESSION_TOKEN_CREDENTIALS	Retain the default settings.	Yes	Yes Retain Default Settings.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
FCC_API_USER	Retain the default settings.	Yes	Yes Retain Default Settings.	Yes	Yes
SSO_TOKEN	This is the API token for FCC_API_USER. See the Generate API token for CS API User section for token value.	Yes	Yes See the Generate API token for CS API User section for token value.	Yes	Yes
MMG_DATASOURCE_MAX_POOL_SIZE	Maximum connection pool size allowed for Config Data Source. 50	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
MMG_DATASOURCE_IDLE_TIMEOUT	Idle timeout for Config Data Source in a millisecond. 30000	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
MMG_DATASOURCE_CONN_TIMEOUT	Connection timeout for Config Data Source in milliseconds. 30000	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
EXT_DATASOURCE_MAX_POOL_SIZE	Maximum connection pool size allowed for Meta/Data Schemas. 50	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
EXT_DATASOURCE_IDLE_TIMEOUT	Idle timeout for Meta/Data Schemas in milliseconds. 30000	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
EXT_DATASOURCE_CONN_TIMEOUT	Connection timeout for Meta/ Data Schemas in milliseconds. 30000	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
SERVER_COOKIE_TIMEOUT	Connection timeout for server cookie in milliseconds. 86400	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
DB Details for Studio Schema You must be logged in as SYSDBA to perform these configurations.					
STUDIO_DB_HOSTNAME	Indicates the hostname of the database where the Compliance Studio schema is created.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes
STUDIO_DB_PORT	Indicates the port number where the Compliance Studio schema is created.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
STUDIO_DB_SERVICE_NAME	Indicates the service name of the database where the Studio schema is created.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes
STUDIO_DB_SID	Indicates the SID of the database where the Studio schema is created. **NOTE:** Set this field as blank if there is no SID for Database.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes
STUDIO_DB_USERNAME	Indicates the username of the Compliance Studio Schema (newly created Oracle Schema).	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes
DB Details of Atomic Schema					
ATOMIC_DB_HOSTNAME	The hostname of the database where Atomic schema is present (BD/ECM config).	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes
ATOMIC_DB_PORT	Port number of database where Atomic schema is present.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes
ATOMIC_DB_SERVICE_NAME	The service name of the database where Atomic schema is present.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
ATOMIC_DB_SID	Service id of database where Atomic schema is present. **NOTE:** Set this field as blank if there is no SID for Database.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes
ATOMIC_DB_USERNAME	Username of the Atomic schema	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes
Studio DB Wallet Details For information on creating a wallet, see Setup Password Stores with Oracle Wallet.					
WALLET_LOCATION	Indicates the Compliance Studio wallet location.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes
TNS_ADMIN_PATH	Indicates the path of the tnsnames.ora file where an entry for the STUDIO_ALIAS_NAME is present.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes
STUDIO_ALIAS_NAME	Indicates the Studio alias name. **NOTE:** Enter the alias name that was created during wallet creation.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
ATOMIC_ALIAS_NAME	Indicates alias name of FCDM source atomic schema given in wallet. **NOTE:** If Legacy Graph (ETL connector job using Hadoop) is not required, then set the value as **NA**.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio.	Yes	Yes
Big Data Setup Details Contact your System Administrator to obtain the required details for these parameters. **NOTE:** This parameter is deprecated in the current release and will be removed in the future release.					
STUDIO_HADOOP_CREDENTIAL_ALIAS	Indicated the alias password saved in Hadoop. For example, studio.password.alias **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
STUDIO_HADOOP_CREDENTIAL_PATH	Indicates the credentials path. For example, <Compliance Studio Installed Path> oracle.password.jceks **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
HADOOP_CREDENTIAL_PROVIDER_PATH	Indicates the path where the Hadoop credential is stored. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
HADOOP_PASSWORD_ALIAS	Indicates the Hadoop alias given when creating the Hadoop credentials. For information on creating a credential keystore, see Create the Credential Keystore. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
Hive_Host_Name	Indicates the Hive hostname. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
Hive_Port_number	Indicates the Hive port number. Contact your Studio Administrator to obtain the port number. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
HIVE_PRINCIPAL	Indicates the Hive Principal. Contact your Studio Administrator to obtain the HIVE_PRINCIPAL value. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
HIVE_SCHEMA	Indicates to create a schema in HIVE. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
Krb_Host_FQDN_Name	Indicates the Kerberos host FQDN name. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
Krb_Realm_Name	Indicates the Kerberos realm name. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
Krb_Service_Name	Indicates the Kerberos service name. Example: Hive **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
server_kerberos_keytab_file	Indicates the Kerberos keytab file. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
server_kerberos_principal	Indicates the Kerberos Principal. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
server_kerberos_krb5_conf_file	Indicates the krb5.conf file name. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
SQOOP_HOSTMACHINE_USER_NAME	Indicates the username of the Host machine where sqoop will run. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
SQOOP_PARAMFILE_PATH	4. Create a file with the name `sqoop.properties` in the Big Data server and add the following entry: oracle.jdbc.mapDateToTimestamp=false 5. Enter the location of the `sqoop.properties` file as the value for this parameter. Example: /scratch/ofsaa/ **NOTE:** Ensure that the location name ends with a'/'. **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
SQOOP_PARTITION_COL	Indicates the column in which the HIVE table is partitioned. The value must be SNAPSHOT_DT. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
SQOOP_TRG_HOSTNAME	Indicates the hostname of the Big Data server where SQOOP will run. Example: <HostName> **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
SQOOP_WORKDIR_HDFS	Indicates the Sqoop working directory in HDFS. Example: /user/ ofsaa **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**.
ETL **NOTE:** This parameter is deprecated in the current release and will be removed in the future release.					

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
HDFS_GRAPH_FILES_PATH	Indicates the file path in the HDFS where the graph.json is formed. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
GRAPH_FILES_PATH	Indicates the directory in the Big Data server for graph files. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
GRAPH_NAME	Indicates the name you want to assign to the global graph at the end of ETL. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
ETL_PROCESSING_RANGE	Indicates the duration for which the data would be moved from Oracle to Hive. For example: If the ETL_PROCESSING_ RANGE = 2Y, 3M, 10D, that is, 2 years, 3 months, and 10 days, and the current date is 20200814, then the data movement occurs for the range 20180504 to 20200814. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
OLD_GRAPH_SESSION_DURATION	Indicates that the session older than this specified duration will be removed from the PGX server. If unsure, you can set this value for a week (7D). **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
REMOVE_TRNXS_EDGE_AFTER_DURATION	Indicates the date range for which transaction edges will be maintained in the graph. For example, 6Y, 3M, 10D, which means 6 years, 3 months, and 15 days. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
CONNECTOR_CHANGESET_SIZE	Indicates the number of nodes or edges you want to process during an update of the graph. If unsure, you can set it to 10000. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
CB_CONFIGURED	Indicates the setting of the graph edges. If the corresponding edges of the graph are needed, set the value to true. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**	Enter true or false	Enter true or false Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Enter true or false	Enter true or false
Logstash					
LOGSTASH_HOME	Logstash home Example: "/`<COMPLIANCE_STUDIO_INSTALLATION_PATH>`/Logstash/logstash- 7.16.3" **NOTE:** See the section Configure Logstash for more details. If Graph Pipeline and Entity Resolution functionalities are not required, then set the value as **NA**.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
Graph Service **NOTE:** The following parameters for graph service are mandatory for successful Compliance Studio installation, and the parameters cannot be set as blank or NA. If you do not want to use graph pipeline functionality, studio schema details should be provided for these parameters.					
GRAPH_DB_SERVER_NAME	Indicates the Graph Database server name.	Yes	Yes	Yes	Yes
GRAPH_DB_PORT	Indicates the Graph Database server port.	Yes	Yes	Yes	Yes
GRAPH_DB_SERVICE_NAME	Indicates the Graph Database service name.	Yes	Yes	Yes	Yes
GRAPH_KEYSTORE_PASSWORD	Indicates the password of the keystore file, which stores the password of the graph schema. **NOTE:** If Graph Pipeline functionality is not required, then set the value as **NA**.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
GRAPH_SCHEMA_DB_SCHEMA_NAME	Indicates the Database schema name of the graph schema.	Yes	Yes	Yes	Yes
GRAPH_SCHEMA_WALLET_ALIAS	Indicates the wallet alias of the graph schema.	Yes	Yes	Yes	Yes
GRAPH_SCHEMA_WALLET_LOCATION	Indicates the wallet location of the graph schema.	Yes	Yes	Yes	Yes
GRAPH_SCHEMA_TNS_ADMIN_PATH	Indicates the TNS admin path of the graph schema.	Yes	Yes	Yes	Yes
PGX server configuration i.e., Interpreter, data memory limits					

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
NUM_CACHED_RESULTSET	Indicates the cached result set. For example, 0	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
RESULTSET_EXPIRATION_TIME_SECS	Indicates the Result set expiration time. For example, 3600.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
MAX_TOTAL_SHARED_DATA_MEMORY_SIZE	The absolute memory limit of shared data (includes published graphs and pinned non-referenced graphs). For example: 20G	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
MAX_TOTAL_PRIVATE_DATA_MEMORY_SIZE	The memory limit of private data (includes non-published graphs and PGQL results) relative to the total PGX engine memory limit. For example, 8G	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
MAX_PER_SESSION_DATA_MEMORY_SIZE	Absolute memory limit for any one session of the PGX engine. For example: 700M	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
MAX_DATA_MEMORY_SIZE_DSUSRGRP	Absolute memory limit for any user of the PGX engine whose role is DSUSRGRP. For example: 2G	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
MAX_DATA_MEMORY_SIZE_DSBATCH	Absolute memory limit for any user of the PGX engine whose role is DSBATCH. For example: 10G	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
MAX_DATA_MEMORY_SIZE_DSINTER	Absolute memory limit for any user of the PGX engine whose role is DSINTER. For example: 5G	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
MAX_DATA_MEMORY_SIZE_DSAPPROVER	Absolute memory limit for any user of the PGX engine whose role is DSAPPROVER. For example: 5G	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
MAX_DATA_MEMORY_SIZE_DSUSER	Absolute memory limit for any user of the PGX engine whose role is DSUSER. For example, 5G	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
Service URLs					
PGX_SERVER_URL	Indicates the URL of the PGX server. **NOTE:** Ensure to provide the correct hostname for the URL of the PGX service. If Legacy Graph (ETL connector job using Hadoop) and Graph Pipeline functionalities are not required, then set the value as **NA**.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	No	No
HTTPS_PROXY_HOST	Indicates the proxy host that is used. For example, test-proxyserver.com	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
HTTPS_PROXY_PORT	Indicates the proxy port that is used. For example, 80	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
HTTP_PROXY_USERNAME	Indicates the proxy username used, if there is any.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
HTTP_PROXY_PASSWORD	Indicates the proxy password used if there is any.	Yes	Yes Enter the value of this parameter from the previous version of the Compliance Studio `config.sh` file. If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.	Yes	Yes
R interpreter settings					

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
RS_CONF_PATH	Absolute path to Rserve.conf file for running Rserve. Example: /scratch/ users/mmg-studio/ conf/Rserve.conf **NOTE**: If required, then configure the parameter.	No	No	No	No
RS_KEYSTORE	Absolute path for the Keystore file made for Rserve.conf. Example: /scratch/ users/mmg-studio/ conf/ rinterpreterkeystore **NOTE**: If required, then configure the parameter.	No	No	No	No
RS_KS_SECRET	Keypass for rinterpreterkeystore Example: Change it. If the target AAI is https, then the certificate of the target machine needs to be imported to the DS Java keystore. **NOTE**: If required, then configure the parameter.	No	No	No	No
Additional Environment variables					
LD_LIBRARY_PATH	Oracle Instant client path For example: /opt/ oracle/ instantclient_19_8/ :$LD_LIBRARY_PAT H				
Cluster Type					

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
CLUSTER_TYPE	Indicates the type of cluster.	Yes The value of the parameter is **OS**.	Yes The value of the parameter is **OS**.	Yes Enter the value as **NA**.	Yes Enter the value as **NA**
OpenSearch Cluster Details					
OPEN_SEARCH_HOSTNAME	Indicates the hostname of the server where the OpenSearch service is installed. **NOTE:** If Graph Pipeline and Entity Resolution functionalities are not required, then set the value as **NA**.	Yes	Yes The value of this parameter has to be created newly based on the OpenSearch configuration.	Yes	Yes
OPEN_SEARCH_PORT	Indicates the port number where the OpenSearch service is installed. **NOTE:** If Graph Pipeline and Entity Resolution functionalities are not required, then set the value as **NA**.	Yes	Yes The value of this parameter has to be created newly based on the OpenSearch configuration.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
OPEN_SEARCH_HADOOP_CREDENTIAL_PATH	Indicates the open search hadoop credential path. For information about path, see the Create Credential Keystore for OpenSearch section. **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.** It is applicable only for legacy ETL. If Graph Pipeline and Entity Resolution functionalities are not required, then set the value as **NA**.	Yes	Yes The value of this parameter has to be created newly based on the OpenSearch configuration.	Yes	Yes
OPEN_SEARCH_USERNAME	OpenSearch Username (Not Applicable, if https enabled is false and authentication is not supported). **NOTE:** If Graph Pipeline and Entity Resolution functionalities are not required, then set the value as **NA**.	Yes	Yes The value of this parameter has to be created newly based on the OpenSearch configuration.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
OPEN_SEARCH_ENCRYPTED_PASSWORD	Encrypted password (Not Applicable, if https enabled is false and authentication is not supported). **NOTE:** To generate an encrypted password, see the Appendix F - Generate an Encrypted Password for OPenSearch section. If Graph Pipeline and Entity Resolution functionalities are not required, then set the value as **NA**.	Yes	Yes The value of this parameter has to be created newly based on the OpenSearch configuration.	Yes	Yes
OPEN_SEARCH_HADOOP_PASSWORD_ALIAS	Indicates the password alias for OpenSearch (Not applicable if OS OPEN_SEARCH_HTTPS_ENABLED is false). **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.** If Graph Pipeline and Entity Resolution functionalities are not required, then set the value as **NA**.	Yes	Yes The value of this parameter has to be created newly based on the OpenSearch configuration.	Yes	Yes
OPEN_SEARCH_HTTPS_ENABLED	True (If OS is https enabled, else false). **NOTE:** If Graph Pipeline and Entity Resolution functionalities are not required, then set the value as **false**.	Yes	Yes The value of this parameter has to be created newly based on the OpenSearch configuration.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
OPEN_SEARCH_TRUSTSTORE_FILE_NAME	The filename of the OpenSearch keystore that contains the certificates of OS host to trust (Not Applicable, if https enabled is false). **NOTE:** If Graph Pipeline and Entity Resolution functionalities are not required, then set the value as **NA**.	Yes	Yes The value of this parameter has to be created newly based on the OpenSearch configuration.	Yes	Yes
OPEN_SEARCH_TRUSTSTORE_SECRET	The password of the OpenSearch keystore file. (Not Applicable, if https enabled is false). **NOTE:** If Graph Pipeline and Entity Resolution functionalities are not required, then set the value as **NA**.	Yes	Yes The value of this parameter has to be created newly based on the OpenSearch configuration.	Yes	Yes
OPEN_SEARCH_KEYSTORE_HADOOP_CREDENTIAL_ALIAS	Indicates the password alias for OpenSearch (Not applicable if OS OPEN_SEARCH_HTTPS_ENABLED is false). **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.** It is applicable only for legacy ETL. If Graph Pipeline and Entity Resolution functionalities are not required, then set the value as **NA**.	Yes	Yes The value of this parameter has to be created newly based on the OpenSearch configuration.	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
All Services	Set the value of the parameter, DEPLOY_ALL_SERVIC E, as : **true** for starting all services **false** for starting selected services. Example: Compliance Studio independent of OFSAA: set "false" for service(s): entity resolution, matching service, and load-to-open Compliance Studio lite: set "false" for service(s): fcc-pgql, fcc-pgx-algorithm, fcc-pgx-java and pgx-server				
DEPLOY_ALL_SERVIC E	True: Indicates that all services are deployed.	Yes	Yes	Yes	Yes
Services					
METASERVICE_ENAB LED	True: Indicates that the service is enabled.	Yes	Yes	Yes	Yes
BATCHSERVICE_ENA BLED	True: Indicates that the service is enabled.	Yes	Yes	Yes	Yes
GRAPH_SERVICE_EN ABLED	True: Indicates that the service is enabled.	Yes	Yes	Yes	Yes
FCC_PYTHON_ENAB LED	True: Indicates that the service is enabled.	Yes	Yes	Yes	Yes
JDBC_ENABLED	True: Indicates that the service is enabled.	Yes	Yes	Yes	Yes
SPARK_ENABLED	True: Indicates that the service is enabled.	Yes	Yes	Yes	Yes
PGX_ENABLED	True: Indicates that the service is enabled.	Yes	Yes	Yes	Yes
ENTITY_RESOLUTIO N_ENABLED	True: Indicates that the service is enabled.	Yes	Yes	Yes	Yes

Table 15: config.sh file

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing without OFSAA (Mandatory)	Upgrading without OFSAA (Mandatory)
MATCHING_SERVICE_ENABLED	True: Indicates that the service is enabled.	Yes	Yes	Yes	Yes
LOAD_TO_OPEN_SEARCH_ENABLED	True: Indicates that the service is enabled.	Yes	Yes	Yes	Yes
MMG_SERVICE_ENABLED	True: Indicates that the service is enabled.	Yes	Yes	Yes	Yes

4.13 Configure the resources.xml for Multiple ER Schemas

NOTE

- The **ER_SCHEMA_ID** should be the same value as the datasource name attached to the ER workspace.
- For ease of execution, it is recommended to have the same **ER_Data_Schema_Alias_Name** as the **ER_Schema_ID**.
- **ER_Data_Schema_Alias_Name** and **ER_Schema_ID** are case sensitive, so it is recommended to use the same case for both of them.

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/ficdb/conf`

NOTE If the user wants to add additional ER schemas post-installation, the path will change to:
`<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/ficdb/conf`
The remaining steps will remain the same.

2. Open `resources.xml` file.
3. Provide the **id** as `ER_Schema_ID` and **ER_DATA_SCHEMA_ALIAS_NAME** as `ER_Schema_Alias`.

   The sample resource tag will look like the following, and users can change the values as applicable:

```
<Resource
            id="##ER_DATA_SCHEMA_ALIAS_NAME##"
            name="jdbc/erdataschema"
            auth="Container"
```

```
            type="javax.sql.DataSource"
            driverClassName="oracle.jdbc.OracleDriver"
            url="jdbc:oracle:thin:@##ER_DATA_SCHEMA_ALIAS_NAME##"
            connectionProperties="oracle.net.wallet_location
=##STUDIO_WALLET_LOCATION##;
oracle.net.tns_admin=##STUDIO_TNS_ADMIN_PATH##;"
            maxTotal="20"
            maxIdle="0"
            maxWaitMillis="-1" >
    </Resource>
```

Example `resource.xml`tag with single ER Schema:

```
<Resource
            id="ER1"
            name="jdbc/erdataschema"
            auth="Container"
            type="javax.sql.DataSource"
            driverClassName="oracle.jdbc.OracleDriver"
            url="jdbc:oracle:thin:@ER1"
            connectionProperties="oracle.net.wallet_location
=##STUDIO_WALLET_LOCATION##;
oracle.net.tns_admin=##STUDIO_TNS_ADMIN_PATH##;"
            maxTotal="20"
            maxIdle="0"
            maxWaitMillis="-1" >
    </Resource>
```

4. The sample can be repeated for multiple ER Schemas with a unique id and ER_Schema_Alias.

   Example **`resource.xml`** tag with multiple ER Schemas:

```
<Resource
            id="ER1"
            name="jdbc/erdataschema"
            auth="Container"
            type="javax.sql.DataSource"
            driverClassName="oracle.jdbc.OracleDriver"
            url="jdbc:oracle:thin:@ER1"
            connectionProperties="oracle.net.wallet_location
=##STUDIO_WALLET_LOCATION##;
```

```
oracle.net.tns_admin=##STUDIO_TNS_ADMIN_PATH##;"
            maxTotal="20"
            maxIdle="0"
            maxWaitMillis="-1" >
      </Resource>
<Resource
            id="ER2"
            name="jdbc/erdataschema"
     auth="Container"
            type="javax.sql.DataSource"
            driverClassName="oracle.jdbc.OracleDriver"
            url="jdbc:oracle:thin:@ER2"
            connectionProperties="oracle.net.wallet_location
=##STUDIO_WALLET_LOCATION##;
oracle.net.tns_admin=##STUDIO_TNS_ADMIN_PATH##;"
            maxTotal="20"
            maxIdle="0"
            maxWaitMillis="-1" >
    </Resource>
```

NOTE Make sure that the following parameters are updated with the values:

```
maxTotal="20"
maxIdle="0"
```

4.14 Configure the resources.xml for Graph Schema

To update the `resources.xml` files, perform the step:

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/ficdb/conf` directory and add details of the wallet alias as shown below:

NOTE For OOB Graph execution, add wallet details of BD Schema and Graph Schema.

```
<?xml version="1.0" encoding="UTF-8"?>
<Resources>
    <Resource
        id="complianceStudioSchemaWalletAlias"
        name="jdbc/studioschema"
        auth="Container"
```

```
        type="javax.sql.DataSource"
        driverClassName="oracle.jdbc.OracleDriver"
        url="jdbc:oracle:thin:@complianceStudioSchemaWalletAlias "
        connectionProperties="oracle.net.wallet_location=/scratch/
fccstudio/OFS_COMPLIANCE_STUDIO/wallet;oracle.net.tns_admin=/scratch/
fccstudio/OFS_COMPLIANCE_STUDIO/wallet;"
        maxTotal="5"
        maxIdle="0"
        maxWaitMillis="-1">
    </Resource>
    <Resource
        id="graphSchemaWalletAlias"
        name="jdbc/erdataschema"
        auth="Container"
        type="javax.sql.DataSource"
        driverClassName="oracle.jdbc.OracleDriver"
        url="jdbc:oracle:thin:@graphSchemaWalletAlias"
        connectionProperties="oracle.net.wallet_location=/scratch/
fccstudio/OFS_COMPLIANCE_STUDIO/wallet;oracle.net.tns_admin=/scratch/
fccstudio/OFS_COMPLIANCE_STUDIO/wallet;"
        maxTotal="5"
        maxIdle="0"
        maxWaitMillis="-1">
    </Resource>
    <Resource
        id="graphSchemaDatasourceName"
        name="jdbc/erdataschema"
        auth="Container"
        type="javax.sql.DataSource"
        driverClassName="oracle.jdbc.OracleDriver"
        url="jdbc:oracle:thin:@graphSchemaWalletAlias"
        connectionProperties="oracle.net.wallet_location=/scratch/
fccstudio/OFS_COMPLIANCE_STUDIO/wallet;oracle.net.tns_admin=/scratch/
fccstudio/OFS_COMPLIANCE_STUDIO/wallet;"
        maxTotal="5"
        maxIdle="0"
        maxWaitMillis="-1">
    </Resource>
```

```
</Resources>
```

4.15 Run the Compliance Studio Installer

This section provides the install, reinstall, start, restart and stop of the services.

Topics:

The Compliance Studio application is installed with or without OFSAA, depending on the configuration provided in the `config.sh` file. The Compliance Studio application and all the interpreters are started.

After completing the Compliance Studio installation, the script displays a URL that can be used to access the Compliance Studio Application.

4.15.1 Installing/Upgrading for the first time

For first-time installation, you can pass argument '-i' or '--install'.

To run the Compliance Studio installer, follow these steps:

1. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory.
2. Run the following command with a Linux user where Compliance Studio is installed:

   ```
   ./compliance-studio.sh –i
   ```

   Or

   ```
   ./compliance-studio.sh --install
   ```

   This will copy the whole compliance studio into the folder 'deployed' and then replaces the placeholders. Now, you can start Compliance Studio.

> **NOTE**
> - Run these commands only from `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin.`
> - It should not be run from `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/bin.`
> - Upon executing `./compliance-studio.sh –i` command. A deployed folder is created that copies all the folders. And replaces placeholders inside the deployed folder.

Congratulations! Your installation is complete.

> **NOTE** For any help on installation commands, Run
>
> `./compliance-studio.sh --help`

4.15.2 Starting Compliance Studio

To start the application, you can run pass argument '-s' or'--start'. Example:

```
./compliance-studio.sh --start
```

This will start the application and, on successful installation, will make the sensitive details blank in `config.sh`.

> **NOTE** If any of the services are not started/running and failed due to lock, perform the following:
>
> 1. Log in to Studio schema.
> 2. Run the following command to Truncate tables:
>
>    ```
>    TRUNCATE TABLE DATABASECHANGELOGLOCK;
>    TRUNCATE TABLE DATABASECHANGELOGLOCK_MMG;
>    ```
> 3. Log in to BD/ECM schema.
> 4. Run the following command to Truncate tables:
>
>    ```
>    TRUNCATE TABLE DATABASECHANGELOGLOCK;
>    ```
> 5. Start the Compliance Studio.

4.15.3 Stopping Compliance Studio

To stop the application, you can run pass argument '-k' or '--stop'. Example:

```
./compliance-studio.sh --stop
```

4.15.4 Restarting Compliance Studio

To restart the application, you can run pass argument '-r' or '--restart'. Example:

```
./compliance-studio.sh --restart
```

4.15.5 Reinstalling Compliance Studio

In case if you need to reinstall compliance Studio due to the wrong configuration or need to update configuration details. Then:

- Stop the Compliance Studio
- Update the `config.sh` file. Do not forget to reconfigure the sensitive details which were removed earlier.

To restart the application, you can run pass argument '-R' or '--reinstall'. Example:

```
./compliance-studio.sh --reinstall
```

Once reinstallation is done, you can start the application.

4.16 Generate the Graph-keystore.p12 File

> **NOTE** Before creating the `graph-keystore.p12` file, ensure that the graph service is up and running.

To generate the `graph-keystore.p12` file, follow these steps;

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-load-to-graph/graph-service/utility/bin` directory.
2. Execute the following command:

   ```
   ./CreatePasswordlessKeystore.sh
   ```

3. It will request for the following values:
   a. Wallet Alias for Graph Schema
   b. Keystore Alias

> **NOTE** For Keystore Alias, use the same value that is provided in the wallet alias of the graph schema.

4. The `graph-keystore.p12` file is generated and available in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-load-to-graph/graph-service/conf` directory.
5. Copy the `graph-keystore.p12` file and place in the `<PGX_HOME>/pgx/pgx-server/conf` directory.

> **NOTE**
> - If you do not have any graph schema then create an empty file with a name "`graph-keystore.p12`" and place it in the `<PGX_HOME>/pgx/pgx-server/deployed/conf` directory.
> - The path where the `pgx-server-<version>.zip` file is unzipped is referred to as `<PGX_HOME>`.
> - If you are updating credentials then copy the updated `graph-keystore.p12` file and place in the `<PGX_HOME>/pgx/pgx-server/conf` directory.

4.17 Configure the PGX Service

> **NOTE** PGX service can be configured on the same server where Compliance Studio is installed or on a different server.

To install PGX service, follow these steps:

> **NOTE** If you are using a Graph pipeline, skip **steps 4, 5,** and **6**. This is applicable only in the case of legacy Graph ETL, which requires a Big Data cluster.

1. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/pgx/pgx-server/` directory.

2. Perform the following:
   - If PGX service is to be installed on the same server where Compliance Studio is installed, extract the `pgx-server-<version>.zip` file.
   - If PGX service is to be installed on a different server, follow these steps:
     — Copy the `pgx-server-<version>.zip` file to the PGX server.
     — Extract the `pgx-server-<version>.zip` file.

> **NOTE** The path where the `pgx-server-<version>.zip` file is unzipped is referred to as `<PGX_HOME>`.

3. Navigate to the `<PGX_HOME>/pgx-server/conf` directory.

> **NOTE** Configure the following properties if applicable:
>
> In the `server.conf` file, configure the following properties:
> - **enable_tls: false,**
> - enable_client_authentication: false
> - The property value is true by default, which means that the SSL certificate is enabled and recommended. Change to false only if you do not have the SSL certificate enabled.

4. Replace the following Kerberos Files in the `<PGX_HOME>/pgx-server/conf/kerberos` directory:

   `krb5.conf`

   keytab file name as mentioned in the `config.sh file.`

> **NOTE** **This step is deprecated in the current release and will be removed in the future release.**
>
> **For upgrade scenario**: Use files from the previous version of CS.

5. Replace the following Hadoop configuration files in the `<PGX_HOME>/pgx-server/conf/hadoop_cluster` directory:
   - `core-site.xml`
   - `hadoop-env.sh`
   - `hdfs-site.xml`
   - `log4j.properties`
   - `ssl-client.xml`
   - `topology.map`
   - `topology.py`
   - `hive-site.xml`
   - `yarn-site.xml`
   - `redaction-rules.json`

- `hive-env.sh`
- `mapred-site.xml`

For additional jars, see Appendix C – Additional Jars – PGX section. Contact your administrator to get the files.

> **NOTE** **This step is deprecated in the current release and will be removed in the future release.**
>
> **For upgrade scenario**: Use files from the previous version of CS.

6. Copy all the obtained jars into `<PGX_HOME>/pgx-server/conf/hdfs_libs` directory.

> **NOTE** **This step is deprecated in the current release and will be removed in the future release.**
>
> **For upgrade scenario**: Use files from the previous version of CS.

7. Navigate to the `<PGX_HOME>/pgx-server/bin` directory and configure the `config.sh` file as described in the Table 16 .

Table 16: Config.sh file for PGX

Interaction Variable Name	Significance
PGX Server Memory Configuration	
`PGX_SERVER_OFF_HEAP_MB`	Indicates the maximum off-heap memory size in megabytes (mainly used for storing graphs except for their string properties) that PGX tries to respect. Recommended Value: 42% of the PGX server memory limit size above.
`PGX_SERVER_ON_HEAP_MB`	Indicates the maximum and minimum heap memory size (mainly used for storing graphs' string properties) for the Java process of PGX. Recommended Value: 58% of the PGX server memory limit size above.
`PGX_SERVER_YOUNG_SPACE_MB`	Indicates the amount of young space (new space) configured for the java heap.
Graph Service's SSL certificate details	
`STUDIO_SERVER_SSL_FILE_NAME`	Indicates the Graph Service SSL file name that is required for HTTPS configuration. For example， studio_server.p12
`STUDIO_SERVER_SSL_PASSWORD`	Indicates the password for Graph Service Studio Server P12 that is required for HTTPS configuration.
External Service Configuration	

Table 16: Config.sh file for PGX

Interaction Variable Name	Significance
GRAPH_SERVICE_URL	It indicates external service configuration where the Graph service is available. For example, https://<Compliance Studio fully qualified hostname>:7059/graph-service
GRAPH_KEYSTORE_PASSWORD	Indicates the password of the keystore file, which stores the password of the graph schemas.
Configuration to enable/disable loading graph from HDFS	
LOAD_GRAPH_FROM_HDFS	It is used to enable or disable loading graphs from HDFS. the value 'LOAD_GRAPH_FROM_HDFS' is "true" or "false" For example, LOAD_GRAPH_FROM_HDFS=false **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.** For Legacy Graph ETL, it is set to true.
Kerberos related configuration	
KERBEROS_TICKET_RENEWAL_PERIOD	For example, 7200 would mean every 2 hours **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**
KERBEROS_PRINCIPAL	For example, USER@PRINCIPAL **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**
KERBEROS_KEYTAB_FILENAME	For example, fccstudio.keytab **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**
KRB5_CONFIG_FILENAME	For example, krb5.conf **NOTE:** **This parameter is deprecated in the current release and will be removed in the future release.**
Advance Configuration	
SHUTDOWN_GRACE_PERIOD	It indicates the grace period in minutes for the graceful shutdown of the PGX Server. To set value, uncomment and set the value. **NOTE:** • The value should be an integer. • If the value is less than 1, then force shutdown is triggered immediately.

Table 16: Config.sh file for PGX

Interaction Variable Name	Significance
KERBEROS_RETRY_TIMEOUT	It indicates the retry timeout in minutes when kerberos ticket generation fails. To set value, uncomment and set the value. **NOTE:** The value should be an integer. **This parameter is deprecated in the current release and will be removed in the future release.**

NOTE

- You can generate `Graph-keystore.p12` file after starting the Compliance Studio.
- **For upgrade scenario**: Enter the value of this parameter from the previous version of Compliance Studio `config.sh` file (PGX service). If you are upgrading from 8.1.2.0.* to CS 8.1.2.5.0, then update this parameter accordingly.

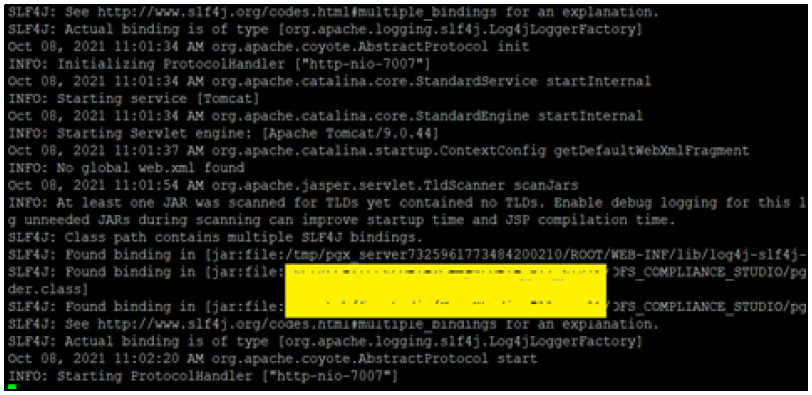
8. Navigate to the <`PGX_Installation_Path`>`/pgx-server/bin` directory and run any one of the following commands:

   ```
   ./pgx-server.sh –install
   ```

   Or

   ```
   ./pgx-server.sh -i
   ```

Figure 6: PGX start service



9. Start the PGX service.

   To start the PGX service, follow these steps:

   a. Navigate to the path where the PGX service is installed.

   b. Navigate to the following directory where the start service for PGX is located:

      `<PGX_Installation_Path>/pgx-server/bin`

   c. Run any one of the following commands:

      ```
      ./pgx-server.sh --start
      ```

      Or

```
./pgx-server.sh -s
```

10. Stop the PGX service.

    To stop the PGX service, run any one of the following commands:

    ```
    ./pgx-server.sh --stop
    ```

    Or

    ```
    ./pgx-server.sh -k
    ```

11. Force stop the PGX service.

    To force stop the PGX service, run any one of the following commands:

    ```
    ./pgx-server.sh --force-stop
    ```

    Or

    ```
    ./pgx-server.sh -f
    ```

12. Restart the PGX service.
13. Reinstall the PGX service.

    To reinstall the PGX service, run any one of the following commands:

    ```
    ./pgx-server.sh –reinstall
    ```

    Or

    ```
    ./pgx-server.sh -R
    ```

4.18 Generating Certificate for PGX Server

We recommend getting a certificate issued by a certificate authority (CA), which is trusted by your organization for the Linux server where the PGX server will be installed. If a CA certificate is not available, then generate it. To generate the certificate for PGX Server, see the Enable SSL for PGX Server or Generating a Self-Signed Server Keystore section.

4.18.1 Enable SSL for PGX Server

To generate a self-signed certificate by executing the following commands:

1. `openssl req -new -newkey rsa:2048 -days 365 -nodes -x509 -subj "/C=<Country Code>/ST=<State>/L=<Location>/O=< Organization >/CN=ROOT" -keyout ca_key.pem -out ca_certificate.pem`
2. `openssl genrsa -out server_key_traditional.pem 2048`
3. `openssl pkcs8 -topk8 -in server_key_traditional.pem -inform pem -out server_key.pem -outform pem -nocrypt`
4. `openssl req -new -subj "/C=<Country Code>/ST=<State>/L=<Location>/O=<Organization>/CN=<FQDN of PGX Server>" -key server_key.pem -out server.csr`
5. `chmod 600 server_key.pem`

6. `openssl x509 -req -CA ca_certificate.pem -CAkey ca_key.pem -in server.csr -out server_certificate.pem -days 365 -CAcreateserial`

> **NOTE** Replace the following placeholders:
> - <Country Code>
> - <State>
> - <Location>
> - <Organization>
> - <FQDN of PGX Server>: Fully qualified hostname of Linux server where PGX Server will be deployed

4.18.2 Generating a Self-Signed Server Keystore

To generate a self-signed server keystore by executing the following command:

1. `keytool -genkey -alias pgx -keyalg RSA -keystore server_keystore.jks`
2. Provide the requested details.

   For example:

   Enter keystore password:

   Re-enter new password:

   What is your first and last name?

   [Unknown]: my.hostname.domain.com

   What is the name of your organizational unit?

   [Unknown]: OU

   What is the name of your organization?

   [Unknown]: MyOrganization

   What is the name of your City or Locality?

   [Unknown]: MyTown

   What is the name of your State or Province?

   [Unknown]: MyState

   What is the two-letter country code for this unit?

   [Unknown]: US

   Is CN= my.hostname.domain.com, OU=OU, O=MyOrganization, L=MyTown, ST=MyState, C=US correct?

   [no]: yes

4.18.3 Configuring PGX Server

Users need to update `server.conf` and `config.sh` files for configuring the PGX server.

4.18.3.1 Updating server.conf File

To update the `server.conf` file, follow these steps:

1. Navigate to the `<PGX_SERVER_INSTALLATION_PATH>/pgx-server/conf/server.conf` directory.
2. If a server certificate is used, then update as follows:

```
{
  "port": 7007,
  "enable_tls": true,
  "context_path": "/",
  "working_dir": "<system-tmp-dir>",
  "server_private_key": "/path/of/server_key.pem",
  "server_cert": "/path/of/server_certificate.pem",
  "enable_client_authentication": false
}
```

> **NOTE** Replace the "/path/of/server_key.pem" and "/path/of/server_certificate.pem" with the correct path where the certificate is placed.

OR

If the server keystore file is used, then update as follows:

```
{
  "port": 7007,
  "enable_tls": true,
  "context_path": "/",
  "enable_client_authentication": false,
  "server_keystore": "/path/of /server_keystore.jks",
  "server_keystore_alias": "pgx",
  "server_keystore_type": "PKCS12",
  "server_keystore_provider": "SUN",
  "ca_certs": [],
  "working_dir": "<system-tmp-dir>"
```

```
}
```

> **NOTE** Replace the "/path/of /server_keystore.jks "with the correct path where the keystore is placed.

4.18.3.2 Updating config.sh File

If the PGX server keystore is used in the `server.conf` file, then the PGX_SERVER_KEYSTORE_PASSWORD parameter must be added to the `config.sh file` which is present in `<PGX_SERVER_INSTALLATION_PATH>/pgx-server/bin` directory.

For example;

```
export PGX_SERVER_KEYSTORE_PASSWORD=##PGX_SERVER_KEYSTORE_PASSWORD##
```

> **NOTE** Replace ##PGX_SERVER_KEYSTORE_PASSWORD## with the password created while generating the pgx server keystore file.

4.18.3.3 Trust Compliance Studio's SSL certificate

To trust Compliance Studio's SSL certificate, follow these steps:

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/conf` directory.
2. Obtain the Compliance Studio's SSL certificate, "`studio_server.p12`".
3. Generate the certificate in '`.cer`' format using "`studio_server.p12`" by executing the following command:

   ```
   keytool -exportcert -keystore <keystore path> -storepass <keystore password> -storetype PKCS12 -alias studio_server -file <ca_cert_dir>/studio_server.cer
   ```

4. Import the generated certificate to JAVA CA certs by executing the following command:

   ```
   keytool -importcert -keystore $JAVA_HOME/lib/security/cacerts -storepass changeit -alias studio_server -file <ca_cert_dir>/studio_server.cer
   ```

> **NOTE** Replace <keystore path> with the absolute path of "studio_server.p12" and replace <ca_cert_dir> with the directory where studio_server.cer should be generated.

4.18.3.4 Updating Start Script for PGX Server

To update the start script for the PGX server, follow these steps:

1. Navigate to `<PGX_SERVER_INSTALLATION_PATH>/pgx-server/bin` directory.
2. Update set_java_opts() as follows:

   ```
   set_java_opts() {
   ```

   ```
   export JAVA_OPTS="-Dpgx.max_off_heap_size=$PGX_SERVER_OFF_HEAP_MB -Xmx${PGX_SERVER_ON_HEAP_MB}m -Xms${PGX_SERVER_ON_HEAP_MB}m -XX:MaxNewSize=${PGX_SERVER_YOUNG_SPACE_MB}m -XX:NewSize=${PGX_SERVER_YOUNG_SPACE_MB}m -
   ```

```
Dpgx.keystore_path=$DEPLOY_APP_HOME/conf/$GRAPH_KEYSTORE_FILE -
Dpgx.keystore_password=$GRAPH_KEYSTORE_PASSWORD $EXTRA_JAVA_OPTS"
}
```

4.18.4 Configuring Compliance Studio Server

Users need to trust PGX Server's certificate for configuring the Compliance Studio server.

4.18.4.1 Trust PGX Server's certificate

To trust PGX Server's certificate, follow these steps:

1. Copy the "`ca_certificate.pem`" from the PGX server to the Compliance Studio server.
2. Import the copied certificate to the java ca certs by executing the following command:

   ```
   keytool -import -trustcacerts -keystore $JAVA_HOME/lib/security/cacerts
   -storepass changeit -alias pgx -file /path/of/ca_certificate.pem -
   noprompt
   ```

   NOTE Replace /path/of/ca_certificate.pem " with the path where the certificate is copied.

3. If the PGX server keystore is generated, copy the "`server_keystore.jks`" from the PGX server to the Compliance Studio server.
4. Import the copied keystore to the java ca certs by executing the following command:

   ```
   keytool -importkeystore -srckeystore /path/of/server_keystore.jks -
   destkeystore $JAVA_HOME/lib/security/cacerts -deststorepass changeit -
   srcstorepass <keystore password>  -noprompt
   ```

   NOTE Replace <keystore password> with the password generated while creating the pgx server keystore.

5. Update the PGX URL to set it as "https" using the following steps:
   a. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory.
   b. Open the `config.sh` file and update the "PGX_SERVER_URL" as https://<FQDN of PGX Server>:7007
   c. Reinstall Compliance Studio.

   OR

   To Update the PGX URL in an alternative way as follows:

   a. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-load-to-graph/graph-service/conf` directory.
   b. Open the `application.yml` and update the PGX_SERVER_URL.
   c. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/server/builtin/interpreters` directory.
   d. Open the `pgx.json` and update the PGX URL in the pgx interpreter's JSON file.
6. Restart Compliance Studio.

4.19 Run ER in different workspaces

1. The ER Data Schema and Compliance Studio Schema should be in the same wallet. For more information on how to create a wallet, see Create a wallet for ER/FSDF schema section.
2. Update the following details for ER schema in the `resources.xml` file. The file can be found in: `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/ficdb/conf`.

   Example:

```
<Resource
        id="ER2_CSA_ABCD"
        name="jdbc/erdataschema"
        auth="Container"
        type="javax.sql.DataSource"
        driverClassName="oracle.jdbc.OracleDriver"
        url="jdbc:oracle:thin:@ER2_CSA_ABCD"
   connectionProperties= "oracle.net.wallet_location
   =<WALLET_PATH/ABCD>;
oracle.net.tns_admin=<WALLET_PATH/ABCD>;”
             maxTotal="5"
             maxIdle="0"
             maxWaitMillis="-1" >
</Resource>
```

   NOTE Log in as either an SYS user or DBA user and grant these permissions to the ER schema created.

3. Ensure that the pre-staging and output tables are present in the given ER Data Schema.
   a. The following are the pre-staging table names by version:
      i. FSDF 81250:
      - STG_PARTY_MASTER_PRE
      - STG_PARTY_DETAILS_PRE
      - STG_DELETED_PARTIES_PRE
      - STG_CUSTOMER_IDENTIFCTN_DOC_PRE
      - STG_ADDRESS_MASTER_PRE
      - STG_PARTY_ADDRESS_MAP_PRE
      - STG_PARTY_PHONE_MAP_PRE
      - STG_PARTY_EMAIL_MAP_PRE
      - FCC_ER_MAPPING
      - FCC_ER_MANUAL_MAPPING

b. The following are the output table names by version:

ii. FSDF 81250:

- STG_PARTY_MASTER
- STG_PARTY_DETAILS
- STG_PARTY_EMAIL_MAP
- STG_ADDRESS_MASTER
- STG_PARTY_ADDRESS_MAP
- STG_PARTY_PHONE_MAP
- STG_CUSTOMER_IDENTIFCTN_DOC
- FCC_ER_MAPPING
- FCC_ER_OUTPUT

5 Post-installation Steps when OFSAA is installed

On successful installation of Compliance Studio, you must perform the following post-installation configurations.

> **ATTENTION** For the utility shell script and patch for Security Alert CVE-2021-44228, see Appendix E – Apache Log4j Security Alert CVE-2021-44228 Patch Details section.

Topics:

- Verify the Installation
- Start the PGX Service
- Access the Compliance Studio Application
- Perform the OFSAA Configuration for Batch Execution
- Configure and Run Published Notebooks
- Importing OOB Graph Definition and related Metadata
- Mapping Graph Datasource in Compliance Studio Workspace
- Using Graph Definition

> **NOTE** Before running the post-installation steps, an SSH connection to the Big Data server must be configured.

5.1 Verify the Installation

To verify the Compliance Studio installation with OFSAA, check the log files in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs` directory. If all the servers are up and running, it indicates that the installation is complete.

> **WARNING** If you notice any errors in the log files, do not proceed further. Contact My Oracle Support (MOS) provide the applicable error code and log files.

If the installation of Compliance Studio is unsuccessful, you must reinstall the application after performing the cleanup tasks. For more information, see Reinstalling Compliance Studio.

5.2 Start the PGX Service

To start the PGX service, follow these steps:

1. Navigate to the path where the PGX service is installed.
2. Navigate to the following directory where the start service for PGX is located:

   `<PGX_Installation_Path>/pgx-server/bin`

3. Run the following command:

   ```
   ./pgx-server.sh --start
   ```

> **NOTE** Make sure to update the correct location of `graph.json` and `csv files` in `config.sh` inside `<PGX Installation Path>/bin` directory before starting the PGX server.

For more information, see the OFS Compliance Studio Administration and Configuration Guide.

5.3 Access the Compliance Studio Application

To access Compliance Studio, follow these steps:

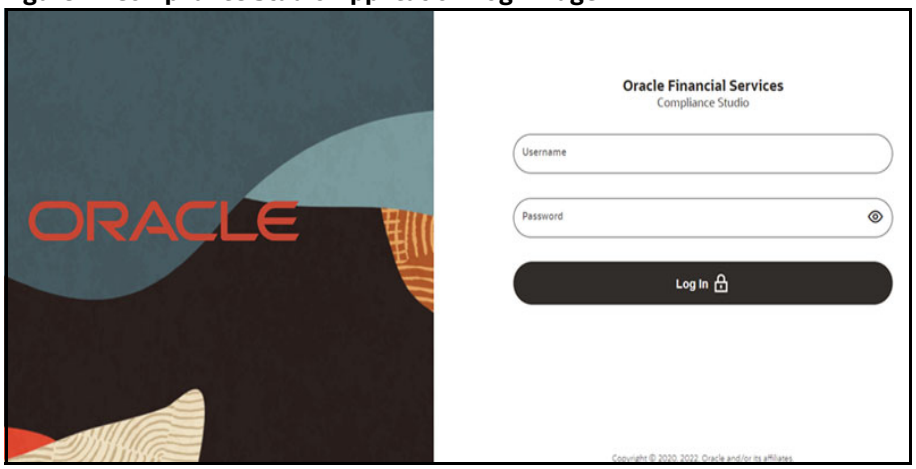
1. Enter the URL in the following format in the web browser:

   `https://<Host_Name>:<Port_Number>/cs/home`

   Here `<Port_Number>` is 7001 for the Compliance Studio application installed on-premise.

   The Compliance Studio application login page is displayed.

   Figure 7: Compliance Studio Application Login Page



2. Enter the Username and Password.

   For Creating Users, Groups, and Mappings in AAI. See Appendix E – Create Users, Groups, and Mappings section.

3. Click **Login**.

   After you access the application, you can view the ready-to-use notebooks. To check if you have been assigned any roles, create a notebook. If you cannot create a notebook, contact My Oracle Support (MOS).

5.4 Perform the OFSAA Configuration for Batch Execution

> **NOTE**
> - **This section is deprecated in the current release and will be removed in the future release.**
> - This configuration is not applicable for Compliance Studio installed without OFSAA.

To perform OFSAA configuration for batch execution, follow these steps:

1. Copy the files in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/ficdb/bin` directory to the server where the BD or ECM pack is installed and to the `$FIC_DB_HOME/bin` directory of the OFSAA setup.
2. Execute the following command to grant Execute permission to the files:

   ```
   chmod +x <filenames>
   ```

3. Copy all the files from the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/ficdb/lib` directory into the `$FIC_DB_HOME/lib` directory.

   See the OFS Compliance Studio Administration and Configuration Guide for running Compliance Studio Batches.

5.5 Configure and Run Published Notebooks

> **NOTE**
> - **This is deprecated in the current release and will be removed in the future release.**
> - This configuration is not applicable for Compliance Studio installed without OFSAA.

To perform the configuration required to run published notebooks, copy the required `FCCM_Studio_NotebookExecution.sh` file from the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/ficdb/bin` directory into the `<FIC_HOME of OFSAA_Installed_Path>/deployed/ficdb/bin` directory.

For information on running published notebooks, see the Executing Published Notebook section in the OFS Compliance Studio Administration and Configuration Guide.

5.6 Importing OOB Graph Definition and related Metadata

After completing the installation of Compliance Studio, you must run the following shell script to import the OOB Graph definition and related metadata:

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/ficdb/graph-metadata/bin` directory.
2. Run the following shell-scripts:
   - `InitializeOOBGraphBatchScheduleECM.sh`
   - `InitializeOOBGraphBatchScheduleBD.sh`
   - `InitializeGraphSchema.sh`
   - `InitializeECMSchema.sh`

- `InitializeBDSchema.sh`
- `CreateMetadataIndexes.sh`

3. To initialize BD schema, execute the following command:

   ```
   ./InitializeBDSchema.sh -bdw <bd_atomic_wallet_alias>
   ```

4. To initialize ECM schema, execute the following command:

   ```
   ./InitializeECMSchema.sh -w <ecm_schema_wallet_alias>
   ```

5. To initialize graph schema, execute the following command:

   For initializing BD graph pipeline:

   ```
   ./InitializeGraphSchema.sh -gw <graph_wallet_alias> -bs <bd_schema_name>
   ```

   For initializing ECM graph pipeline:

   ```
   ./InitializeGraphSchema.sh -gw <graph_wallet_alias> -es
   <ecm_schema_name>
   ```

   For initializing both BD and ECM graph pipeline:

   ```
   ./InitializeGraphSchema.sh -gw <graph_wallet_alias> -es
   <ecm_schema_name> -bs <bd_schema_name>
   ```

6. To initialize OOB graph batch schedule, execute the following command:

   For initializing BD graph schedule:

   ```
   ./InitializeOOBGraphBatchScheduleBD.sh -gw <graph_wallet_alias> -bw
   <bd_wallet_alias> -s <start_date> -e <end_date> -gd <graph_datasource>
   -u <complianceStudioUserName>
   ```

   For initializing ECM graph schedule:

   ```
   ./InitializeOOBGraphBatchScheduleECM.sh -gw <graph_wallet_alias> -ew
   <ecm_wallet_alias> -s <start_date> -e <end_date> -gd <graph_datasource>
   -u <complianceStudioUserName>
   ```

7. If metadata indices are not available in the OpenSearch, then execute the following script to load metadata indices:

   ```
   ./CreateMetadataIndexes.sh
   ```

> **NOTE** If `./CreateMetadataIndexes.sh` script is not executed, ensure that F_IS_RECENTLY_CHANGED column values are **N** in the FCC_IDX_M_LOOKUP table for all the records. For more information, see **Default Data in the tables** section in the OFS Compliance Studio Administration and Configuration Guide.

5.6.1 Importing OOB graph

To import OOB graph, follow these steps:

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/ficdb/graph-metadata/graph` directory.
2. Download zip file from the ECM and BD folders.
3. Import graph from the graph summary screen in the Compliance Studio workspace. To import graph, see **Adding a Graph Pipeline** section in the OFS Compliance Studio User Guide.

5.6.2 Cleanup Steps when Import Failed in Graph Pipeline

After installation, query the table FCC_M_PIPELINE_IMPORT_LOG to check the imported pipeline status. The _V_IMPORT_STATUS_ column denotes the status and should be 'SUCCESS' for all the imported pipelines.

If any pipelines have the status of 'FAILED,' perform the following steps to reimport:

1. Find the entry for the failed pipeline in the **FCC_M_EXTERNALSERVICE_RUN** table of this **_C_TABLELIST_** column.
2. Remove that entry from the table. If the entry is not there, skip this step.
3. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-pipeline/pipeline/data-metadata-job-<version>/bin` directory.
4. Run the `import_metadata.sh` shell script using the following command:

   ```
   ./import_metadata.sh
   ```

5. Once the script is executed, verify the status in the **FCC_M_PIPELINE_IMPORT_LOG** to ensure that the status is a success.

5.7 Mapping Graph Datasource in Compliance Studio Workspace

For details, see the **Mapping Graph Datasource** section in the OFS Compliance Studio User Guide.

5.8 Using Graph Definition

For details, see **Using Graph Definition** section in the OFS Compliance Studio User Guide.

5.9 Additional Grants for Graph Schema

For post-installation grants of the graph schema, see the Create Graph Schema and Grant Permission section.

6 Post-installation Steps when OFSAA is Not Installed

On successful installation of Compliance Studio, you must perform the following post-installation configurations.

> **ATTENTION** For the utility shell script and patch for Security Alert CVE-2021-44228, see Appendix E – Apache Log4j Security Alert CVE-2021-44228 Patch Details section.

Topics:

- Verify the Installation
- Start the PGX Service
- Access the Compliance Studio Application

> **NOTE** Before running the post-installation steps, an SSH connection to the Big Data server must be configured.

6.1 Verify the Installation

To verify the Compliance Studio installation without OFSAA, check the log files in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs` directory. If all the servers are up and running, it indicates that the installation is complete. Also, ensure all the interpreters are displayed and the JDBC interpreter is working on the Compliance Studio application home page.

> **WARNING** If you notice any errors in the log files, do not proceed further. For additional information, see the Frequently Asked Questions in Compliance Studio section first and Contact My Oracle Support (MOS) provide the applicable error code and log files.

If the installation of Compliance Studio is unsuccessful, you must reinstall the application after performing the cleanup tasks. For more information, see Reinstalling Compliance Studio.

6.2 Start the PGX Service

To start the PGX service, follow these steps:

1. Navigate to the path where the PGX service is installed.
2. Navigate to the following directory where the start service for PGX is located:

   ```
   <PGX_Installation_Path>/pgx-server/bin
   ```

3. Run the following command:

   ```
   ./pgx-server.sh --start
   ```

For more information, see the OFS Compliance Studio Administration and Configuration Guide.

6.3 Access the Compliance Studio Application

To access Compliance Studio, follow these steps:

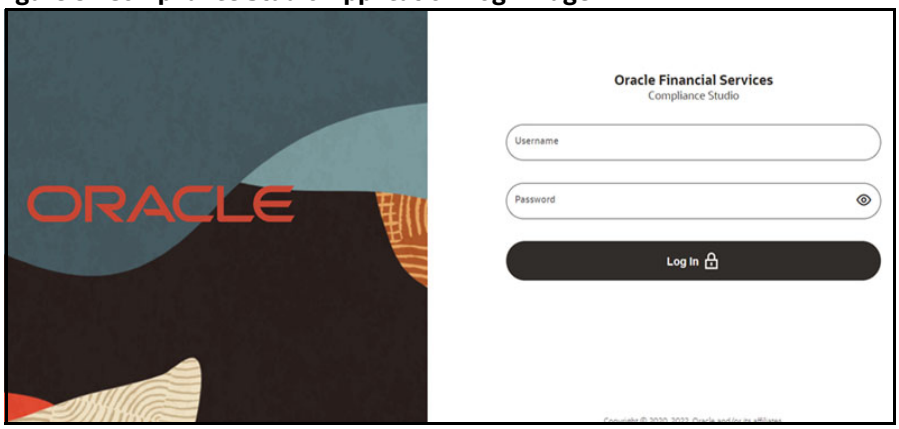
1. Enter the URL in the following format in the web browser:

   `https://<Host_Name>:<Port_Number>/cs/home`

   Here `<Port_Number>` is **7001** for the Compliance Studio application installed on-premise.

   The Compliance Studio application login page is displayed.

   Figure 8: Compliance Studio Application Login Page

2. Enter the **Username** and **Password**.
3. Click **Login**.

   After you access the application, you can view the ready-to-use notebooks. To check if you have been assigned any roles, create a notebook. If you cannot create a notebook, contact My Oracle Support (MOS).

7 Upgrade

Follow these steps to upgrade an existing instance of Compliance Studio.

Topics:

Customization in Entity Resolution

If any customization is applied before the upgrade, then re-check and evaluate whether these changes are still in place after the upgrade. If changes are not available, re-apply the customization wherever required.

Out-of-the-box rules should not be edited for customizations. If there are any customizations, create a copy of out-of-the-box pipeline definitions to apply any customizations otherwise the customizations will not persist when upgraded.

If out-of-the-box pipeline definitions are modified for any customizations, follow these steps to preserve the customizations:

1. Before upgrade, create a copy of the modified/customized pipeline definitions.
2. Merge the customizations into upgraded out-of-the-box pipeline definitions appropriately after the successful upgrade of Compliance Studio 8.1.2.5.0.

   For any clarifications, contact My Oracle Support (MOS).

7.1 Pre-upgrade Steps

To do pre-upgrade, follow these steps:

1. Stop the existing Compliance Studio service. To stop, see the Stopping Compliance Studio section.
2. Stop the pgx server. To stop, see the Stop the PGX Service section.

7.2 Upgrade Steps with OFSAA

This section describes generic steps for the upgrade. For specific upgrades, see Cleanup for Upgrade section.

Table 16 provides the steps to upgrade Compliance Studio with OFSAA.

Table 17: Upgrade Steps with OFSAA

Sl. No.	Activity
Pre-installation Steps	
1	Hardware and Software Requirements
2	Download the Installer Kit
Installation Steps	
1	Extract the Installer Kit
2	Configure the OpenSearch Component
3	Add Synonyms and Stopword files in OpenSearch
4	Place Files in the Installation Directories
5	Generate the Public and Private Keys
6	Generate API token for CS API User
7	Generate Compliance Studio Server SSL Configuration Mandatory File
8	Import the certificate to JDK security
9	Place the Key Store File for Secure Batch Service
10	Configure the Extract Transfer and Load (ETL) Process **NOTE:** **This step is deprecated in the current release and will be removed in the future release.**
11	Additional Grants for Studio Schema (See Assign Grants for the Studio Schema)
12	Additional Grants for ER Schema (See Assign Grants for the Studio Schema)
13	Create Graph Schema and Grant Permission
14	Configure the config.sh File
15	Generate the Graph-keystore.p12 File
16	Configure the PGX Service
17	Run the Compliance Studio Installer
Post-Installation Steps	
1	Verify the Installation
2	Cleanup for Upgrade
3	Stop the PGX Service

Table 17: Upgrade Steps with OFSAA

4	Stop the Compliance Studio Installer
5	**Configure the SSH Connection** – See OFS Compliance Studio Administration and Configuration Guide
6	**Configure the Schema Creation** – See OFS Compliance Studio Administration and Configuration Guide. **NOTE:** This is applicable only if you want to load a graph using the Legacy ETL, i.e., the Hadoop cluster.
7	**Configure the ICIJ Data** – See OFS Compliance Studio Administration and Configuration Guide. **NOTE:** This is applicable only if you want to load a graph using the Legacy ETL, i.e., the Hadoop cluster.
8	Start the PGX Service
9	Starting Compliance Studio
10	Configure the Graph Pipeline (For Graph Pipeline)
11	Configure the Entity Resolution (For Entity Resolution)
12	Access the Compliance Studio Application

7.3 Cleanup for Upgrade

This section provides cleanup steps for the upgrade.

7.3.1 Perform Extract Transfer and Load (ETL) Cleanup

> **NOTE** **This section is deprecated in the current release and will be removed in the future release.**

To perform the ETL cleanup, follow these steps:

- Extract the contents of the installer archive file in the download directory using the `unzip -a <Compliance_Studio_Installer_Archive_File>.zip`. The Compliance Studio installer file is extracted in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>` directory.
- Configure the applicable parameters in the `config.sh` file. For more information, see Configure the config.sh File.
- Generate the keystore file. For more information, see Generate Compliance Studio Server SSL Configuration Mandatory File.

7.3.2 Perform Cleanup for Templates

> **NOTE** This is applicable only if you want to use the new FCGM Default Template. Otherwise, the template will not be updated.

To delete the templates, perform the following:

1. Log in to the Compliance Studio application.

2. Launch the **CS Production** Workspace.
3. Hover the mouse over the **Data Studio Options** widget and Click **Templates.**

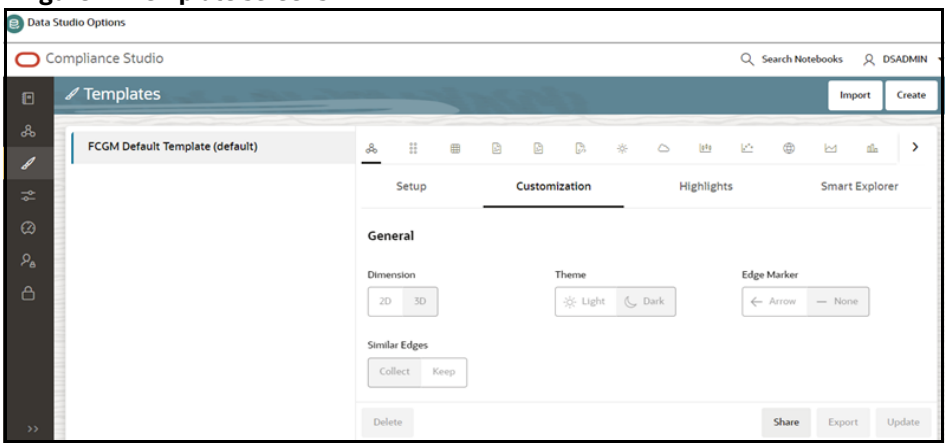
   By default, the Templates page lists all the available templates.

   You can see the following templates among all the templates:

   - FCGM Default Template (default)
   - FCGM Default Template

   You should delete the **FCGM Default Template** that is without **(default).**
4. Click the **FCGM Default Template** on the LHS. The default details are displayed on the RHS:

Figure 9: Template screens



5. Click **Delete** on the RHS. A confirmation message is displayed for deletion.
6. Click **Delete**. The template will be deleted.

7.3.3 Perform Cleanup for Interpreters

NOTE

- Ensure that the following interpreters are deleted:
  - fcc-jdbc
  - fcc-ore
  - fcc-pyspark
  - fcc-spark-scala
  - fcc-spark-sql
- For 8.1.2.0.0 and later versions, you can rename the fcc interpreter variants in all cases except for the different Python Virtual Environments, so simpler interpreter's names will be used.
- The steps in this section explain removing the fcc versions before installing the generic versions
- See **Create an Interpreter Variant** in the OFS Compliance Studio Administration and Configuration Guide on creating new interpreter variants if you want to use the notebooks that use the deleted interpreter name.

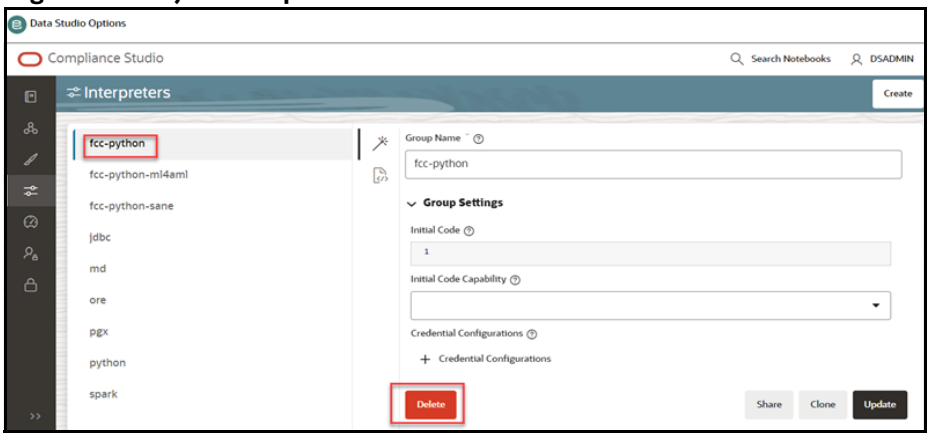
NOTE

- For example, if the notebook has an **fcc-jdbc** paragraph, and these paragraphs' interpreter cannot be replaced with **jdbc**, you can create/clone an interpreter variant of jdbc with the name **fcc-jdbc**.

To delete the interpreter, perform the following:

1. Log in to the Compliance Studio application.
2. Launch the **CS Production** Workspace.
3. Hover the mouse over the **Data Studio Options** widget and Click **Interpreters**.
4. By default, the Interpreters page lists all the available interpreters.
5. Click the **fcc-jdbc** interpreter on the LHS. The default configured interpreter variant is displayed on the RHS:

Figure 10: fcc-jdbc interpreter screen



6. Click **Delete** on the RHS. A confirmation message is displayed for deletion.
7. Click **Delete**. The template will be deleted.
8. Repeat the steps **4, 5,** and **6** for the following interpreters:
   - fcc-ore,
   - fcc-pyspark,
   - fcc-spark-scala
   - fcc-spark-sql

7.3.4 Perform Cleanup for Entity Resolution

You can follow the approach based on the following scenario:

In case of resetting Entity Resolution completely, see the **Resetting Entity Resolution Back to Day 0** section in the OFS Compliance Studio Administration and Configuration Guide.

In case of detailed cleanup steps to continue with Entity Resolution, you can contact My Oracle Support (MOS).

7.4 Stop the PGX Service

To stop the PGX service, follow these steps:

1. Navigate to the path where the PGX service is installed.
2. Navigate to the following directory where the start service for PGX is located:

   `<PGX_Installation_Path>/pgx/pgx-server/bin`
3. Run `./pgx-server.sh --stop`.

7.5 Stop the Compliance Studio Installer

To stop the Compliance Studio installer, follow these steps:

1. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin/`directory.
2. Run `./compliance-studio.sh -k`

7.6 Upgrade Steps without OFSAA

Table 18 lists the steps to upgrade Compliance Studio without OFSAA.

Table 18: Upgrade Steps without OFSAA

Sl. No.	Activity
Pre-installation Steps	
1	Download the Installer Kit
Installation Steps	
1	Extract the Installer Kit
2	Place Files in the Installation Directories
3	Generate API token for CS API User
4	Generate the Public and Private Keys
5	Place the Key Store File for Secure Batch Service
6	Configure the config.sh File
7	Run the Compliance Studio Installer
Post-Installation Steps	
1	Stop the Compliance Studio Installer
2	**Add the Python Packages to Compliance Studio** - See OFS Compliance Studio Administration and Configuration Guide
3	**Configure the SSH Connection** – See OFS Compliance Studio Administration and Configuration Guide
4	Starting Compliance Studio

Table 18: Upgrade Steps without OFSAA

5	Access the Compliance Studio Application

7.7 Configure Python Interpreter Setting

To use a python interpreter in an upgraded environment, you need to configure the following:

Zeppelin.python:

```
<COMPLIANCE STUDIO INSTALLATION PATH>/deployed/python-packages/
defaultVirtualEnv/bin/python3
```

Initialization:

```
import os; os.environ['TNS_ADMIN'] = '<WALLET_LOCATION>';
from ds_interpreter_client.context.ds_context import PyDataStudioContext
ds = PyDataStudioContext()
```

To configure, perform the following:

1. Login to the Compliance Studio application.
2. Launch the **CS Production** Workspace.
3. Hover the mouse over the Data Studio Options  widget and Click **Interpreters**.

   By default, the Interpreters page lists all the available interpreters.
4. Click the **fcc-python** interpreter on the LHS. The default configured interpreter variant is displayed on the RHS:

Figure 11: fcc-python interpreter screens

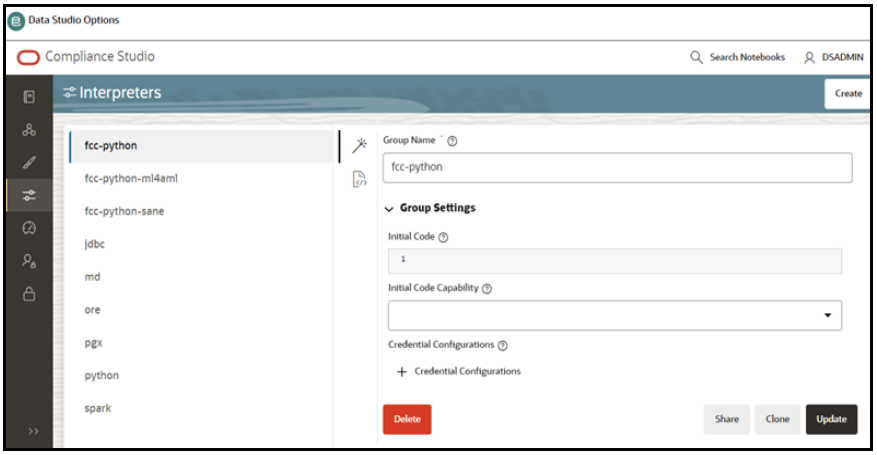
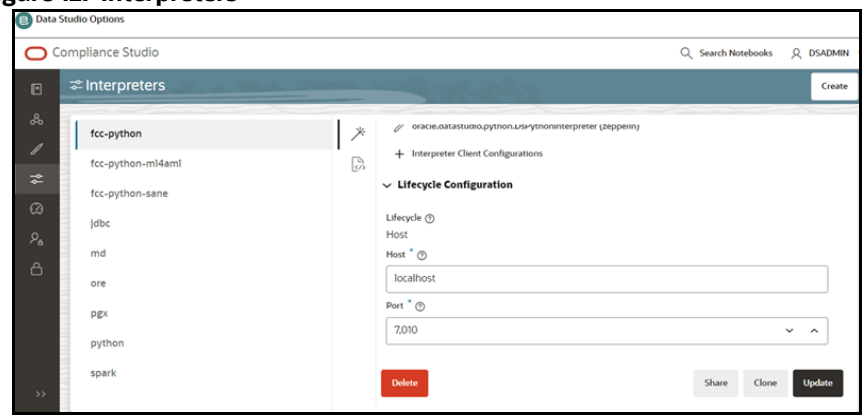


Figure 12: Interpreters



7.8 Configure Interpreters after Upgrade

To change the port numbers after upgrade in the interpreters, follow these steps:

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/server/builtin/interpreters` directory.

   The interpreter's `.json` files are available.

   For example,

2. Click the `fcc-python.json` file to view the port number which has to be modified in the Compliance Studio application.

To configure interpreters after the upgrade, perform the following:

1. Login to the Compliance Studio application.
2. Launch the **CS Production** Workspace.
3. Hover the mouse over the Data Studio Options  widget and Click **Interpreters.**

   By default, the Interpreters page lists all the available interpreters on the LHS.

Figure 13: Port Number Configuration

4. Click **fcc-python** interpreter on the LHS and then click **Plain Configuration** on the RHS.
5. Update the port number as per the `fcc-python.json` file.

Similarly, modify all the port numbers available on the Interpreters page with respect to JSON files.

7.9 Upgrade Steps for Graph Pipeline

To upgrade the graph pipeline, follow these steps:

1. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/ficdb/GraphPipeline-Cleanup-Scripts` directory.
2. Cleanup the existing graph using the following scripts:

   `GraphPipeline_cleanup_day0_in_studioschema.sql`

   It should be executed from the Compliance Studio schema.

   `GraphPipeline_cleanup_day0_in_graphschema.sql`

   It should be executed from the graph schema. For more information, see the **Resetting Graph Pipeline Back to Day 0** section in the OFS Compliance Studio Administration and Configuration Guide.

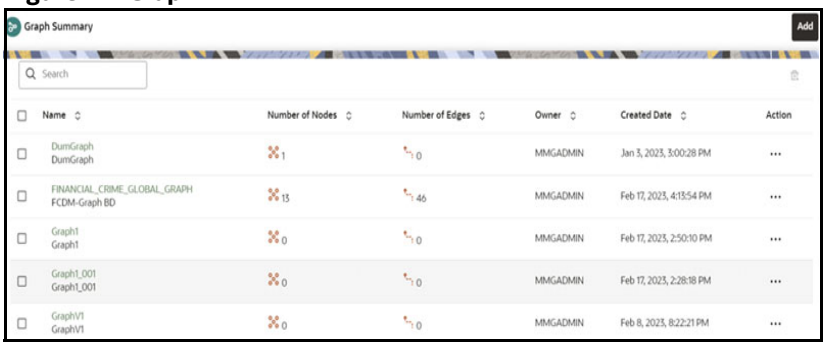
   `cleanup_es_indexes.sh`

   It should be executed from previous version (v8.1.2.1.0 or v8.1.2.3.0) of the Compliance Studio.
3. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/ficdb/graph-metadata/upgrade` directory.
4. Execute the following command to upgrade the new graph model:

   `update-oob-graph.sh`
5. Restart PGX server.
6. Login to Compliance Studio.
   a. Launch the **CS Production** Workspace.
   b. In the LHS menu, click **Graphs**.

      The new OOB graph pipeline is displayed in the graph summary page.

      Figure 14: Graph

   c. Update the OOB graph refresh schedules for the new graph pipeline.

      To create a Graph Refresh Schedules, see **Creating Graph Refresh Schedules** section in the OFS Compliance Studio User Guide.

8 Reinstall Compliance Studio for Cleanup of Studio and BD/Atomic Schema

> **NOTE** All the user data created after Compliance Studio installation, such as notebooks and customization will be cleaned.

If the installation of Compliance Studio is unsuccessful, you must reinstall the application after performing the required cleanup tasks.

To reinstall Compliance Studio for cleanup of Studio and BD/Atomic schema, follow these steps:

1. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory to update `config.sh` file.
2. Perform the database cleanup for the following schemas:

   Table 19 lists Schemas applicable for cleanup.

   Table 19: Schemas applicable for cleanup

Schema	Applicable for Compliance Studio with OFSAA	Applicable for Compliance Studio without OFSAA
Clean up for Compliance Studio Schema	Yes	Yes
Cleanup for BD or ECM Atomic Schema	Yes	No

3. Run the following command:

   `./compliance-studio.sh –k and ./compliance-studio.sh -R`
4. Reinstall Compliance Studio.

Topics:

- Clean up for Compliance Studio Schema
- Cleanup for BD or ECM Atomic Schema

8.1 Clean up for Compliance Studio Schema

To clean up the Studio schema, follow these steps:

1. Drop the existing Compliance Studio schema and create a new Studio schema.

   > **NOTE** The username and password credentials of the Compliance Studio Schema in the wallet files must be updated accordingly. (If applicable)

2. Grant permission to the newly created Oracle Database Schema. For more information about grant, see the Assign Grants for the Studio Schema section.

   > **NOTE** Note that the tablespace size can be as per the user's requirement.

8.2 Cleanup for BD or ECM Atomic Schema

To clean up the BD or ECM Atomic schema, follow these steps:

1. Login to the BD or ECM Atomic Schema.
2. Truncate the `DATABASECHANGELOG` and `DATABASECHANGELOGLOCK` tables using the following command:

```
TRUNCATE TABLE DATABASECHANGELOGLOCK;

TRUNCATE TABLE DATABASECHANGELOG;
```

9 Setup Compliance Studio Instance for Cloning the Filesystem

This section describes the process of cloning files from the Compliance Studio's Primary server to the Compliance Studio's Secondary server for the purpose of disaster recovery. For more information about disaster recovery, see the **Setup Disaster Recovery (DR) in Compliance Studio** section in the OFS Compliance Studio Architecture Guide.

9.1 Prerequisites

- OpenSearch should be installed in the Secondary server.

> **NOTE** The following prerequisites are applicable for the case where Database schemas (for example: Studio Schema, ECM/BD Atomic Schema, and Graph Schema) are cloned in a different database.

- Create a wallet for the Secondary Database. To create a wallet, see the Setup the Password Stores for Database User Accounts section.
- Add Secondary database credentials in the wallet.
- ER/FSDF Schema, Atomic Schema, Studio Schema, and Graph Schema names should be the same as the Primary Database.

9.2 Cloning Process

To clone files in the Compliance Studio, follow these steps:

1. Zip the **OFS_COMPLIANCE_STUDIO** and **Logstash-<version>** directories from the Compliance Studio's Primary Server.

> **NOTE** The Logstash for OpenSearch is different, so based on the configuration the respective logstash should be configured. Logstash version depends on the OpenSearch version.

2. To zip those directories, execute the following command:

   ```
   zip -r <DIRECTORY_NAME>.zip <DIRECTORY_NAME>
   ```

   For example, `zip -r OFS_COMPLIANCE_STUDIO.zip  OFS_COMPLIANCE_STUDIO`
3. Copy the zip file into the Compliance Studio's Secondary server.
4. Unzip the file from the Compliance Studio's Secondary server by executing the following command:

   ```
   unzip -a <zip_file_name>.zip
   ```

9.2.1 Cloning OpenSearch

To clone the OpenSearch, follow these steps:

1. Copy the folder "data" from this `<OS_Installation_Path>/opensearch-<version>` directory in the OpenSearch's Primary server.

   Where `<OS_Installation_Path>` refers to OpenSearch installed path.

2. Place the copied "data" folder into `<OS_Installation_Path>/opensearch-<version>` directory of the OpenSearch's Secondary server.

If HTTPS and AUTH are enabled for OpenSearch, then follow these steps:

1. To generate `ca.crt` file in the OpenSearch's Secondary server, execute the following command:

   ```
   openssl x509 -outform der -in <path to/admin.pem> -out ca.crt
   ```

2. Copy `ca.crt` file from OpenSearch's Secondary server and place in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/logstash/config` directory.
3. To generate `admin.p12` file in the OpenSearch's Secondary server, execute the following command:

   ```
   openssl pkcs12 -export -out admin.p12 -inkey <path to/admin-key.pem> -in
   <path to/admin.pem>
   ```

4. Copy `admin.p12` file and place it in the following directories of the Compliance Studio's server:

   `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/load-to-open-search/conf`

   `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/load-to-open-search/conf`

   `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/matching-service/conf`

   `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/matching-service/conf`

9.2.2 Cloning PGX Service

> **NOTE** This section is applicable only for Graph use case.

To clone the PGX service, follow these steps:

1. Copy the `studio_server.p12` file from `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-studio/conf` directory of the Compliance Studio's Primary server and place it to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/pgx-server/conf` directory in the Compliance Studio's Secondary server.
2. Generate the `graph-keystore.p12` file for PGX's Secondary server. To generate `graph-keystore.p12` file , see the Generate the Graph-keystore.p12 File section.
3. Navigate to the `<PGX_HOME>/pgx/pgx-server/bin` directory.
4. Open the `config.sh` file and update the following parameter.

   ```
   GRAPH_SERVICE_URL=##SECONDARY_GRAPH_SERVICE_URL##
   GRAPH_KEYSTORE_PASSWORD=##SECONDARY_GRAPH_KEYSTORE_PASSWORD##
   ```

   The path where the `pgx-server-<version>.zip` file is unzipped and it is referred to as **<PGX_HOME>**. For more information, see the Configure the PGX Service section.

5. To reinstall the PGX service, execute the following command:

   ```
   ./pgx.sh --reinstall
   ```

6. To start the PGX service, execute the following command:

   ```
   ./pgx.sh --start
   ```

9.2.3 Configuring config.sh File

> **NOTE** This section is applicable when Database schemas are cloned to the different database.

To configure the `config.sh` file, follow these steps:

1. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory in Compliance Studio's Secondary server.
2. Open the `config.sh` file and update parameters as mentioned in Table 20.

Table 20: Parameter Values in Config.sh File

Parameter	Placeholder Value
COMPLIANCE_STUDIO_INSTALLATION_PATH	##SECONDARY_SERVER_COMPLIANCE_STUDIO_PATH## **NOTE**: If the OFS_COMPLIANCE_STUDIO path is same in the Compliance Studio's Primary and Secondary Servers, then this parameter need not be changed.
DB Details of Studio Schema **NOTE**: This parameter is applicable when Database schemas are cloned to the different databases.	
STUDIO_DB_HOSTNAME	## SECONDARY_STUDIO_DB_HOSTNAME ##
STUDIO_DB_PORT	## SECONDARY _STUDIO_DB_PORT##
STUDIO_DB_SERVICE_NAME	## SECONDARY _STUDIO_DB_SERVICE_NAME ##
STUDIO_DB_SID	## SECONDARY _STUDIO_DB_SERVICE_NAME ##
STUDIO_DB_USERNAME	For example, CS81250_DR_2477
DB Details of Atomic Schema **NOTE**: This parameter is applicable when Database schemas are cloned to the different databases.	
ATOMIC_DB_HOSTNAME	## SECONDARY _ATOMIC_DB_HOSTNAME ##
ATOMIC_DB_PORT	## SECONDARY _STUDIO_DB_PORT##
ATOMIC_DB_SERVICE_NAME	## SECONDARY _ATOMIC_DB_SERVICE_NAME ##
ATOMIC_DB_SID	## SECONDARY _ATOMIC_DB_SERVICE_NAME ##
ATOMIC_DB_USERNAME	For example, STD_ATOM8125
Graph Schema Configuration **NOTE**: This parameter is applicable when Database schemas are cloned to the different databases.	
GRAPH_DB_SERVER_NAME	## SECONDARY _GRAPH_DB_SERVER_NAME ##
GRAPH_DB_PORT	## SECONDARY_GRAPH_DB_PORT ##
GRAPH_DB_SERVICE_NAME	## SECONDARY _GRAPH_DB_SERVICE_NAME ##

Table 20: Parameter Values in Config.sh File

Parameter	Placeholder Value
GRAPH_KEYSTORE_PASSWORD	For example, password123
GRAPH_SCHEMA_DB_SCHEMA_NAME	For example, GS81250_DR_2477
GRAPH_SCHEMA_WALLET_ALIAS	For example, GS81250_DR_2477_alias
GRAPH_SCHEMA_WALLET_LOCATION	##POINTING_TO_DR_DB##
GRAPH_SCHEMA_TNS_ADMIN_PATH	##POINTING_TO_DR_DB##
Wallet Details **NOTE**: This parameter is applicable when Database schemas are cloned to the different databases.	
WALLET_LOCATION	##POINTING_TO_DR_DB##
TNS_ADMIN_PATH	##POINTING_TO_DR_DB##
Service Urls	
PGX_SERVER_URL	##SECONDARY_PGX_SERVER_URL##

NOTE For the parameter description, see the Configure the config.sh File section.

3. Add Secondary Database credentials in the wallet. To add credentials, see the Setup the Password Stores for Database User Accounts section.
4. In Studio Schema, delete the following row from the DATABASECHANGELOG table:

   author = 'Compliance Studio 8.1.2.1' and id = 'FCC_DATASTUDIO_CONFIG_8121'
5. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory.
6. Reinstall Compliance Studio's Secondary Server. To reinstall, execute the following command:

   ```
   ./compliance-studio.sh --reinstall
   ```
7. Start Compliance Studio's Secondary Server. To start, execute the following command:

   ```
   ./compliance-studio.sh --start
   ```
8. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/ficdb/conf` directory.
9. Open the `resources.xml` file and update the following details for ER/FSDF schema.

   Example:

```
<Resource
        id="##ER_DATA_SCHEMA_ALIAS_NAME##"
        name="jdbc/erdataschema"
        auth="Container"
        type="javax.sql.DataSource"
        driverClassName="oracle.jdbc.OracleDriver"
```

```
        url="jdbc:oracle:thin:@##ER_DATA_SCHEMA_ALIAS_NAME##"

        connectionProperties="oracle.net.wallet_location=/scratch/
fccstudio/CS81250_CS_Cloning_0204/compStudio_02040741/
OFS_COMPLIANCE_STUDIO/wallet;oracle.net.tns_admin=/scratch/fccstudio/
CS81250_CS_Cloning_0204/compStudio_02040741/OFS_COMPLIANCE_STUDIO/
wallet;"

        maxTotal="20"

        maxIdle="0"

        maxWaitMillis="-1">

    </Resource>
```

10 Frequently Asked Questions (FAQs) and Error Dictionary

This section consists of resolutions to the frequently asked questions and error codes noticed during the Compliance Studio installation.

Topics:

The Compliance Studio installer performs all the pre-requisite validation checks during installation. Any error encountered in the process is displayed with an appropriate Error Code. You can refer to the Error Dictionary to find the exact cause and resolution to rectify the error.

10.1 Frequently Asked Questions in Compliance Studio

You can refer to the Frequently Asked Questions, which are developed with interest to help you resolve some of the Compliance Studio Installation and configuration issues. This intends to share problem resolution knowledge to a few of the known issues. This is not an official support document and just attempts to share problem resolution knowledge to a few known issues.

1. Why does my console show an unsuccessful message during wallet creation?

   You can check if you have run the following commands correctly. For more information on wallet creation, see Setup Password Stores with Oracle Wallet.

   a. `mkstore -wrl <wallet_location> -create` //creates a wallet in the specified location

   b. `mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name>` //creates an alias in the studio schema

   c. `mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name>` //creates an alias in the atomic schema

   d. `mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name>` //creates an alias in the config schema

   If your issue is still not resolved, contact My Oracle Support (MOS).

2. Where can I find my created wallet?

   Your wallet will be in the directory you have set as your wallet location.

   If your issue is still not resolved, contact My Oracle Support (MOS).

3. When should I create a Database link, and if yes, how do I do it?

   Create a Database link to connect the Atomic and Config database schemas to the Studio database schema if the databases are different. You must create the link in the Studio database.

   In the following example, a link has been created from the config schema to the atomic schema by running the following script:

   ```
   create public database link <studio database link>

   connect to <Config Schema>

   identified by password

   using ' (DESCRIPTION = ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP) (HOST
   =<host name> (PORT = <port number>)) (CONNECT_DATA = (SERVICE_NAME =
   <service name>))) ';
   ```

```
Config schema : <Config Schema>/password

' (DESCRIPTION = ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP) (HOST =<host
name> (PORT = <port number>)) (CONNECT_DATA = (SERVICE_NAME = <service
name>))) ';
```

After running the script, run the FCDM connector and ICIJ connector jobs.

4. Why does my installed studio setup not have any notebooks?

   Some default notebooks are ready to use when you install Compliance Studio. If you do not see any notebooks when you log in to the application, you may not be assigned any roles. Check the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs` directory to see if you have been assigned any roles, and if not, contact your administrator.

   If your issue is still not resolved, contact My Oracle Support (MOS).

5. What can I do if the schema creation fails?

   If the Atomic schema creation fails, login to the BD and ECM Atomic schemas and run the following query:

   ```
   select * from fcc_orahive_datatypemapping;
   ```

   The `fcc_orahive_datatypemapping` table must not have duplicate data types.

   If the Compliance Studio schema creation fails, login as a Studio user and run the following query:

   ```
   select * from fcc_datastudio_schemaobjects
   ```

   Run the following query to replace all `Y` values with '':

   ```
   update fcc_datastudio_schemaobjects set SCHEMA_OBJ_GENERATED=''
   ```

   After the schema creation is successful, the value of the `SCHEMA_OBJ_GENERATED` attribute changes to Y.

   You can also check for errors in the application log file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs` directory.

   If your issue is still not resolved, contact My Oracle Support (MOS).

6. What can I do if the Import_training_model batch execution fails?

   Batch execution status always displays success in case of success or failure.

   You can also check for errors in the application log file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs` directory.

   According to the log details, you can fix the failure and rerun the same batch.

7. Why is the PGX Server not starting?

   The PGX server starts only after the FCDM tables are created after the FCDM connector job is run. Check if all FCDM tables are created, and start the PGX server. You can also check for any errors in the application log file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/logs` directory.

   If your issue is still not resolved, contact My Oracle Support (MOS).

8. What should I do if there is a below Error while selecting edges in manual Decision UI?

   ```
   java.lang.IllegalStateException: Unable to create
   PgxSessionWrapperjava.lang.IllegalStateException: Unable to create
   ```

```
PgxSessionWrapper at
oracle.datastudio.interpreter.pgx.CombinedPgxDriver.getOrCreateSession(C
ombinedPgxDriver.java:147) at
oracle.pgx.graphviz.driver.PgxDriver.getGraph(PgxDriver.java:334) at
oracle.pgx.graphviz.library.QueryEnhancer.createEnhancer(QueryEnhancer.j
ava:223) at
oracle.pgx.graphviz.library.QueryEnhancer.createEnhancer(QueryEnhancer.j
ava:209) at
oracle.pgx.graphviz.library.QueryEnhancer.query(QueryEnhancer.java:150)
at
oracle.pgx.graphviz.library.QueryEnhancer.execute(QueryEnhancer.java:136
) at
oracle.pgx.graphviz.interpreter.PgqlInterpreter.interpret(PgqlInterprete
r.java:131) at
oracle.datastudio.interpreter.pgx.PgxInterpreter.interpret(PgxInterprete
r.java:120) at
org.apache.zeppelin.interpreter.LazyOpenInterpreter.interpret(LazyOpenIn
terpreter.java:103) at
org.apache.zeppelin.interpreter.remote.RemoteInterpreterServer$Interpret
Job.jobRun(RemoteInterpreterServer.java:632) at
org.apache.zeppelin.scheduler.Job.run(Job.java:188) at
org.apache.zeppelin.scheduler.FIFOScheduler$1.run(FIFOScheduler.java:140
) at java.base/
java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:515)
at java.base/java.util.concurrent.FutureTask.run(FutureTask.java:264) at
java.base/
java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.run
(ScheduledThreadPoolExecutor.java:304) at java.base/
java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.jav
a:1128) at java.base/
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.ja
va:628) at java.base/java.lang.Thread.run(Thread.java:834)Caused by:
java.util.concurrent.ExecutionException:
oracle.pgx.common.auth.AuthorizationException: PgxUser(FCCMDSADMIN) does
not own session 6007f00a-8305-4576-9a56-9fa0f061586f or the session does
not exist code: PGX-ERROR-CQAZPV67UM4H at java.base/
java.util.concurrent.CompletableFuture.reportGet(CompletableFuture.java:
395) at java.base/
java.util.concurrent.CompletableFuture.get(CompletableFuture.java:1999)
at oracle.pgx.api.PgxFuture.get(PgxFuture.java:99) at
oracle.pgx.api.ServerInstance.getSession(ServerInstance.java:670) at
oracle.datastudio.interpreter.pgx.CombinedPgxDriver.getOrCreateSession(C
ombinedPgxDriver.java:145) ... 17 moreCaused by:
oracle.pgx.common.auth.AuthorizationException: PgxUser(FCCMDSADMIN) does
not own session 6007f00a-8305-4576-9a56-9fa0f061586f or the session does
not exist code: PGX-ERROR-CQAZPV67UM4H at
oracle.pgx.common.marshalers.ExceptionMarshaler.toUnserializedException(
ExceptionMarshaler.java:107) at
oracle.pgx.common.marshalers.ExceptionMarshaler.unmarshal(ExceptionMarsh
aler.java:123) at
oracle.pgx.client.RemoteUtils.parseExceptionalResponse(RemoteUtils.java:
130) at
oracle.pgx.client.HttpRequestExecutor.executeRequest(HttpRequestExecutor
.java:198) at
```

```
oracle.pgx.client.HttpRequestExecutor.get(HttpRequestExecutor.java:165)
at
oracle.pgx.client.RemoteControlImpl$10.request(RemoteControlImpl.java:31
3) at
oracle.pgx.client.RemoteControlImpl$ControlRequest.request(RemoteControl
Impl.java:119) at
oracle.pgx.client.RemoteControlImpl$ControlRequest.request(RemoteControl
Impl.java:110) at
oracle.pgx.client.AbstractAsyncRequest.execute(AbstractAsyncRequest.java
:47) at
oracle.pgx.client.RemoteControlImpl.request(RemoteControlImpl.java:107)
at
oracle.pgx.client.RemoteControlImpl.getSessionInfo(RemoteControlImpl.jav
a:296) at
oracle.pgx.api.ServerInstance.lambda$getSessionInfoAsync$14(ServerInstan
ce.java:490) at java.base/
java.util.concurrent.CompletableFuture.uniComposeStage(CompletableFuture
.java:1106) at java.base/
java.util.concurrent.CompletableFuture.thenCompose(CompletableFuture.jav
a:2235) at oracle.pgx.api.PgxFuture.thenCompose(PgxFuture.java:158)
```

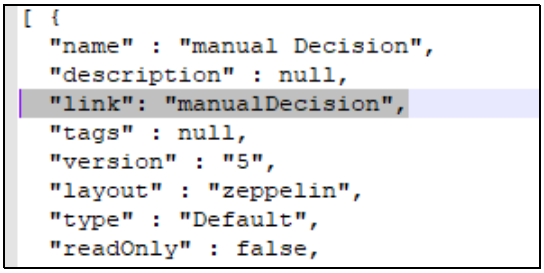
You can perform the following steps as a workaround -

d. Export the "Manual Decision" Notebook

e. Add the link parameter just below Description

For example, "link": "manual Decision",

Figure 15: link parameter



f. Truncate the table “fcc_er_paragraph_manual" in Studio Schema.

g. Import the modified notebook again.

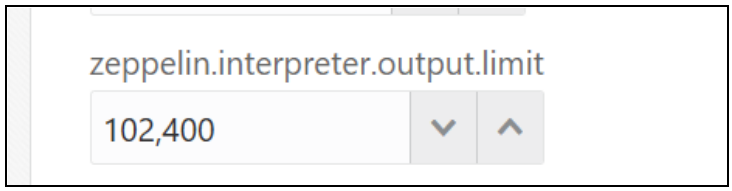
9. What should I do when the result set is truncated if the size goes above '102400' bytes?

a. Perform the following steps:

b. Login to Compliance Studio.

c. Navigate to interpreter zeppelin.interpreter.output.limit.

Figure 16: Interpreter zeppelin parameter



d. Set the value to the required size.

e. Restart the Studio Application.

10. What should I do when the spark interpreter is not working?

a. Log in to the server where Compliance Studio is installed.

b. Navigate to $SPARK_HOME directory. If the path is not set, then navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/interpreter-server/spark-interpreter-<version>/extralibs` directory.

c. Export the following environment variables:

```
export HADOOP_CONF_DIR=<HADOOP Configuration Directory path>

export SPARK_HOME=<SPARK CLIENT DIRECTORY path>

export SPARK_CONF_DIR=<spark-defaults.conf directory path >

export SPARK_SUBMIT_OPTS="-Djava.security.krb5.conf=<kerberos
directory path>/krb5.conf"
```

d. Run the following commands for specific cases:

— The result of the following command should be Pie value. (It ensures that the client is configured successfully.

```
./bin/run-example --master yarn SparkPi 10
```

— The result of the following command is displayed as a Pie value. (It ensures that the client can successfully connect to the remote cluster

```
./bin/spark-submit  --class org.apache.spark.examples.SparkPi --
master yarn <SPARK_HOME/examples/jars/>/spark-examples_<Ver-
sion>.jar 10
```

For example, in case of spark 2.11-2.4.0, the command is as follows:

```
./bin/spark-submit  --class org.apache.spark.examples.SparkPi --
master yarn <SPARK_HOME/examples/jars/>/spark-examples_2.11-
2.4.0.jar 10
```

— The result of the following command displays the list of databases that exist in HIVE.

```
./bin/spark-submit --class org.apache.spark.sql.hive.thrifts-
erver.SparkSQLCLIDriver --master yarn -e "Show databases"
```

— The result of the following command ensures that the client can query from the HIVE schema.

```
./bin/spark-submit --class org.apache.spark.sql.hive.thrifts-
erver.SparkSQLCLIDriver --master yarn -e "select * from
<hiveSchema>.<tableName> limit 10"
```

11. How can I increase the memory of entity resolution and matching services?

    For more information on increasing memory of entity resolution and matching services, see the **Appendix - Setting Memory of Entity Resolution and Matching Services** in the OFS Compliance Studio Administration and Configuration Guide.

12. What should I do when a runtime error occurs while executing a paragraph in Compliance Studio?

    When Compliance Studio is just started (restart/upgrade/fresh installation), every interpreter gives a runtime error for the first time. Re-run the paragraph to get a result.

    In addition, a user with admin privileges has to run a dummy notebook with a simple paragraph of all the used interpreters once.

13. What should I do if I encounter an error on the login?

    If you log in to Compliance Studio for the first time, log out and log back in to resolve the error.

14. How can I retain the logs after restarting the Compliance Studio?

    a. Log in to the Compliance Studio.

    b. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory.

    c. Open the `compliance-studio.sh` file and modify the following for service(s) as per your requirement:

    Search with **"$LOGS_FOLDER"** text for each service and add > (Greater than) special character and space before the text as specified below:

    ```
    "$DEPLOY_APP_HOME"/<service name>/bin/<service name> >>
    "$LOGS_FOLDER"/<service name>.log
    ```

    For example, batchservice, entity-resolution

    ```
    function start_services() {
      service=$1
      case $service in
      batchservice)
        export JAVA_OPTS="-Djavax.net.ssl.trustStore=$DEPLOY_APP_HOME/
    mmg-home/mmg-studio/conf/<studio server>
        -Djavax.net.ssl.trustStorePassword=$STUDIO_SERVER_SSL_PASSWORD"
        sh "$DEPLOY_APP_HOME"/batchservice/bin/batchservice >>
    "$LOGS_FOLDER"/batchservice.log 2>&1 &
        unset JAVA_OPTS
        ;;
      entity-resolution)
        export JAVA_OPTS=<JAVA Options>
        export ER_LOG_PATH="$COMPLIANCE_STUDIO_INSTALLATION_PATH/
    deployed"
        export ER_LOG_LEVEL=INFO
        export LD_LIBRARY_PATH="$COMPLIANCE_STUDIO_INSTALLATION_PATH/
    deployed/python-packages/saneVirtualEnv/lib/python<version>/site-
    ```

```
packages/jep:$COMPLIANCE_STUDIO_INSTALLATION_PATH/deployed/python-
packages/saneVirtualEnv/lib/":$LD_LIBRARY_PATH

    export PATH_ORG=$PATH

    export PATH=$DEPLOY_APP_HOME/python-packages/saneVirtualEnv/
bin:$PATH

    export TNS_ADMIN=$TNS_ADMIN_PATH

    export PYTHONPATH_ORG=$PYTHONPATH

    export PYTHONPATH="$DEPLOY_APP_HOME"/python-packages/
saneVirtualEnv/lib/python<version>/site-packages:$PYTHONPATH_ORG

    sh "$DEPLOY_APP_HOME"/entity-resolution/bin/entity-resolution >>
"$LOGS_FOLDER"/entity-resolution.log &

    unset JAVA_OPTS

    export PATH=$PATH_ORG

    ;;
```

d. For load to OpenSearch, you need to add one more > (Greater than) special character as specified below:

```
sh "$DEPLOY_APP_HOME"/load-to-open-search/bin/load-to-open-search

>>"$DEPLOY_APP_HOME"/logs/load-to-open-search.log &
```

e. Restart Compliance Studio. To do this, run the following command:

```
./compliance-studio.sh –restart
```

Or

```
./compliance-studio.sh –r script
```

15. What should I do if the following error message is displayed while starting Compliance Studio services?

```
Java Memory error: unable to create new native thread
```

The user should perform the following steps:

a. Login to the Linux server as a root user where Compliance Studio is installed.

b. Open `/etc/security/limits.conf` file.

c. Add the following parameters in the file:

```
soft nofile 65536

hard nofile 65536

<linux username> soft nproc 10240

@svrtech soft memlock 500000

@svrtech hard memlock 500000
```

d. Save the file.

e. Restart the Compliance Studio.

16. What should I do when unable to refresh Graph and fail due to the following error?

```
Failed to load graph '<Graph name>' in PGX server: http://
<hostname>:7007

08:22:54.878 [se-nio-7059-exec-1] ERROR
s.fccm.graphService.service.GraphExecutorService - Failed to refresh PGX
Graph, <Graph name>, in all PGX servers
```

a. Stop the PGX server.

b. Log in to Studio schema.

c. Delete the entries that are related to the graph in the tables - **fcc_graph_m_config_json** and **fcc_pgx_m_config**

d. Start the PGX server.

a. Re-execute the Batch for the Graph pipeline or Refresh the Graph task. See the **Managing Graph Pipeline** section in the OFS Compliance Studio User Guide.

17. What should I do if there is a below error in the umm-service logs?

```
[29-06-22 07:30:48,095 GMT AM] [INFO ] [WEB]  [UMM]  [NA]
[GETUSERSESSION] Exception occurred while getting x-auth-token in
initSession method of GetUserSession classjavax.net.ssl.SSLKeyException:
Hostname verification failed:
HostnameVerifier=weblogic.security.utils.SSLWLSHostnameVerifier,
hostname=129.80.90.202.
```

Perform the steps provided in the https://docs.oracle.com/middleware/1213/wls/WLACH/taskhelp/security/DisableHostNameVerification.html link.

18. What should I do when upgrading the version JDK 11.0.13 to 11.0.15 using shell script?

To upgrade bundled JDK, perform the following steps:

a. Use the `wget` command to download jdk 11.0.15 from the https://www.oracle.com/java/technologies/javase/jdk11-archive-downloads.html link.

b. Change the directory where mmg-studio is installed and navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/interpreter-server/pgx-interpreter-bundledJRE-<version>`.

c. Run the `./update-jdk.sh [-j JDK11_HOME ] [-o OUTPUT_DIR]` script. `<JDK11_HOME>` specifies the downloaded JDK11 path, and `<OUTPUT_DIR>` specifies where the updated interpreter is saved.

d. Replace the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/interpreter-server/pgx-interpreter-bundledJRE-<version>` directory with the `<OUTPUT_DIR>/pgxjava`.

19. What should I do when unable to update the SSO token to the latest value while reinstalling the Compliance Studio?

The user should perform the following steps:

a. Log in to Studio schema.

b. Edit the table NEXTGENEMF_CONFIG and change the SSO token to the proper value.

c. Commit the changes.

d. Restart the Compliance Studio.

20. What should I do If it is a time-out issue observed in the Graph Pipeline?

    The user should perform the following steps:

    a. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-pipeline/pipeline/data-pipeline-service-<version>/conf/application.properties` directory.

    b. Change the value from 1200000 to 120000000 in the `server.jetty.connection-idle-timeout=` property file.

    c. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-pipeline/pipeline/pipeline-service-<version>/conf/application.properties` directory

    d. Change the value from 1200000 to 120000000 in the `server.jetty.connection-idle-timeout=` property file.

21. What should I do if there is a below error in the Graph Pipeline?

```
08/Aug/2022 10:21:26,761- [LoggerConnection] LoggerConnection: Trying to fetch connection for log.

 08/Aug/2022 10:21:26,761- [LoggerConnection] LoggerConnection: isJNDI value retrieved is true

 08/Aug/2022 10:21:26,769- [LoggerConnection] LoggerConnection: Trying to fetch connection for log.

 08/Aug/2022 10:21:26,769- [LoggerConnection] LoggerConnection: isJNDI value retrieved is true

 08/Aug/2022 10:21:26,760- [DatabaseLogger] ExecutionLogger: Exception while executing queries

 java.lang.Exception:

 at com.oracle.fccm.amlxe.dataPipelineService.sequencer.impl.SequencerDAOImpl.getQueries(SequencerDAOImpl.java:152) ~[classes!/:?]

 at com.oracle.fccm.amlxe.dataPipelineService.sequencer.impl.SequencerDAOImpl$$FastClassBySpringCGLIB$$7e36e608.invoke(<generated>) ~[classes!/:?]

 at org.springframework.cglib.proxy.MethodProxy.invoke(MethodProxy.java:218) ~[spring-core-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at org.springframework.aop.framework.CglibAopProxy$CglibMethodInvocation.invokeJoinpoint(CglibAopProxy.java:771) ~[spring-aop-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:163) ~[spring-aop-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at org.springframework.aop.framework.CglibAopProxy$CglibMethodInvocation.proceed(CglibAopProxy.java:749) ~[spring-aop-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]
```

```
 at
org.springframework.dao.support.PersistenceExceptionTranslationIntercept
or.invoke(PersistenceExceptionTranslationInterceptor.java:139) ~[spring-
tx-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(Ref
lectiveMethodInvocation.java:186) ~[spring-aop-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
org.springframework.aop.framework.CglibAopProxy$CglibMethodInvocation.pr
oceed(CglibAopProxy.java:749) ~[spring-aop-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
org.springframework.aop.framework.CglibAopProxy$DynamicAdvisedIntercepto
r.intercept(CglibAopProxy.java:691) ~[spring-aop-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
com.oracle.fccm.amlxe.dataPipelineService.sequencer.impl.SequencerDAOImp
l$$EnhancerBySpringCGLIB$$c38b7c42.getQueries(<generated>) ~[classes!/
:?]

 at
com.oracle.fccm.amlxe.dataPipelineService.impl.ExecutorDAOImpl.executePi
peline(ExecutorDAOImpl.java:247) ~[classes!/:?]

 at
com.oracle.fccm.amlxe.dataPipelineService.impl.ExecutorDAOImpl$$FastClas
sBySpringCGLIB$$14f27fdb.invoke(<generated>) ~[classes!/:?]

 at
org.springframework.cglib.proxy.MethodProxy.invoke(MethodProxy.java:218)
~[spring-core-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.springframework.aop.framework.CglibAopProxy$CglibMethodInvocation.in
vokeJoinpoint(CglibAopProxy.java:771) ~[spring-aop-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(Ref
lectiveMethodInvocation.java:163) ~[spring-aop-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
org.springframework.aop.framework.CglibAopProxy$CglibMethodInvocation.pr
oceed(CglibAopProxy.java:749) ~[spring-aop-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
org.springframework.dao.support.PersistenceExceptionTranslationIntercept
or.invoke(PersistenceExceptionTranslationInterceptor.java:139) ~[spring-
tx-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(Ref
```

```
lectiveMethodInvocation.java:186) ~[spring-aop-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
org.springframework.aop.framework.CglibAopProxy$CglibMethodInvocation.pr
oceed(CglibAopProxy.java:749) ~[spring-aop-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
org.springframework.aop.framework.CglibAopProxy$DynamicAdvisedIntercepto
r.intercept(CglibAopProxy.java:691) ~[spring-aop-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
com.oracle.fccm.amlxe.dataPipelineService.impl.ExecutorDAOImpl$$Enhancer
BySpringCGLIB$$3277859b.executePipeline(<generated>) ~[classes!/:?]

 at
com.oracle.fccm.amlxe.dataPipelineService.services.ExecutorService.execu
tePipeline(ExecutorService.java:154) ~[classes!/:?]

 at sun.reflect.GeneratedMethodAccessor112.invoke(Unknown Source) ~[?:?]

 at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessor
Impl.java:43) ~[?:1.8.0_321]

 at java.lang.reflect.Method.invoke(Method.java:498) ~[?:1.8.0_321]

 at
org.springframework.web.method.support.InvocableHandlerMethod.doInvoke(I
nvocableHandlerMethod.java:190) ~[spring-web-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
org.springframework.web.method.support.InvocableHandlerMethod.invokeForR
equest(InvocableHandlerMethod.java:138) ~[spring-web-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
org.springframework.web.servlet.mvc.method.annotation.ServletInvocableHa
ndlerMethod.invokeAndHandle(ServletInvocableHandlerMethod.java:105)
~[spring-webmvc-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.springframework.web.servlet.mvc.method.annotation.RequestMappingHand
lerAdapter.invokeHandlerMethod(RequestMappingHandlerAdapter.java:879)
~[spring-webmvc-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.springframework.web.servlet.mvc.method.annotation.RequestMappingHand
lerAdapter.handleInternal(RequestMappingHandlerAdapter.java:793)
~[spring-webmvc-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.springframework.web.servlet.mvc.method.AbstractHandlerMethodAdapter.
handle(AbstractHandlerMethodAdapter.java:87) ~[spring-webmvc-
5.2.5.RELEASE.jar!/:5.2.5.RELEASE]
```

```
 at
org.springframework.web.servlet.DispatcherServlet.doDispatch(DispatcherS
ervlet.java:1040) ~[spring-webmvc-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.springframework.web.servlet.DispatcherServlet.doService(DispatcherSe
rvlet.java:943) ~[spring-webmvc-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.springframework.web.servlet.FrameworkServlet.processRequest(Framewor
kServlet.java:1006) ~[spring-webmvc-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.springframework.web.servlet.FrameworkServlet.doPost(FrameworkServlet
.java:909) ~[spring-webmvc-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at javax.servlet.http.HttpServlet.service(HttpServlet.java:652)
~[tomcat-embed-core-9.0.37.jar!/:4.0.FR]

 at
org.springframework.web.servlet.FrameworkServlet.service(FrameworkServle
t.java:883) ~[spring-webmvc-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at javax.servlet.http.HttpServlet.service(HttpServlet.java:733)
~[tomcat-embed-core-9.0.37.jar!/:4.0.FR]

 at
org.eclipse.jetty.servlet.ServletHolder.handle(ServletHolder.java:755)
~[jetty-servlet-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.servlet.ServletHandler$CachedChain.doFilter(ServletHan
dler.java:1617) ~[jetty-servlet-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.springframework.web.filter.RequestContextFilter.doFilterInternal(Req
uestContextFilter.java:100) ~[spring-web-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
org.springframework.web.filter.OncePerRequestFilter.doFilter(OncePerRequ
estFilter.java:119) ~[spring-web-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.eclipse.jetty.servlet.ServletHandler$CachedChain.doFilter(ServletHan
dler.java:1604) ~[jetty-servlet-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.springframework.web.filter.FormContentFilter.doFilterInternal(FormCo
ntentFilter.java:93) ~[spring-web-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.springframework.web.filter.OncePerRequestFilter.doFilter(OncePerRequ
estFilter.java:119) ~[spring-web-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.eclipse.jetty.servlet.ServletHandler$CachedChain.doFilter(ServletHan
dler.java:1604) ~[jetty-servlet-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.springframework.boot.actuate.metrics.web.servlet.WebMvcMetricsFilter
```

```
.doFilterInternal(WebMvcMetricsFilter.java:109) ~[spring-boot-actuator-
2.2.6.RELEASE.jar!/:2.2.6.RELEASE]

 at
org.springframework.web.filter.OncePerRequestFilter.doFilter(OncePerRequ
estFilter.java:119) ~[spring-web-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.eclipse.jetty.servlet.ServletHandler$CachedChain.doFilter(ServletHan
dler.java:1604) ~[jetty-servlet-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.springframework.web.filter.CharacterEncodingFilter.doFilterInternal(
CharacterEncodingFilter.java:201) ~[spring-web-5.2.5.RELEASE.jar!/
:5.2.5.RELEASE]

 at
org.springframework.web.filter.OncePerRequestFilter.doFilter(OncePerRequ
estFilter.java:119) ~[spring-web-5.2.5.RELEASE.jar!/:5.2.5.RELEASE]

 at
org.eclipse.jetty.servlet.ServletHandler$CachedChain.doFilter(ServletHan
dler.java:1604) ~[jetty-servlet-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.servlet.ServletHandler.doHandle(ServletHandler.java:54
5) ~[jetty-servlet-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.handler.ScopedHandler.handle(ScopedHandler.java
:143) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.security.SecurityHandler.handle(SecurityHandler.java:5
90) ~[jetty-security-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.handler.HandlerWrapper.handle(HandlerWrapper.ja
va:127) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.handler.ScopedHandler.nextHandle(ScopedHandler.
java:235) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.session.SessionHandler.doHandle(SessionHandler.
java:1607) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.handler.ScopedHandler.nextHandle(ScopedHandler.
java:233) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.handler.ContextHandler.doHandle(ContextHandler.
java:1297) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.handler.ScopedHandler.nextScope(ScopedHandler.j
ava:188) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]
```

```
 at
org.eclipse.jetty.servlet.ServletHandler.doScope(ServletHandler.java:485
) ~[jetty-servlet-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.session.SessionHandler.doScope(SessionHandler.j
ava:1577) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.handler.ScopedHandler.nextScope(ScopedHandler.j
ava:186) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.handler.ContextHandler.doScope(ContextHandler.j
ava:1212) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.handler.ScopedHandler.handle(ScopedHandler.java
:141) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.handler.HandlerWrapper.handle(HandlerWrapper.ja
va:127) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at org.eclipse.jetty.server.Server.handle(Server.java:500) ~[jetty-
server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.HttpChannel.lambda$handle$1(HttpChannel.java:38
3) ~[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at org.eclipse.jetty.server.HttpChannel.dispatch(HttpChannel.java:547)
[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at org.eclipse.jetty.server.HttpChannel.handle(HttpChannel.java:375)
[jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.server.HttpConnection.onFillable(HttpConnection.java:2
70) [jetty-server-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.io.AbstractConnection$ReadCallback.succeeded(AbstractC
onnection.java:311) [jetty-io-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at org.eclipse.jetty.io.FillInterest.fillable(FillInterest.java:103)
[jetty-io-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.io.ssl.SslConnection$DecryptedEndPoint.onFillable(SslC
onnection.java:543) [jetty-io-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.io.ssl.SslConnection.onFillable(SslConnection.java:398
) [jetty-io-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.io.ssl.SslConnection$2.succeeded(SslConnection.java:16
1) [jetty-io-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at org.eclipse.jetty.io.FillInterest.fillable(FillInterest.java:103)
[jetty-io-9.4.26.v20200117.jar!/:9.4.26.v20200117]
```

```
 at org.eclipse.jetty.io.ChannelEndPoint$2.run(ChannelEndPoint.java:117)
[jetty-io-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.util.thread.strategy.EatWhatYouKill.runTask(EatWhatYou
Kill.java:336) [jetty-util-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.util.thread.strategy.EatWhatYouKill.doProduce(EatWhatY
ouKill.java:313) [jetty-util-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.util.thread.strategy.EatWhatYouKill.tryProduce(EatWhat
YouKill.java:171) [jetty-util-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.util.thread.strategy.EatWhatYouKill.run(EatWhatYouKill
.java:129) [jetty-util-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.util.thread.ReservedThreadExecutor$ReservedThread.run(
ReservedThreadExecutor.java:388) [jetty-util-9.4.26.v20200117.jar!/
:9.4.26.v20200117]

 at
org.eclipse.jetty.util.thread.QueuedThreadPool.runJob(QueuedThreadPool.j
ava:806) [jetty-util-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at
org.eclipse.jetty.util.thread.QueuedThreadPool$Runner.run(QueuedThreadPo
ol.java:938) [jetty-util-9.4.26.v20200117.jar!/:9.4.26.v20200117]

 at java.lang.Thread.run(Thread.java:750) [?:1.8.0_321]

 08/Aug/2022 10:21:26,786- [LoggerConnection] LoggerConnection: Trying
to fetch connection for log.

 08/Aug/2022 10:21:26,786- [LoggerConnection] LoggerConnection: isJNDI
value retrieved is true
```

Re-execute the failed graph pipeline from the scheduler service. To execute the Graph pipeline, see the **Using Scheduler Service** section in the OFS Compliance Studio User Guide.

22. What should I do if there is a below error while executing the ER job 2 - `./ER_Run_Bulk_Similarity_Job.sh` in the matching-service.log?

```
ERROR ss.fccm.matchingservice.service.BulkQueryService - Exception
occurred in bulk processingERROR
ss.fccm.matchingservice.service.BulkQueryService - Exception occurred in
bulk processingjava.lang.IndexOutOfBoundsException: Index 1 out of
bounds for length 1 at
jdk.internal.util.Preconditions.outOfBounds(Preconditions.java:64)
~[?:?] at
jdk.internal.util.Preconditions.outOfBoundsCheckIndex(Preconditions.java
:70) ~[?:?] at
jdk.internal.util.Preconditions.checkIndex(Preconditions.java:248)
~[?:?] at java.util.Objects.checkIndex(Objects.java:372) ~[?:?] at
java.util.ArrayList.get(ArrayList.java:459) ~[?:?] at
com.oracle.ofss.fccm.matchingservice.service.BulkQueryService.preProcess
(BulkQueryService.java:159) [classes!/:?] at
```

```
com.oracle.ofss.fccm.matchingservice.controller.BulkUsingApiController2.
executeAsyncBulkQueryMatch(BulkUsingApiController2.java:76) [classes!/
:?] at jdk.internal.reflect.GeneratedMethodAccessor164.invoke(Unknown
Source) ~[?:?] at
jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMetho
dAccessorImpl.java:43) ~[?:?] at
java.lang.reflect.Method.invoke(Method.java:566) ~[?:?] at
org.springframework.web.method.support.InvocableHandlerMethod.doInvoke(I
nvocableHandlerMethod.java:205) [spring-w
```

This error is displayed only when the OpenSearch index does not have the proper data.

a. Fix the data in the pre tables and cleanup the ER schema.

b. Re-run the job again. To run the job, see **Perform Matching** section in the OFS Compliance Studio Administration and Configuration Guide.

23. What should I do if interpreter settings are changed after restarting the Compliance Studio?

    To retain the interpreter settings, follow these steps:

    a. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-studio/conf` directory.

    b. Open the `application.yml` file and change the value of **overwrite-builtin** to **false** in the interpreter parameter .

    > **NOTE** While upgrading Compliance Studio, you should change the value to **true**.

    c. Restart Compliance Studio.

24. How to upgrade the python virtual environment for the fcc-python interpreter?

    To upgrade, follow these steps:

    a. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory.

    b. Open the `compliance-studio.sh` file and modify the **PYTHONPATH** variable in the default fcc-python interpreter as per your requirement:

    For example,

    ```
    export PYTHONPATH=<absolute/path/to/virtual-environment-or-python-
    installation-folder/lib/python<version>>/site-
    packages:$PYTHONPATH_ORG
    ```

25. What should I do if ER Bulk similarity job fails due to metadata indices?

    To load the indices, follow these steps:

    a. Execute the following cleanup scripts:

    — `ER_Run_Bulk_Similarity_Job.sh`

    — `ER_Create_And_Load_Data_Into_Index.sh`

    b. Delete the indices.

    c. Verify that the indices are deleted completely.

    d. Set the **F_IS_RECENTLY_CHANGED** flag to **Y** in the **fcc_idx_m_lookup**, and **fcc_idx_m_-matching_lookup** tables.

    e. Execute `ER_Create_And_Load_Data_Into_Index.sh`.

f. Ensure all the indices are created (Generally, it should create 19).

g. If all the indices are available, then execute the `ER_Run_Bulk_Similarity_Job.sh`.

26. Unable to open UI (Ruleset details, Manual Decisioning and Merge and Split Global Entities) in the Firefox browser?

    The reason could be:

    a. Compliance Studio UI does not open in the Firefox browser if self-signed certificates are used while installation.

27. The UI (Ruleset details, Manual Decisioning and Merge and Split Global Entities) takes more time to load in other browsers?

    The reason could be:

    a. The Compliance Studio UI screens are not cached if self-signed certificates are used and it takes time to load screen every time.

28. What should I do if the workspaces are not displayed and below error is encountered in the server.log?

    ```
    12:02:16.272 [se-nio-7008-exec-2] ERROR
    er.network.base.exception.ExceptionHandlerAdvice - Internal server
    error.

    io.jsonwebtoken.security.SignatureException: JWT signature does not
    match locally computed signature. JWT validity cannot be asserted and
    should not be trusted.
    ```

    To resolve the error, follow these steps:

    a. Generate the public and private keys. For more information, see the Generate the Public and Private Keys section.

    b. Replace the keys in the paths as mentioned in the Generate the Public and Private Keys section.

    c. Generate the SSO (API) token. For more information, see the Generate API token for CS API User section.

    d. Replace token in the `config.sh` file. For more information, see parameter “SSO_TOKEN” in the Table 14.

    e. Stop Compliance Studio.

    f. Reinstall Compliance Studio.

    g. Replace the value of SSO_TOKEN in the nextgenemf_config table in the studio schema.

    h. Start Compliance Studio.

11 Appendix A - Change Port Numbers for the Applicable Services

Change the port number in the applicable files as shown in the following sections. And also, update the respective port numbers in the **install.sh** in `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin`.

> **WARNING** You must re-install and restart Compliance Studio after changing the service(s) with the new port number.

> **NOTE** Only follow this if you want to update the port number of the service(s).

Topics:

11.1 Server

To change the port number for the server, go to the **application.yml** file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-home/mmg-studio/conf/` directory and edit the following values with the new port, for example, 7008:

- `authserviceUrl: "http://<hostname>:<port>/authservice"`
- `metaserviceUrl: "http://<hostname>:<port>/metaservice"`
- `erserviceUrl: "http://<hostname>:<port>"`
- `batchserviceUrl: "https://<hostname>:<port>/batchservice"`
- `mmgServiceUrl: "https://<hostname>:<port>/cs"`

11.2 Batchservice and Metaservice

To change the port number for the Batchserviceserver, go to the `server-config.properties` file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/batchservice/conf/` directory and edit the following values with the new port.

- `server.http.port:7043`
- `server.shutdownPort:7044`

Follow this step to make the same changes to the Metaservice server.

11.3 Interpreter Service

To change the port number for the Interpreter service, follow these steps:

1. Navigate to the `install.sh` file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin` directory and edit the port number of the required service.
2. Reinstall and restart the service.

11.4 PGX Service

To change the port number for the PGX service, go to the `server.conf` file in the `<PGX installation Path>/pgx-server/conf/` directory and update the new port number as **7007**.

11.5 Graph Service

To change the port number for the Graph service, go to the `application.yml` file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/mmg-home/mmg-load-to-graph/graph-service/conf` directory and update the new port number. By default, it is set as **7059**.

11.6 Matching Service

To change the port number for the matching service, go to the `application.yml` file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/matching-service/conf` directory and update the new port number as **7049**.

11.7 Entity Resolution Service

To change the port number for the entity resolution service, go to the `application.yml` file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/entity-resolution/conf` directory and update the new port number **7051**.

12 Appendix B – Spark or PySpark Interpreter

This section provides additional details for Spark or PySpark Interpreter.

Topics:

To set up an additional Spark or PySpark interpreter, for example, to connect to two different external clusters at the same time, follow these steps:

1. Create a start-script for the second Spark interpreter.

   NOTE This is an optional step.

   a. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/interpreters/bin` directory and create a new start-script called `start-spark2-interpreter.sh` using the following command:

      ```
      cp start-spark-interpreter.sh start-spark2-interpreter.sh
      ```

   b. Edit the start-spark2-interpreter.sh file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/interpreters/bin/` directory to update:

      i. Port number to a new port number that is not in use (for example, 7030)

      ii. Rename the log file, search for the text, .log and give a new name to the log (for example, from spark.log to spark2.log).

   c. Edit the `start-all-interpreters.sh` file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/interpreters/bin/` directory as follows:

      i. Search for the `text sh "$DEPLOY_APP_HOME"/interpreters/bin/start-spark-interpreter.sh &`

      ii. Add an additional entry with `sh "$DEPLOY_APP_HOME"/interpreters/bin/start-spark2-interpreter.sh &`

   NOTE For the **2nd Spark** interpreter variant, use `start-spark2-interpreter.sh`, when configuring for a 3rd variant, use as `start-spark3-interpreter.sh` etc.

2. Create the interpreter JSON for the additional Spark interpreter.

   a. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/interpreters/conf` directory and create the new interpreter JSON called `spark2.json` using the following command:

      ```
      cp spark.json spark2.json
      ```

   b. Edit the `spark2.json` file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/interpreters/conf/` directory as follows:

      i. Update the following parameter values:

      ```
      group: <new-spark-interpreter-name>,

      name: <new-spark-interpreter-name>,

      groupSettings.initialCodeCapability: <new-spark-interpreter-name>,
      ```

```
      port: 7030 (the port chosen in the step 1),
      capabilities.name: <new-spark-interpreter-name>,
      capabilities.button.label: <new-spark-interpreter-name>,
```

3. After the update, the file will look like the following:

```
[
  {
    "group": "spark",
    "name": "spark",
    "className": "org.apache.zeppelin.spark.SparkInterpreter",
    "groupSettings": {
      "initialCode": "1+1",
      "initialCodeCapability": "spark"
    },
    "host": "localhost",
    "port": 7017,
    "capabilities": [
      {
        "name": "spark",
        "highlightLanguage": "scala",
        "formEscapeCharacter": "@",
        "button": {
          "defaultCode": "println(\"Hello, world\")",
          "icon": "fa fa-fw fa-building-o",
          "label": "Spark"
        }
      }
    ],
    "defaultInterpreter": true,
    "properties": {
      "spark.executor.memory": {
        "envName": null,
        "propertyName": "spark.executor.memory",
        "defaultValue": "",
        "description": "Executor memory per worker instance. ex) 512m,
32g",
        "type": "string"
```

```
      },
      "args": {
        "envName": null,
        "propertyName": null,
        "defaultValue": "",
        "description": "spark commandline args",
        "type": "textarea"
      },
      "zeppelin.spark.useHiveContext": {
        "envName": "ZEPPELIN_SPARK_USEHIVECONTEXT",
        "propertyName": "zeppelin.spark.useHiveContext",
        "defaultValue": true,
        "description": "Use HiveContext instead of SQLContext if it is
true.",
        "type": "checkbox"
      },
      "spark.app.name": {
        "envName": "SPARK_APP_NAME",
        "propertyName": "spark.app.name",
        "defaultValue": "Zeppelin",
        "description": "The name of spark application.",
        "type": "string"
      },
      "spark.pyspark.python": {
        "envName": null,
        "propertyName": "spark.pyspark.python",
        "defaultValue": "python3",
        "description": "Python command to run pyspark workers with",
        "type": "string"
      },
      "zeppelin.spark.printREPLOutput": {
        "envName": null,
        "propertyName": "zeppelin.spark.printREPLOutput",
        "defaultValue": true,
        "description": "Print REPL output",
        "type": "checkbox"
```

```
      },
      "spark.cores.max": {
        "envName": null,
        "propertyName": "spark.cores.max",
        "defaultValue": "",
        "description": "Total number of cores to use. Empty value uses
all available core.",
        "type": "number"
      },
      "zeppelin.spark.maxResult": {
        "envName": "ZEPPELIN_SPARK_MAXRESULT",
        "propertyName": "zeppelin.spark.maxResult",
        "defaultValue": "1000",
        "description": "Max number of Spark SQL result to display.",
        "type": "number"
      },
      "spark.master": {
        "envName": "MASTER",
        "propertyName": "spark.master",
        "defaultValue": "yarn",
        "description": "Spark master uri. ex) spark://masterhost:7077",
        "type": "string"
      },
      "spark.yarn.archive": {
        "envName": null,
        "propertyName": "spark.yarn.archive",
        "defaultValue": "",
        "description": "An archive containing needed Spark jars for
distribution to the YARN cache",
        "type": "string"
      },
      "spark.driver.bindAddress": {
        "envName": "DRIVER_BIND_ADDRESS",
        "propertyName": "spark.driver.bindAddress",
        "defaultValue": "0.0.0.0",
        "description": "Hostname or IP address where to bind listening
sockets.",
```

```
        "type": "string"
      },
      "zeppelin.spark.enableSupportedVersionCheck": {
        "envName": null,
        "propertyName": "zeppelin.spark.enableSupportedVersionCheck",
        "defaultValue": true,
        "description": "Do not change - developer only setting, not for
production use",
        "type": "checkbox"
      },
      "zeppelin.spark.uiWebUrl": {
        "envName": null,
        "propertyName": "zeppelin.spark.uiWebUrl",
        "defaultValue": "",
        "description": "Override Spark UI default URL",
        "type": "string"
      },
      "zeppelin.spark.useNew": {
        "envName": null,
        "propertyName": "zeppelin.spark.useNew",
        "defaultValue": true,
        "description": "Whether use new spark interpreter
implementation",
        "type": "checkbox"
      },
      "zeppelin.spark.ui.hidden": {
        "envName": null,
        "propertyName": "zeppelin.spark.ui.hidden",
        "defaultValue": false,
        "description": "Whether to hide spark ui in zeppelin ui",
        "type": "checkbox"
      },
  "zeppelin.interpreter.output.limit": {
        "envName": null,
        "propertyName": "zeppelin.interpreter.output.limit",
        "defaultValue": "102400",
```

```
        "description": "Output message from interpreter exceeding the
limit will be truncated",
        "type": "number"
      }
    },
    "initialCode": [],
    "editor": {
      "language": "scala",
      "editOnDblClick": false
    }
  }
]
```

4. Create the interpreter JSON for the second PySpark interpreter.
   a. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/interpreters/conf` directory and create the new interpreter JSON called `pyspark2.json` using the following command:

      ```
      cp pyspark.json pyspark2.json
      ```

   b. Edit the `pyspark2.json` file in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/interpreters/conf/` directory as follows:
      i. Update the following parameter values:

      ```
      group: <new-spark-interpreter-name>,
      name: <new-spark-interpreter-name>,
      groupSettings.initialCodeCapability: <new-spark-interpreter-name>,
      port: 7030 (the port chosen in the step 1),
      capabilities.name: <new-spark-interpreter-name>,
      capabilities.button.label: <new-spark-interpreter-name>,
      ```

5. After the update, the file will look like the following:

```
[
  {
    "group": "spark",
    "name": "pyspark",
    "classNam": "org.apache.zeppelin.spark.PySparkInterpreter",
    "host": "localhost",
    "port": 7017,
    "capabilities": [
      {
        "name": "pyspark",
```

```
        "highlightLanguage": "python",
        "button": {
          "defaultCode": "print('Hello World')",
          "icon": "icon-python",
          "label": "PySpark"
        },
        "formEscapeCharacter": "$"
      }
    ],
    "properties": {
      "zeppelin.pyspark.python": {
        "envName": "PYSPARK_PYTHON",
        "propertyName": null,
        "defaultValue": "python3",
        "description": "Python executable to run pyspark with",
        "type": "string"
      },
      "zeppelin.pyspark.useIPython": {
        "envName": null,
        "propertyName": "zeppelin.pyspark.useIPython",
        "defaultValue": false,
        "description": "whether use IPython when it is available",
        "type": "checkbox"
      },
  "zeppelin.interpreter.output.limit": {
        "envName": null,
        "propertyName": "zeppelin.interpreter.output.limit",
        "defaultValue": "102400",
        "description": "Output message from interpreter exceeding the
limit will be truncated",
        "type": "number"
      }
    },
    "initialCode": []
  }
]
```

> **NOTE** If you try to connect two interpreters to different external clusters when setting the environment variables, `SPARK_HOME` and `HADOOP_CONF_DIR`, as part of providing custom Spark libraries in Yarn Mode, ensure that you append the environment variables to the respective Spark interpreter start-scripts.

6. Restart Compliance Studio. To do this, navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin/` directory and run the `./compliance-studio.sh –restart or ./compliance-studio.sh –r script`

12.1 Spark Interpreter User Impersonation

Configure the Spark cluster and Studio to allow proxy users.

Add the below properties and values in `core-site.xml` in the Spark cluster as well as Studio and restart the Spark cluster and Studio:

```
<property>
  <name>hadoop.proxyuser.zeppelin.groups</name>
  <value>*</value>
</property>
<property>
  <name>hadoop.proxyuser.zeppelin.hosts</name>
  <value>*</value>
</property>
```

Configure the Spark interpreter to run the spark-submit job as the currently logged-in user.

Add the below property in `spark.json`:

```
"zeppelin.spark.run.asLoginUser": {
    "envName": null,
    "propertyName": "zeppelin.spark.run.asLoginUser",
    "defaultValue": true,
    "description": "Whether run spark job as the zeppelin login user, it is only applied when running spark job in hadoop yarn cluster and shiro is enabled",
    "type": "checkbox"
}
```

> **NOTE** There will be only a single keytab used by all Spark interpreter runs.

12.2 Sample spark-default.conf Configuration File

Here is the sample code block for creating `spark-default.conf` file:

```
spark.driver.port 30303

spark.blockManager.port 31313

spark.driver.bindAddress 0.0.0.0

spark.yarn.dist.files <COMPLIANCE STUDIO INSTALLTION PATH>/deployed/mmg-home/
mmg-studio/interpreter-server/spark-interpreter-<version>/extralibs/spark-
<version>-bin-hadoop<version>/python/lib/pyspark.zip,<COMPLIANCE STUDIO
INSTALLTION PATH>/deployed/mmg-home/mmg-studio/interpreter-server/spark-
interpreter-<version>/extralibs/spark-<version>-bin-hadoop<version>/python/
lib/py4j-0.10.7-src.zip

spark.executorEnv.PYTHONPATH pyspark.zip:py4j-0.10.7-src.zip

spark.driver.defaultJavaOptions "-Dsun.security.krb5.debug=false -
Djavax.security.auth.useSubjectCredsOnly=false -
Djava.security.krb5.conf=<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/
batchservice/user/conf/krb5.conf"

spark.driver.host <FQDN_HOSTNAME>

spark.yarn.keytab <COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/
batchservice/user/conf/fccstudio.keytab

spark.yarn.principal <KRBS_PRINCIPAL>

spark.yarn.kerberos.relogin.period 1m
```

NOTE

- **FQDN_HOSTNAME** stands for compliance Studio Fully Qualified hostname, and **KRBS_PRINCIPAL** stands for Kerberos principal.
- For example, the Spark version is **spark-2.4.0-bin-hadoop2.7.**

13 Appendix C – Additional Jars – PGX

PGX-Server does not include Hadoop-client for reading graphs from HDFS.

> **NOTE**
> - **This section is deprecated in the current release and will be removed in the future release.**
> - This section can be skipped if the deployer intends to use only ready to use sample-graph or PGX server without ETL.

When deploying Studio, you must obtain the following libraries. These libraries can be obtained from your existing big data cluster or the internet. The following list of jars is for **Hadoop-client 3.0.0-cdh6.3.0**. These libraries are referred to as '**hdfs-libs**'.

> **NOTE** The following Jar files for your reference. you can use the similar **hdfs-libs** jars based on your Big Data cluster.

Table 21 lists required libraries:

Table 21: List of libraries

accessors-smart-1.2.jar	jaxb-api-2.2.11.jar
aopalliance-1.0.jar	jaxb-impl-2.2.3-1.jar
asm-5.0.4.jar	jcip-annotations-1.0-1.jar
avro-1.8.2-cdh6.3.1.jar	jersey-client-1.19.jar
commons-beanutils-1.9.4.jar	jersey-core-1.19.jar
commons-cli-1.2.jar	jersey-guice-1.19.jar
commons-codec-1.11.jar	jersey-json-1.19.jar
commons-collections-3.2.2.jar	jersey-server-1.19.jar
commons-compress-1.18.jar	jersey-servlet-1.19.jar
commons-configuration2-2.1.1.jar	jettison-1.1.jar
commons-io-2.6.jar	jetty-security-9.3.25.v20180904.jar
commons-lang-2.6.jar	jetty-servlet-9.3.25.v20180904.jar
commons-lang3-3.7.jar	jetty-util-9.3.25.v20180904.jar
commons-logging-1.2.jar	jetty-webapp-9.3.25.v20180904.jar
commons-math3-3.1.1.jar	jetty-xml-9.3.25.v20180904.jar
commons-net-3.1.jar	jline-0.9.94.jar
curator-client-2.12.0.jar	json-smart-2.3.jar

Table 21: List of libraries

curator-framework-2.12.0.jar	jsp-api-2.1.jar
curator-recipes-2.12.0.jar	jsr305-3.0.0.jar
gson-2.2.4.jar	jsr311-api-1.1.1.jar
guava-16.0.1.jar	kerb-admin-1.0.0.jar
guice-4.0.jar	kerb-client-1.0.0.jar
hadoop-annotations-3.0.0-cdh6.3.1.jar	kerb-common-1.0.0.jar
hadoop-auth-3.0.0-cdh6.3.1.jar	kerb-core-1.0.0.jar
hadoop-client-3.0.0-cdh6.3.1.jar	kerb-crypto-1.0.0.jar
hadoop-common-3.0.0-cdh6.3.1.jar	kerb-identity-1.0.0.jar
hadoop-hdfs-client-3.0.0-cdh6.3.1.jar	kerb-server-1.0.0.jar
hadoop-mapreduce-client-common-3.0.0-cdh6.3.1.jar	kerb-simplekdc-1.0.0.jar
hadoop-mapreduce-client-core-3.0.0-cdh6.3.1.jar	kerb-util-1.0.0.jar
hadoop-mapreduce-client-jobclient-3.0.0-cdh6.3.1.jar	kerby-asn1-1.0.0.jar
hadoop-yarn-api-3.0.0-cdh6.3.1.jar	kerby-config-1.0.0.jar
hadoop-yarn-client-3.0.0-cdh6.3.1.jar	kerby-pkix-1.0.0.jar
hadoop-yarn-common-3.0.0-cdh6.3.1.jar	kerby-util-1.0.0.jar
htrace-core4-4.1.0-incubating.jar	kerby-xdr-1.0.0.jar
httpclient-4.5.3.jar	log4j-1.2.17.jar
httpcore-4.4.6.jar	netty-3.7.0.Final.jar
jackson-annotations-2.9.9.jar	nimbus-jose-jwt-4.41.1.jar
jackson-core-2.9.9.jar	okhttp-2.7.5.jar
jackson-core-asl-1.9.13.jar	okio-1.6.0.jar
jackson-databind-2.9.9.3.jar	paranamer-2.8.jar
jackson-jaxrs-1.9.2.jar	protobuf-java-2.5.0.jar
jackson-jaxrs-base-2.9.9.jar	re2j-1.1.jar
jackson-jaxrs-json-provider-2.9.9.jar	slf4j-api-1.7.25.jar
jackson-mapper-asl-1.9.13-cloudera.1.jar	slf4j-log4j12-1.7.25.jar
jackson-module-jaxb-annotations-2.9.9.jar	snappy-java-1.1.4.jar
jackson-xc-1.9.2.jar	stax2-api-3.1.4.jar
javax.activation-api-1.2.0.jar	woodstox-core-5.0.3.jar
javax.inject-1.jar	xz-1.6.jar
javax.servlet-api-3.1.0.jar	zookeeper-3.4.8.jar

14 Appendix D – Additional Jars – Batch Service

When deploying Studio, you must obtain the following files for Batch Service.

> **NOTE**
>
> - **This section is deprecated in the current release and will be removed in the future release.**
> - The following Jar files for your reference. you can use the similar **hdfs-libs** jars based on your Big Data cluster.

Table 22 lists the required files:

Table 22: List of Files

accessors-smart-1.2.jar	jersey-server-1.19.jar
activation-1.1.jar	jersey-servlet-1.19.jar
asm-5.0.4.jar	jettison-1.1.jar
avro-1.8.2-cdh6.3.1.jar	jetty-http-9.3.25.v20180904.jar
commons-beanutils-1.9.4.jar	jetty-io-9.3.25.v20180904.jar
commons-cli-1.2.jar	jetty-security-9.3.25.v20180904.jar
commons-codec-1.11.jar	jetty-server-9.3.25.v20180904.jar
commons-collections-3.2.2.jar	jetty-servlet-9.3.25.v20180904.jar
commons-compress-1.18.jar	jetty-util-9.3.25.v20180904.jar
commons-configuration2-2.1.1.jar	jetty-webapp-9.3.25.v20180904.jar
commons-io-2.6.jar	jetty-xml-9.3.25.v20180904.jar
commons-lang-2.6.jar	jline-0.9.94.jar
commons-lang3-3.7.jar	jsch-0.1.54.jar
commons-logging-1.2.jar	json-smart-2.3.jar
commons-math3-3.1.1.jar	jsp-api-2.1.jar
commons-net-3.1.jar	jsr305-3.0.0.jar
curator-client-2.12.0.jar	jsr311-api-1.1.1.jar
curator-framework-2.12.0.jar	kerb-admin-1.0.0.jar
curator-recipes-2.12.0.jar	kerb-client-1.0.0.jar
gson-2.2.4.jar	kerb-common-1.0.0.jar
guava-16.0.1.jar	kerb-core-1.0.0.jar
hadoop-annotations-3.0.0-cdh6.3.1.jar	kerb-crypto-1.0.0.jar
hadoop-auth-3.0.0-cdh6.3.1.jar	kerb-identity-1.0.0.jar
hadoop-common-3.0.0-cdh6.3.1.jar	kerb-server-1.0.0.jar
hive-exec-1.1.0-cdh5.13.0.jar	kerb-simplekdc-1.0.0.jar

Table 22: List of Files

HiveJDBC4.jar	kerb-util-1.0.0.jar
hive-metastore-1.1.0-cdh5.13.0.jar	kerby-asn1-1.0.0.jar
hive-service-1.1.0-cdh5.13.0.jar	kerby-config-1.0.0.jar
htrace-core4-4.1.0-incubating.jar	kerby-pkix-1.0.0.jar
httpclient-4.5.3.jar	kerby-util-1.0.0.jar
httpcore-4.4.6.jar	kerby-xdr-1.0.0.jar
jackson-annotations-2.9.0.jar	log4j-1.2.17.jar
jackson-core-2.9.9.jar	netty-3.7.0.Final.jar
jackson-core-asl-1.9.13.jar	nimbus-jose-jwt-4.41.1.jar
jackson-databind-2.9.9.3.jar	paranamer-2.8.jar
jackson-jaxrs-1.9.2.jar	protobuf-java-2.5.0.jar
jackson-mapper-asl-1.9.13-cloudera.1.jar	re2j-1.1.jar
jackson-xc-1.9.2.jar	slf4j-api-1.7.25.jar
javax.activation-api-1.2.0.jar	slf4j-log4j12-1.7.25.jar
javax.servlet-api-3.1.0.jar	snappy-java-1.1.4.jar
jaxb-api-2.2.2.jar	stax2-api-3.1.4.jar
jaxb-impl-2.2.3-1.jar	stax-api-1.0-2.jar
jcip-annotations-1.0-1.jar	woodstox-core-5.0.3.jar
jersey-core-1.19.jar	xz-1.6.jar
jersey-json-1.19.jar	zookeeper-3.4.8.jar

15 Appendix E – Create Users, Groups, and Mappings

This AAI User Provisioning SQL Scripts Generator Utility allows you to use AAI for authN in the Compliance Studio. Identity administrators can create new user groups/roles, perform appropriate roles, usergroup and domain mapping, and so on.

This is provided as a SQL generator utility. This SQL scripts is executed in the AAI's config schema to create the required metadata.

You must execute this script multiple times against each username. Also, generate the merge scripts accordingly.

Execute the following command from `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/<mmg-home>/bin` directory.

```
./userprovisioning-script-generator.sh <user> <comma separated listof user groups or ALL> <infodom> <segment>
```

Sample Commands:

```
./userprovisioning-script-generator.sh SCRIPTUSER ALL OFSAAAIINFO EMFLD ./userprovisioning-script-generator.sh SCRIPTUSER MDLREV,MDLUSR,IDENTITY_ADMIN OFSAAAIINFO EMFLD
```

Table 23 list the pre-configured groups in the Compliance Studio.

Table 23: Pre-configured Group

User Group Name	User Group Description
IDNTYADMN	Identity Administrator group
IDNTYAUTH	Identity Authorizer group
MDLREV	The Modeling Reviewer Group. Users mapped to this group have access to the menu items in the application that are related to model review activities.
MDLAPPR	The Modeling Approver Group. Users mapped to this group have the rights to approve models created by the users.
MDLBATCHUSR	The Modeling Batch User. Scheduler can use this Group for executing batches.
WKSPADMIN	The Workspace Administrator Group. Users mapped to this group have access to create and populate workspaces. For viewing the landing page this group is required.
MDLUSR	The Modeling User Group. Users mapped to this group have access to all the menu items in the application that is related to model creation.
DSUSRGRP	Data Studio User Group This User Group provide access to modify Interpreter configurations.

Table 23: Pre-configured Group

User Group Name	User Group Description
GRPADMIN	The Graph Administrator Group Users mapped to this group have access to all the menu items in the application related to graph as well as Pipeline/Refresh graphs related health services.
GRPUSR	The Graph User Group Users mapped to this group have access to all the menu items in the application related to graph as well as Pipeline/Refresh graphs related health services.
DSREDACTGRP	Roles for applying redaction in graph. This group will be applicable to only those users for whom graph redaction is required. **NOTE:** This group has to be created manually in AAI and map it to the users.
ERADMIN	Entity resolution admin group. **NOTE:** This group has to be created manually in AAI and map it to the users.
ERUSER	Entity resolution user group. **NOTE:** This group has to be created manually in AAI and map it to the users.

16 Appendix F - Generate an Encrypted Password for OPenSearch

To generate encrypted passwords required during configuration, i.e., while configuring encrypted passwords. For example, OPEN_SEARCH_ENCRYPTED_PASSWORD.

To generate an encrypted password, follow these steps:

1. Set the export `FIC_DB_HOME` path in the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/ficdb` directory.
2. Run the echo `$FIC_DB_HOME` command.
3. Navigate to `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/ficdb/bin` directory and run the `./FCCM_Studio_Base64Encoder.sh <password to be encrypted>` command.

17 Appendix G - Disable Initialization in fcc-python-sane Interpreter

To disable the fcc-python-sane interpreter, follow these steps:

1. Navigate to the following directories and update the **MMG_PYTHON_INTERPRETER** property as MMG_PYTHON_INTERPRETER=fcc-python,fcc-python-ml4aml
   - `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/bin/install.sh`
   - `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/bin/config.sh`
   - `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-service/bin/config.sh`
2. Navigate to the `<COMPLIANCE_STUDIO_INSTALLATION_PATH>/deployed/mmg-home/mmg-service/conf/application.properties` directory.
3. Update the **mmg.python_interpreter** property as mmg.python_interpreter=fcc-python,fcc-python-ml4aml.
4. Restart Compliance Studio.

OFSAA Support

Raise a Service Request (SR) in My Oracle Support (MOS) for queries related to OFSAA applications.

Send Us Your Comments

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, indicate the title and part number of the documentation along with the chapter/section/page number (if available) and contact the Oracle Support.

Before sending us your comments, you might like to ensure that you have the latest version of the document wherein any of your concerns have already been addressed. You can access My Oracle Support site which has all the revised/recently released documents.