Oracle Financial Services Crime and Compliance Studio

Installation and Configuration Guide

Release 8.0.7.4.0

April 2020

E91246-01

ORACLE
Financial Services

OFS Crime and Compliance Studio Installation and Configuration Guide

Document Control

Version Number	Revision Date	Changes Done
8.0.7.4.0	Updated: April 2020	• FCC Studio can use either the BD or ECM schema as the source of FCDM data for the graph. A new parameter, FCDM_Source added to the config.sh file. For more information, see Configuring the config.sh File. • IDCS Realm Integration with FCC Studio allows SSO for both FCC Studio and ECM to provide seamless integration and eliminates the requirement to log in separately to FCC Studio. For more information, see Configuring the config.sh File. • FCC Studio supports the installation of any version of Python and also FCC Studio Installer is packaged with python 3.6 libraries. For more information, see Configuring the fcc-python Interpreter. • ES_Hadoop jar files must be placed in all nodes of the Spark cluster. For more information, see Configuring the Data Movement and Graph Load. • Elastic search component has been enhanced to prevent the OutOfMemory error. For more information, see Appendix - Configuring the Elastic Search Component.
8.0.7.3.0	Updated: March 2020	• A new component called the Entity Resolution is introduced that enables Entity linking in graphs and Entity Searching in notebooks based on multi-attribute name matching using the Elastic Search service. For more information, see Configuring the Elastic Search Component and Appendix - Configuring the Elastic Search Component. • A new script is introduced for the preparation of ICIJ to resolve data quality issues before graph loading. For more information, see Cleaning the ICIJ Data.
8.0.7.2.0	Updated: February 2020	• Install FCC Studio with or without OFSAA. For more information, see Installing FCC Studio and Installing FCC Studio With Non-OFSAA. • The support for the Data Forward service is deprecated. • Upgrade an existing instance of FCC Studio (v8.0.7.x) to v8.0.7.2.0 version. For more information, see Upgrading FCC Studio. • Configure newly introduced interpreters such as Spark and PySpark interpreters. For more information, see Interpreter Settings.

Version Number	Revision Date	Changes Done
8.0.7.1.0	Updated: October 2019	• Updated the guide for the following newly introduced FCC Studio services: ▪ ETL ▪ Data Forwarding ▪ Cross Language Name Matching
8.0.7.0.0	Created: November 2019	Created the first version of the Oracle Financial Services Crime and Compliance Studio Installation Guide for v8.0.7.0.0 Release.

Table of Contents

1 Preface

This section provides supporting information for the Oracle Financial Services (OFS) Crime and Compliance Studio (FCC Studio) Application Deployment Guide and includes the following topics:

Topics:

1.1 Summary

Before you begin the installation, ensure that you have access to the Oracle Support Portal with valid login credentials to quickly notify us of any issues at any stage. You can obtain the login credentials by contacting the Oracle Support.

1.2 Audience

This document is intended for System Engineers who are responsible for installing and configuring or upgrading FCC Studio.

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

- UNIX commands
- Database concepts
- Big Data

1.3 Related Documents

You can access the following additional documents related to the OFS Crime and Compliance Studio application from the *Oracle Help Center (OHC)* Documentation Library:

- *Oracle Financial Services Crime and Compliance Studio Deployment Guide (Using Kubernetes)*
- *Oracle Financial Services Crime and Compliance Studio Administration Guide*
- *Oracle Financial Services Crime and Compliance Studio User Guide*
- *Oracle Financial Services Crime and Compliance Studio Data Model Guides*
- *Oracle Financial Services Crime and Compliance Studio Release Notes and Readme*

1.4 Abbreviations

The following table lists the abbreviations used in this document.

Table 1: Abbreviations

Abbreviation	Meaning
OFS	Oracle Financial Services

Table 1: Abbreviations

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

2 Installation Overview

This chapter provides the information required to understand the installation of the Oracle Financial Services (OFS) Crime and Compliance Studio (FCC Studio) application.

This release (v8.0.7.4.0) of FCC Studio can be used for the following:

- To install a new instance of FCC Studio as follows:
  - FCC Studio with OFSAA (Oracle Financial Services Analytical Application). Here, with OFSAA is with BD (Behavior Detection) or ECM (Enterprise Case Management).
  - FCC Studio without OFSAA
- To upgrade an existing instance of FCC Studio as follows:
  - Upgrade FCC Studio from v8.0.7.x onwards to v8.0.7.4.0 with OFSAA.
  - Upgrade FCC Studio from v8.0.7.2.0 onwards to v8.0.7.4.0 without OFSAA.

2.1 Quick Start Steps to Install Studio with OFSAA

This section provides the quick start steps to install a new instance of FCC Studio with OFSAA.

Table 1: Quick Start Steps to Install Studio with OFSAA

Sl. No.	Steps	Reference Links
1.	Prepare for installation	1. Prerequisites 2. Hardware and Software Requirements 3. Prerequisite Environmental Settings 4. Performing Pre-installation Tasks a. Obtaining the Software b. Extracting the Software 5. Required File Structure 6. Interpreter Settings
2.	Install FCC Studio with OFSAA	1. Configuring the Elastic Search Component 2. Enabling Synonym/Stopword with the Elastic Search Service 3. Placing the Required Files 4. Configuring the config.sh File 5. Running the FCC Studio Installer 6. Verifying the FCC Studio Installation 7. Installing PGX Service 8. Accessing the FCC Studio Application

Table 1: Quick Start Steps to Install Studio with OFSAA

Sl. No.	Steps	Reference Links
3	Post-installation configuration	• Configuring the SSH Connection • Configuring the Interpreters • Performing the OFSAA Configuration for Batch Execution • Performing Hive Data Movement • Configuring ICIJ • Configuration for Running Published Notebooks

2.2 Quick Start Steps to Install Studio with Non-OFSAA

This section provides the quick start steps to install a new instance of FCC Studio with Non-OFSAA.

Table 2: Quick Start Steps to Install Studio with Non-OFSAA

Sl. No.	Steps	Reference Links
1.	Prepare for installation	1. Prerequisites 2. Hardware and Software Requirements 3. Prerequisite Environmental Settings 4. Performing Pre-installation Tasks a. Obtaining the Software b. Extracting the Software 5. Required File Structure 6. Interpreter Settings
2.	Install FCC Studio with non-OFSAA	1. Configuring the Elastic Search Component 2. Enabling Synonym/Stopword with the Elastic Search Service 3. Placing the Required Files 4. Configuring the config.sh File 5. Running the FCC Studio Installer 6. Verifying the FCC Studio Installation 7. Installing PGX Service 8. Accessing the FCC Studio Application
3	Post-installation configuration	• Configuring the SSH Connection • Configuring the Interpreters • Configuring ICIJ

3 Preparing for Installation

This chapter provides information about the tasks that must be performed before installing FCC Studio.

Topics:

3.1 Prerequisites

To install FCC Studio with OFSAA, ensure the BD (Behavior Detection) or the ECM (Enterprise Case Management) application pack is installed.

3.2 Hardware and Software Requirements

The following hardware and software are required to install FCC Studio.

Table 1: Hardware and Software Requirements

Hardware/Software Category	Component Version
Browser	• Chrome 57.x • Firefox 52.x
Java Version	Java 8
Processing Server	• RHEL 7.4+ • SFTP • Oracle JRE Standard Edition 1.8.x(with JCE)
Database Server	• Oracle Database Server 12c Release 2 (12.2.0.1+) Enterprise Edition • Oracle R Enterprise 1.5.1 with Oracle R Distribution 3.3.0
PGX (Graph) Server	• RHEL 7.4+ • Minimum gcc library v4.8.2
Elastic Search	elasticsearch-7.3.2
Elastic Search Hadoop Jars	ES - Hadoop v7.3.2 You must download the ZIP file from *https://www.elastic.co/downloads/past-releases/elasticsearch-apache-hadoop-7-3-2*
Big Data	

Table 1: Hardware and Software Requirements

Hardware/Software Category	Component Version
Cloudera Distribution Hadoop 5.12	• CDH Version 5.12 • Hadoop-2.5.0+cdh5.3.3+844 • Hive-0.13.1+cdh5.3.3+350 • Sqoop1 V 1.4.4+cdh5.3.3+67 • The `.profile` file must be present with the `SPARK_HOME` and `PYTHON_HOME` parameters already set. • Set `spark2-shell alias` in the `.profile` file as follows: `alias spark2-shell=spark-shell`
Cloudera Hive Connectors	Hive JDBC Connectors V 2.5.15
Hadoop Security Protocol	• Kerberos R release 1.6.1 • Sentry-1.4.0

3.3 Prerequisite Environmental Settings

The following prerequisite environmental settings must be set before beginning the installation of FCC Studio.

Table 2: Prerequisite Environmental Settings

Category	Expected Value
PGX Settings	PGX version 20.0.2 Set the following paths in the environment variables in the `.profile` file: • `PGX_HOME`: Indicates the path of the server where PGX client is installed. For more information, see Installing PGX Service. • `PGX_TMP_DIR`: Indicates the path of the PGX temporary directory. • `SPARK_HOME`: Indicates the path where `SPARK_HOME` is installed by the client.
Java Settings	• `PATH` in the `.profile` file must be set to include the Java Runtime Environment (Java 8) absolute path. **NOTE:** • Ensure the absolute path to `JRE/bin` is set at the beginning of the `PATH` variable. For example: `PATH=/usr/java/jre1.8/bin:$PATH` • Ensure no SYMBOLIC links to Java installation are set in the `PATH` variable.

Table 2: Prerequisite Environmental Settings

Category	Expected Value
PGX Server	• The following packages must be installed or present in the server where PGX service is installed: ▪ `krb5-libs` ▪ `krb5-workstation` ▪ `procps-ng` ▪ `nc` • Execute the following command to install the above-mentioned packages: `yum install -y krb5-libs krb5-workstation procps-ng nc`
Oracle Database Settings	**NOTE:** This setting is required only if the Wallet has to be created on the same server as that of the FCC Studio server. **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/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.
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.
Studio Schema	1. Create a new Oracle Database schema user using the following script: `CREATE USER <Studio Schema User Name> IDENTIFIED BY <Password>;` A new Oracle Database schema is created. 2. Grant the permissions given in the next row. This newly created schema is referred to as Studio Schema.

Table 2: Prerequisite Environmental Settings

Category	Expected Value

Oracle Database Schema Settings	Grant the following permissions to the newly created Oracle Database 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>;`
	`ALTER USER <Studio Schema User> QUOTA 100M ON users;`
	`GRANT create sequence TO <Studio Schema User>;`
	`GRANT create SYNONYM TO <Studio Schema User>;`
	`GRANT create any context TO <BD or ECM Schema User>;`
	`GRANT execute on dbms_rls TO <BD or ECM Schema User>;`
	`GRANT ALL privileges TO <Studio Schema User>;`

Wallet Settings	Set a password store with Oracle Wallet. For more information, see Appendix - Setting Up Password Stores with Oracle Wallet.

3.4 Performing Pre-installation Tasks

The pre-installation tasks to perform before installing FCC Studio are as follows:

1. Obtaining the Software
2. Extracting the Software

3.4.1 Obtaining the Software

To download the FCC Studio application installer software, follow these steps:

1. Login to *My Oracle Support* with a valid Oracle account and search for the Patch ID **31230568** under the *Patches & Updates* tab.
2. Download the FCC Studio installer archive file to the download directory (in Binary Mode) on the setup identified for the FCC Studio installation.

3.4.2 Extracting the Software

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

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

The FCC Studio installer file is extracted and the `OFS_FCCM_STUDIO` directory is obtained and is referred to as `<Studio_Installation_Path>`.

> **NOTE** 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 0755 OFS_FCCM_STUDIO -R
```

3.5 Required File Structure

Obtain the following configuration files from the Cloudera installation setup:

> **NOTE** These files must be kept ready and provided in the following file structure that is used during FCC Studio installation.

Table B–1

File Category	File Names
Hadoop Cluster	• `core-site.xml` • `hadoop-env.sh` • `hdfs-site.xml` • `log4j.properties` • `ssl-client.xml` • `topology.map` • `topology.py`
Kerberos Files	• `krb5.conf` • `ofsaa.keytab` **NOTE:** Ensure to rename your `.keytab` file to `ofsaa.keytab`.
Additional Jars	• `hive-exec-1.1.0-cdh5.13.0.jar` • `HiveJDBC4.jar` • `hive-metastore-1.1.0-cdh5.13.0.jar` • `hive-service-1.1.0-cdh5.13.0.jar` **NOTE:** • The version of the jars is client/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.

Table B–1

File Category	**File Names**
ES-Hadoop Jars	`elasticsearch-spark-20_2.11-7.3.2.jar` To obtain the `elasticsearch-spark-20_2.11-7.3.2.jar` file, follow these steps: 1. Download the ZIP file from *https://www.elastic.co/downloads/past-releases/elasticsearch-apache-hadoop-7-3-2.* 2. Extract the downloaded file. 3. Navigate to the `dist` directory and obtain the `elasticsearch-spark-20_2.11-7.3.2.jar` file.

3.6 Interpreter Settings

To perform the interpreter settings for the interpreters that you need, see Appendix - Pre-installation Interpreter Settings.

4 Installing FCC Studio

To install FCC Studio, follow these steps:

1. Configuring the Elastic Search Component
2. Enabling Synonym/Stopword with the Elastic Search Service
3. Placing the Required Files
4. Configuring the config.sh File
5. Running the FCC Studio Installer
6. Verifying the FCC Studio Installation
7. Installing PGX Service
8. Accessing the FCC Studio Application

4.1 Configuring the Elastic Search Component

Configure the Elastic Search component as per FCC Studio requirements. For more information, see Appendix - Configuring the Elastic Search Component.

4.2 Enabling Synonym/Stopword with the Elastic Search Service

To enable Synonym and Stopword with the Elastic Search service, follow these steps:

1. Navigate to the `<Elastic_Search_Installed_Path>/config` directory.
2. Create a directory named `analysis` using the following command:

   ```
   mkdir analysis
   ```
3. Place the following Stopword and Synonym files in the newly created `analysis` directory:

> **NOTE**
> - User can decide whether to provide any data or not 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.

- `Synonyms.txt`: Contains name synonyms like bob, bobby, and so on.
- `Country.txt`: Contains country synonyms like US, United States, America, and so on.
- `Organisation_suffix.txt`: Contains organization suffices that are used as stopwords.
- `Title.txt`: Contains title stopwords used as the title for name
- `Gender.txt`: Contains gender-related synonyms.
- `Organisation_strip.txt`: Contains organization stopwords.

4.3 Placing the Required Files

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:
   a. Navigate to the `<Studio_Installation_Path>/batchservice/user/lib` directory.
   b. Place the following additional jar files:
      — `hive-exec-1.1.0-cdh5.13.0.jar`
      — `HiveJDBC4.jar`
      — `hive-metastore-1.1.0-cdh5.13.0.jar`
      — `hive-service-1.1.0-cdh5.13.0.jar`

> **NOTE**
> - The version of the jars is client/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.

2. To place the Kerberos files, follow these steps:
   a. Navigate to the `<Studio_Installation_Path>/batchservice/user/conf` directory.
   b. Place the following Kerberos files:
      — `krb5.conf`
      — `ofsaa.keytab`

4.4 Configuring the config.sh File

Configure the `config.sh` file to install FCC Studio. For more information, see Appendix - Configuring the config.sh File.

4.5 Running the FCC Studio Installer

To run the FCC Studio installer, follow these steps:

1. Navigate to the `<Studio_Installation_Path>/bin/` directory.
2. Execute the following command in the console:

   ```
   ./install.sh
   ```

> **NOTE** Execution of the `install.sh` command does not generate any log file.

3. Execute the following command in the console:

   ```
   ./fcc_studio.sh
   ```

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

   After the successful completion of the FCC Studio installation, the script displays a URL that can be used to access the FCC Studio Application. For more information, see Accessing the FCC Studio Application.

4.6 Verifying the FCC Studio Installation

- Verifying FCC Studio Installed with OFSAA
- Verifying FCC Studio Installed with Non-OFSAA

4.6.1 Verifying FCC Studio Installed with OFSAA

To verify the FCC Studio installation with OFSAA, follow these steps:

1. Check the log files in the `<STUDIO_INSTALLATION_PATH>/logs` directory.

   If all the servers are up and running, it indicates that the installation is complete.

   > **NOTE** Any error encountered in the process is displayed with an appropriate error code in the log file. Do not proceed with further installation and contact Oracle Support with relevant log files.

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

4.6.2 Verifying FCC Studio Installed with Non-OFSAA

To verify the FCC Studio installation with non-OFSAA, follow these steps:

- Check the log files in the `<STUDIO_INSTALLATION_PATH>/logs` directory.
  - Ensure the metaservice is up and running.
  - The authservice and batchservice will not be up.
- In the FCC Studio application UI, navigate to the ***Interpreters*** page to ensure all the interpreters are displayed and the JDBC interpreter is working.

  > **NOTE** Any errors encountered in the process is displayed with an appropriate error code in the log file. Do not proceed with further installation and contact Oracle Support with relevant log files.

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

4.7 Installing PGX Service

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

To install PGX service, follow these steps:

1. Navigate to the `<Studio_Installation_Path>/pgx/server/` directory.
2. Perform the following:
   - If PGX service is to be installed on the same server where FCC Studio is installed, unzip the `pgx-distribution-20.0.2-server.zip` file.
   - If PGX service is to be installed on a different server, follow these steps:

i. Copy the `pgx-distribution-20.0.2-server.zip` file to the PGX server.

ii. Unzip the `pgx-distribution-20.0.2-server.zip` file.

> **NOTE** The path where the `pgx-distribution-20.0.2-server.zip` file is unzipped is referred to as `<PGX_Installation_Path>`.

3. Navigate to the `<PGX_Installation_Path>/pgx/server/conf` directory.
4. Configure the following properties as per the requirement:
   - Set the following in the `server.conf` file:

     ```
     enable_tls: false,
     enable_client_authentication: false
     ```

   - Set the following in the `pgx.conf` file:

     ```
     allow_local_filesystem: true
     ```

   Here, true indicates to enable, and false indicates to disable.
5. Place the following Kerberos Files in the `<PGX_Installation_Path>/pgx/server/conf/kerberos` directory:
   - `krb5.conf`
   - `ofsaa.keytab`
6. Place the following Hadoop configuration files in the `<PGX_Installation_Path>/pgx/server/conf/hadoop_cluster` directory:
   - `core-site.xml`
   - `hadoop-env.sh`
   - `hdfs-site.xml`
   - `log4j.properties`
   - `ssl-client.xml`
   - `topology.map`
   - `topology.py`
7. Navigate to the `<PGX_Installation_Path>/pgx/server/bin` and configure the `config.sh` file as described in the following table:

Table G–1 config.sh Parameters

InteractionVariable Name	Significance
KERBEROS_TICKET_RENEWAL_PERIOD	For example: 7200 would mean every 2 hours
KERBEROS_PRINCIPAL	For example: USER@PRINCIPAL
KERBEROS_KEYTAB_FILENAME	For example: ofsaa.keytab

Table G–1 (Continued)config.sh Parameters

InteractionVariable Name	Significance
KRB5_CONFIG_FILENAME	For example: krb5.conf
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 container's memory limit size above. For example: 10240
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 container's memory limit size above. For example: 10240
URL_GLOBAL_GRAPH_CONFIG_JSON	Indicates the URL of the global graph to be pre-loaded. The value can be on HDFS. For example: `hdfs:///user/fccstudio/graph.json`
PGX_GLOBAL_GRAPH_NAME	Indicates the name that the pre-loaded global graph is published with and the FCC Studio users can use to reference the global graph. For example: GlobalGraphIH

8. Navigate to the `<PGX_Installation_Path>/pgx/server/bin` directory and execute the following command:

   ```
   ./install.sh
   ```

9. Start the PGX service. For more information, see Starting/Stopping PGX Service.

NOTE

- You must run at least one successful ETL batch to start the PGX service with the `graph.json` file located in the `URL_GLOBAL_GRAPH_CONFIG_JSON` path is present. For more information, see the Data Movement and Graph Loading for Big Data Environment section in the *OFS Crime and Compliance Studio Administration Guide*.
- Ensure to execute the Graph_Alive notebook after each time you start/restart the PGX service. For more information, see Appendix - Executing the Graph_Alive Notebook.

4.8 Accessing the FCC Studio Application

Access the FCC Studio application. For more information, see Appendix - Accessing the FCC Studio Application.

5 Post-installation Configuration

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

- For post-installation configuration for FCC Studio installed with OFSAA, follow these steps:

  Topics:

  - Configuring the SSH Connection
  - Configuring the Interpreters
  - Configuring ICIJ
  - Performing the OFSAA Configuration for Batch Execution
  - Performing Hive Data Movement
  - Configuration for Running Published Notebooks

- For post-installation configuration for FCC Studio installed with non-OFSAA, follow these steps:

  Topics:

  - Configuring the SSH Connection
  - Configuring the Interpreters
  - Configuring ICIJ

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

5.1 Configuring the SSH Connection

To configure the SSH connection, follow these steps:

1. Run the following command in the console:
   a. `Run ssh-keygen`

      `Generating public/private rsa key pair`

   b. `Enter file in which to save the key (<Linux_Home>/.ssh/id_rsa)`: [Press Enter]

   c. `Enter passphrase (empty for no passphrase)`: [Press Enter]

   d. `Enter same passphrase again`: [Press Enter]

   e. `ssh-copy-id -i ~/.ssh/id_rsa.pub <BigData_Server>`

   f. `ssh <BigData_Server>`

5.2 Configuring the Interpreters

After starting the FCC Studio application, you can perform the interpreter settings. For more information, see Appendix - Post-installation Interpreter Settings.

5.3 Configuring ICIJ

To configure ICIJ (International Consortium of Investigative Journalists), follow these steps:

1. Cleaning the ICIJ Data
2. Configuring the FILEPATH for ICIJ

5.3.1 Cleaning the ICIJ Data

To clean the ICIJ data, follow these steps:

1. Download the four dataset directories from *https://offshoreleaks.icij.org/pages/database*.
2. Unzip the four dataset directories and place the unzipped directories in the `<Studio_Installation_Path>/icij_data_cleaning` directory.
3. Navigate to the `<Studio_Installation_Path>/icij_data_cleaning/bin` directory and execute the following command:

   ```
   ./clean.sh
   ```

   NOTE Ensure that Python 3 is available in the machine before executing this command.

   After successful execution of the command:

   - The cleaned data is available for the sqoop job to load it in Hive and HDFS.
   - A directory named "clean" is created inside each of the dataset where a clean version of each CSV file is created.

5.3.2 Configuring the FILEPATH for ICIJ

NOTE The FCC Studio graph model is configured to include ICIJ watchlist files.

To configure the `FILEPATH` for ICIJ, follow these steps:

1. Place the watchlist files in HDFS, that is accessible by the user.
2. Update the `FILEPATH` of the watch ist files in the `fcc_studio_etl_files` table.

Figure 1: fcc_studio_etl_files Table

	DF_NAME	FILEPATH	DF_SEQ_NO	FILE_ORDER
1	Offshore_edges_is_related_to		12	1
2	Bahama_External_Address		13	1

5.4 Performing the OFSAA Configuration for Batch Execution

NOTE This configuration is not applicable for FCC Studio installed with non-OFSAA.

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

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

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

3. Copy all the files in the `<Studio_Installation_Path>/ficdb/lib` directory and paste into the `$FIC_DB_HOME/lib` directory.

   For information on running FCC Studio Batches, see *Managing Studio Batches* chapter in the *Oracle Financial Services Crime and Compliance Studio Administration Guide*.

5.5 Performing Hive Data Movement

> **NOTE** This configuration is not applicable for FCC Studio installed with non-OFSAA.

To perform Hive data movement, follow these steps:

- Configuring Schema Creation
- Creating the Credential Keystore
- Configuring the Data Movement and Graph Load

5.5.1 Configuring Schema Creation

- Configuring Schema Creation from FCC Studio Server
- Configuring Schema Creation from OFSAA Server

5.5.1.1 Configuring Schema Creation from FCC Studio Server

To configure Schema Creation from FCC Studio server, follow these steps:

1. Set `FIC_DB_HOME` path to `<Studio_Installation_Path>/ficdb`.

   > **NOTE** The `$FIC_DB_HOME` path can be set from the `.profile` file as well.

2. Create a Hive Schema with the name mentioned in the `HIVE_SCHEMA` parameter in the `config.sh` file.

   For information on `config.sh` file, see Appendix - Configuring the config.sh File.

3. Execute the following shell script in the `<Studio_Installation_Path>/ficdb/bin/` path to create tables in Hive Schema:

   ```
   FCCM_Studio_SchemaCreation.sh HIVE
   ```

4. Check the `<Studio_Installaton_Path>/logs/batchservice.logs` for more information.

5.5.1.2 Configuring Schema Creation from OFSAA Server

To configure Schema Creation from OFSAA server, follow these steps:

1. Copy all the jar files located in the `<Studio_Installation_Path>/ficdb/lib` path and paste into the `<OFSAA_FIC_HOME_PATH>/ficdb/lib` path.
2. Copy all the `.sh` files located in the `<Studio_Installation_Path>/ficdb/bin` path and paste into the `<OFSAA_FIC_HOME_PATH>/ficdb/bin` path.

3. Create a Hive Schema with the name mentioned in the `HIVE_SCHEMA` parameter in the `config.sh` file.

   For information on `config.sh` file, see Appendix - Configuring the config.sh File.

4. Execute the following shell script in the `<OFSAA_FIC_HOME_PATH>/ficdb/bin` path to create tables in Hive Schema:

   ```
   FCCM_Studio_SchemaCreation.sh HIVE
   ```

5. Check the `<Studio_Installaton_Path>/logs/batchservice.logs` for more information.

5.5.2 Creating the Credential Keystore

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.

5.5.3 Configuring the Data Movement and Graph Load

> **NOTE**
> - The Big Data System Administrator must place the following files in all nodes of the Spark cluster:
>   - `batchservice-8.0.7.4.0.jar`
>   - ES_Hadoop jar (`elasticsearch-spark-20_2.11-7.3.2.jar`)
> - Ensure that the path of the jar files are present in the Spark `classpath` in the `spark-defaults.conf` file.
> - Ensure to remove the older batchservice jars from the Spark `classpath` in the `spark-defaults.conf` file.

To configure the Data Movement and Graph Load, follow these steps:

1. Copy the required `FCCM_Studio_SqoopJob.sh` files from the `<Studio_Installation_Path>/ficdb/bin` directory and paste into the `<FIC_HOME of OFSAA_Installed_Path>/ficdb/bin` directory.

For information on performing Data Movement and Graph Load, see the *Data Movement and Graph Loading for Big Data Environment* section in the *OFS Crime and Compliance Studio Administration Guide*.

5.6 Configuration for Running Published Notebooks

> **NOTE** This configuration is not applicable for FCC Studio installed with non-OFSAA.

To perform the configuration required to run published notebooks, follow these steps:

1. Copy the required `FCCM_Studio_NotebookExecution.sh` file from the `<Studio_Installation_Path>/ficdb/bin` directory and paste into the `<FIC_HOME of OFSAA_Installed_Path>/ficdb/bin` directory.

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

6 Upgrading FCC Studio

This chapter provides the information required to upgrade FCC Studio.

Topics:

6.1 Additional Cleanup for Upgrade

To perform the cleanup required for the upgrade, follow these steps:

1. Navigate to the `DATABASECHANGELOG` table in the Studio Schema.
2. Run the following SQL command in the Studio Schema:

   ```
   select * from DatabaseChangeLog a where REGEXP_LIKE
   (a.id,'FCC_DATASTUDIO_CONFIG')
   ```

   The entries in the `DATABASECHANGELOG` table that match the criteria are displayed.
3. Delete all the entries generated as the result of the command.

   ```
   For example:
   FCC_DATASTUDIO_CONFIG_8.0.7.3.0
   FCC_DATASTUDIO_CONFIG_8.0.7
   ```

6.2 Upgrade Overview

You can upgrade an existing instance of FCC Studio as follows:

- Upgrade FCC Studio from v8.0.7.x onwards to v8.0.7.4.0 with OFSAA.

  NOTE Here, ensure to provide the same BD database, Studio schema, Hive schema, wallet related information that you used during the installation of the existing instance FCC Studio.

- Upgrade FCC Studio from v8.0.7.2.0 onwards to v8.0.7.4.0 without OFSAA.

  NOTE Here, ensure to provide the same Studio schema and wallet related information that you used during the installation of the existing instance of FCC Studio.

This section provides quick-start steps to upgrade FCC Studio.

Table 1: Quick-Start Steps to Upgrade FCC Studio

Sl. No.	Steps	Reference Links
1.	Prepare for installation	**NOTE:** Ensure that the Prerequisite Environmental Settings are met before proceeding with the upgrade. 1. Performing Pre-installation Tasks a. Obtaining the Software b. Extracting the Software 2. Required File Structure 3. Interpreter Settings
2.	Install FCC Studio	1. Configuring the Elastic Search Component 2. Enabling Synonym/Stopword with the Elastic Search Service 3. Placing the Required Files 4. Configuring the config.sh File 5. Running the FCC Studio Installer 6. Verifying the FCC Studio Installation 7. Installing PGX Service 8. Accessing the FCC Studio Application
3	Post-installation configuration for FCC Studio installed with OFSAA	**NOTE:** The post-installation configuration that are not performed as part of the existing installation of FCC Studio must be performed during the upgrade. • Configuring the SSH Connection • Configuring the Interpreters • Configuring ICIJ • Performing the OFSAA Configuration for Batch Execution • Performing Hive Data Movement **NOTE:** During the upgrade, ensure that you have removed all the older jars and have placed the new jars in all nodes of the Spark cluster. For more information, see Configuring the Data Movement and Graph Load. • Configuration for Running Published Notebooks
4	Post-installation configuration for FCC Studio installed with non-OFSAA	**NOTE:** The post-installation configuration that are not performed as part of the existing installation of FCC Studio must be performed during the upgrade. • Configuring the SSH Connection • Configuring the Interpreters • Configuring ICIJ

6.2.1 Configuring the config.sh File for Upgrade

Configure the `config.sh` file to upgrade FCC Studio. For more information, see Appendix - Configuring the config.sh File.

7 Reinstalling FCC Studio

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

To reinstall FCC Studio, follow these steps:

1. Navigate to the `<Studio_Installation_Path>/bin` directory.
2. Create a backup for the existing `config.sh` file.
3. Delete the `<Studio_Installation_Path>` directory.
4. Obtain and extract the FCC Studio installer archive file. For more information, see Performing Pre-installation Tasks.
5. Perform database cleanup by performing the following:

Table 1: Database Cleanup

Schema	Applicable for FCC Studio with OFSAA	Applicable for FCC Studio with Non-OFSAA
Clean up for Studio Schema	Yes	Yes
Cleanup for BD or ECM Atomic Schema	Yes	No
Cleanup for BD or ECM Config Schema	Yes	No

6. Reinstall FCC Studio. For more information, see Chapter 4, "Installing FCC Studio" and Chapter 7, "Installing FCC Studio With Non-OFSAA".

7.1 Clean up for Studio Schema

To clean up the Studio schema, follow these steps:

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

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

2. Grant the following permissions to the newly created Oracle Database Schema:
   - `GRANT create session TO <Schema User>;`
   - `GRANT create table TO <Schema User>;`
   - `GRANT create view TO <Schema User>;`
   - `GRANT create any trigger TO <Schema User>;`
   - `GRANT create any procedure TO <Schema User>;`
   - `GRANT create sequence TO <Schema User>;`
   - `GRANT execute on dbms_rls TO <Schema User>;`
   - `GRANT execute on sys.dbms_session TO <Schema User>;`
   - `ALTER USER <Schema User> QUOTA 100M ON users;`

- `GRANT create sequence TO <Schema User>;`
- `GRANT create SYNONYM TO <Schema User>;`
- `GRANT ALL privileges TO <Studio Schema User>;`

> **NOTE** If dropping the schema is not an option, drop the tables and sequences as mentioned in the Appendix - Tables and Sequences section.

7.2 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;
```

7.3 Cleanup for BD or ECM Config Schema

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

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

```
TRUNCATE TABLE DATABASECHANGELOGLOCK;

TRUNCATE TABLE DATABASECHANGELOG;
```

8 Appendix - Pre-installation Interpreter Settings

You must perform the interpreter settings for the required interpreters before installing FCC Studio.

> **NOTE** Ensure to perform the following pre-installation settings only for the interpreters that you need.

Table 1: Pre-installation Interpreter Settings

Interpreter	Prerequisite Settings
fcc-jdbc	For the required configuration, see Configuring the fcc-jdbc Interpreter. **NOTE:** The FCC Studio application installed with non-OFSAA can use the Vanilla jdbc interpreter instead of the fcc-jdbc interpreter to connect to the Studio schema.
fcc-ore	For the required configuration, see Configuring the fcc-ore Interpreter.
fcc-pyspark	• Install the py4j package in the Spark cluster. • Install the Livy server (0.5.0) on the master node of the Big Data cluster.
fcc-python	For the required configuration, see Configuring the fcc-python Interpreter.
fcc-spark-scala	Install the Livy server (0.5.0) on the master node of the Big Data cluster.
fcc-spark-sql	Install the Livy server (0.5.0) on the master node of the Big Data cluster.
jdbc	No additional configuration is required.
md	No additional configuration is required.
pgql	No additional configuration is required.
pgx-algorithm	No additional configuration is required.
pgx-java	No additional configuration is required.
pyspark	For the required configuration, see Configuring the PySpark Interpreter.
spark	For the required configuration, see Configuring the Spark Interpreter

Topics:

8.1 Configuring the fcc-jdbc Interpreter

To create context for the fcc-jdbc interpreter, follow these steps:

1. Log in to Oracle Database as a SYSDBA user.
2. Grant Execute permission to the user using the following command:

```
grant execute dbms_rls to <Studio_DB_Username>;
```

The Execute permission is granted to user.

3. Grant Create permission to the context using the following command:

```
grant create any context to <Studio_DB_Username>;
```

The Create permission is granted to context.

8.2 Configuring the fcc-ore Interpreter

8.2.1 Installing the Oracle R Distribution

To install the Oracle R Distribution (ORD), enable the **addons** and **optional_latest** channels in yum as shown in the following code:

```bash
(root)# yum-config-manager --enable ol7_addons
(root)# yum-config-manager --enable ol7_optional_latest
```

After completing the previous step, pull ORD from the yum repository using the following command:

```bash
(root)# yum install R.x86_64 R-core-extra
```

To install ORD, see *https://docs.oracle.com/cd/E83411_01/OREAD/installing-R-for-ORE.htm#OREAD129*.

8.3 Configuring the fcc-python Interpreter

FCC Studio supports any version of Python. The FCC Studio installer is also packaged with the following Python 3.6 libraries:

- `pandas 0.25.3`
- `numpy 1.17.4`
- `scipy 1.3.2`
- `scikit-learn 0.21.3`
- `matplot-lib 3.1.1`
- `seaborn 0.9.0`
- `cx-oracle 7.2.2`
- `sqlalchemy 1.3.11`

Based on your preference, you can choose to install either the Python 3.6 libraries or different versions/packages of Python.

8.3.1 Prerequisites for Installing Python Libraries

- FCC Studio supports any-version of Python with the following required conditions:

  - The preferred Python version is installed on the Processing Server (Studio Notebook Server) and is accessible to the Linux user for FCC Studio.
  - The Python Library (py4j) package is present for the preferred Python version.
- The prerequisites to install the Python 3.6 libraries that are packaged with the FCC Studio Installer are as follows:
  - The Python3.6 version is installed on the Processing Server (Studio Notebook Server) and is accessible to the Linux user for FCC Studio.
  - The Python Library (py4j) package is installed and is accessible to the Linux user of FCC Studio.

8.3.2 Installing the Python 3.6 Libraries

For information on installing Python 3.6 libraries, see Installing the Python 3.6 Libraries.

8.4 Configuring the Spark Interpreter

8.4.1 Prerequisites

To operate the Spark interpreter in local mode or yarn mode, perform the following:

8.4.1.1 Local Mode

No additional configuration is required to operate the Spark interpreter in local mode.

8.4.1.2 Yarn Mode

To operate the Spark interpreter in yarn mode, follow these steps:

1. Provide custom Spark libraries.

   To provide your own Spark libraries and/or Hadoop client-configuration files for the Spark interpreter to connect to a Spark Standalone or Yarn cluster, follow these steps:

   a. Download the Spark libraries from the *Spark's Official Release Archive.*

   b. Place the unarchived Spark libraries in a directory that is accessible by FCC Studio.

   c. Set the system environment variable, `SPARK_HOME` to the absolute path of the directory that contains the Spark libraries.

   d. Obtain the Hadoop client-configuration files from the Yarn cluster by performing one of the following:

      — Copy the Hadoop configuration directory, `$HADOOP_HOME/etc/hadoop` or `$HADOOP_HOME/conf` that contains the `core-site.xml` and `hdfs-site.xml` files.

      Where,

      `HADOOP_HOME` is either `/etc/hadoop` or `/usr/libs/hadoop`.

- Download the Hadoop client-configuration files directly from the cluster manager's UI if you are using a Cloudera cluster.

e. Place the Hadoop client-configuration files to a directory accessible by FCC Studio.

f. Set the system environment variable, `HADOOP_CONF_DIR` to the absolute path of the directory that contains the Hadoop client-configuration files.

2. Provide the cluster's Hadoop client-side configuration files that include XML files such as `yarn-site.xml` along with the Spark libraries.

   Obtain the cluster's Hadoop client-side configuration files by performing one of the following:

   - Copy the Hadoop configuration directory, HADOOP_CONF_DIR of the cluster.
   - Download the cluster's Hadoop client-side configuration files directly from the cluster manager's UI if you are using a Cloudera cluster.

8.4.2 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 FCC Studio application UI. This includes properties such as `spark.executor.memory`.

  NOTE The properties related to the driver cannot be set during runtime and are considered deployment configuration. 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 are available in the *Spark Official Documentation*. All the properties prefixed with the term "zeppelin", that are listed in the *Zeppelin Spark Configuration Document* can also be set from the ***Interpreters*** page of the FCC Studio application UI.

8.5 Configuring the PySpark Interpreter

8.5.1 Prerequisites

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

8.5.2 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 FCC Studio application UI during runtime by changing the following properties:

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

8.5.3 Using 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. Creating a Virtual Environment with Conda
2. Updating the Interpreter Properties

8.5.3.1 Creating 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
```

8.5.3.2 Updating the Interpreter Properties

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

- In the ***Spark Interpreter Settings*** page of the FCC 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 FCC Studio application UI (or `pyspark.json`), change the value of the `zeppelin.pyspark.python` parameter to `<environment-abs-path>/bin/python`.

9 Appendix - Setting Up Password Stores with Oracle Wallet

Topics:

9.1 Overview

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.

9.2 Setting Up 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 set up a wallet and the aliases for the database user accounts. For more information on configuring authentication and password stores, refer to the Oracle Database Security Guide.

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

To set up a password store for the database user accounts, follow these steps:

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

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

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

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

   The wallet is created with the auto-login feature enabled. This feature enables the database client to access the wallet contents without using the password. For more information, refer to the Oracle Database Advanced Security Administrator's Guide.
3. Create the database connection credentials in the wallet using the following command:

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

Run the above command for the following <alias-name>:

Schema	Applicable for FCC Studio with OFSAA	Applicable for FCC Studio with Non-OFSAA
BD_Config_Schema	Yes	No
BD_Atomic_Schema	Yes	No
Studio_Schema	Yes	Yes

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.

4. Repeat step 2 for all the database user accounts.
5. Update the `tnsnames.ora` file to include the following entry for each alias name to be set up.

```
<alias-name> =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP) (HOST = <host>) (PORT = <port>))
)
(CONNECT_DATA =
(SERVICE_NAME = <service>)
)
)
```

NOTE

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

9.3 Verifying the Connectivity of the Wallet

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

1. Create a `sqlnet.ora` 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 what the wallet was created with.

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

`$ export TNS_ADMIN=<tnsnames.ora_location>`, Here ensure to use the wallet to point to the alternate `tnsnames.ora` as created above.

```
$ 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>"
```

10 Appendix - Configuring the Elastic Search Component

To configure the Elastic Search component, follow these steps:

> **NOTE**
> - Ensure that a minimum of 4GB free RAM space is available for elastic search. If RAM is low, the shards of the elastic search fails and correct result is not fetched.
> - You must manually clean the cache if facing performance issue.

1. Navigate to the `elastimancsearch-7.3.2/config` directory.
2. Configure the `elasticsearch.yml` with the following variables:

Table 1: elasticsearch.yml File

InteractionVariable Name	Significance
cluster.name	Indicates the name of the cluster.
node.name	Indicates the name given for the node.
node.master	Indicates whether the node is a master.
node.data	Indicates the node data.
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 elastic search service.
http.port	Indicates the port number where the elastic search service is installed.
discovery.seed_hosts	(Optional) Indicates the hostnames of the nodes of the cluster.
cluster.initial_master_nodes	(Optional) Indicates the number given to the nodes of the cluster.
indices.breaker.total.use_real_memory	• Indicates the static setting to determine whether the parent breaker must consider the real memory usage into account or only consider the amount that is reserved by the child circuit breakers. • This setting is used to prevent the `OutOfMemory` error.

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

Table 2: elasticsearch.yml File

InteractionVariable Name	Significance
-Xms1g	• 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.
-Xmx1g	

4. Enter the URL in the following format into the browser:

   `http://<network.host>:<http.port>`

   The following output is displayed to indicate successful installation of the Elastic Search service.

```
{
  "name" : "node-1",
  "cluster_name" : "my-application",
  "cluster_uuid" : "_D-2qEGGSgKQPd3i-UtgWQ",
  "version" : {
    "number" : "7.3.2",
    "build_flavor" : "oss",
    "build_type" : "tar",
    "build_hash" : "1c1faf1",
    "build_date" : "2019-09-06T14:40:30.409026Z",
    "build_snapshot" : false,
    "lucene_version" : "8.1.0",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

11 Appendix - Configuring the config.sh File

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

1. Login to the server as a non-root user.
2. Navigate to the `<Studio_Installation_Path>/bin/` directory.
3. Configure the `config.sh` file as follows:

NOTE

- Do not alter the parameter values that are already set in the `config.sh` file
- 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.
- Depending on the installation architecture, ensure to provide the correct hostname for URL of PGX service in the `config.sh` file.
- During upgrading FCC Studio with OFSAA, ensure to provide the same BD database, Studio schema, Hive schema, wallet related information that you used during the installation of the existing instance FCC Studio.
- During upgrading FCC Studio with non-OFSAA, ensure to provide the same Studio schema and wallet related information that you used during the installation of the existing instance of FCC Studio.

Table C–1 config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
FCC_STUDIO_INSTALLATION_PATH	Indicates the path where the FCC Studio installer file is extracted.	Yes	Yes	Yes	Yes
NON_OFSAA	• To install FCC Studio with OFSAA, enter "false" • To install FCC Studio with non-OFSAA, enter "true"	Enter false	Enter false	Enter true	Enter true

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
REALM	Realm indicates 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 additional privileges to its owner or participants. FCC Studio uses realm based authorization and authentication for its users. For more information, see the *Realm Based Authorization for FCC Studio* section in the *OFS Crime and Compliance Studio Administration Guide.* The FCC Studio application can be accessed using the following realms: • FCCMRealm Value=`com.oracle.ofss.fccm.studio.datastudio.auth.FCCMRealm` • IdcsRealm Value=`oracle.datastudio.realm.idcs.IdcsRealm` • DemoRealm Value=`com.oracle.ofss.fccm.studio.datastudio.auth.DemoRealm` **NOTE:** The DemoRealm is used only for demo purpose and is not recommended for usage.	Yes	Yes	Yes	Yes

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
FCDM_SOURCE	Indicates the source database for FCC Studio. The available options are ECM and BD. **NOTE:** • FCC 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, ECM schema is used as the source of the FCDM data to load the case information into the graph.	Enter BD or ECM	Enter BD or ECM	Enter NA	Enter NA
IDCS **NOTE:** The IDCS related parameters are applicable only if IdcsRealm is used in the Realm parameter.					
IDCS_HOST	Indicates the server of the Oracle Identity Cloud Service (IDCS) instance.	Yes	Yes	Yes	Yes
IDCS_PORT	Indicates the port number of the IDCS instance.	Yes	Yes	Yes	Yes
IDCS_SSL_ENABLED	Indicates if SSL is enabled for the IDCS application. Default value: true	Yes	Yes	Yes	Yes
LOGOUT_URL	Indicates the URL to redirect after logout from FCC Studio.	Yes	Yes	Yes	Yes
IDCS_TENANT	Indicates the IDCS tenant provided by the IDCS Administrator while creating the IDCS application for FCC Studio.	Yes	Yes	Yes	Yes

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
IDCS_CLIENT_ID	Indicates the IDCS client identifier provided by the IDCS Administrator while creating the IDCS application for FCC Studio.	Yes	Yes	Yes	Yes
IDCS_CLIENT_SECRET	Indicates the IDCS client secret provided by the IDCS Administrator while creating the IDCS application for FCC Studio.	Yes	Yes	Yes	Yes
External Services					
OFSAA_SERVICE_URL	Indicates the URL of the OFSAA instance. Do not enter '/' at the end of the URL. **NOTE:** For OFSAAI, the value must be set in the following format: `https://<HostName>:<PortNo>/<ContextName>/rest-api`	Yes	Yes	No	No
PGX_SERVER_URL	Indicates the URL of the PGX server. Example: `http://<HostName>:<PortNo>/` Here, default `<PortNo>` is 7007.	Yes	Yes	Yes	Yes
LIVY_HOST_URL	Indicates the URL of the Livy application. Example: `http://<HostName>:<PortNo>` **NOTE:** This parameter is applicable only if the fcc-spark-sql, fcc-spark-scala and/or fcc-pyspark interpreters are to be used.	No	No	No	No
Internal Services					

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
AUTH_SERVICE_URL	Indicates the AUTH service URL that gets activated after the `fcc-studio.sh` file runs. Example: `http://<HostName>:7041/authservice`	Yes	Yes	No	No
BATCH_SERVICE_URL	Indicates the batch service URL that gets activated after the `fcc-studio.sh` file runs. Example: `http://<HostName>:7043/batchservice`	Yes	Yes	Yes	Yes
META_SERVICE_URL	Indicates the metaservice URL that gets activated after the `fcc-studio.sh` file runs. Example: `http://<HostName>:7045/metaservice`	Yes	Yes	Yes	Yes
SESSION_SERVICE_URL	Indicates the session service URL that gets activated after the `fcc-studio.sh` file runs. Example: `http://<HostName>:7047/sessionservice`	Yes	Yes	Yes	Yes
FCC_STUDIO_URL	Indicates the FCC Studio URL. Example: `http://<HostName>:7008`	Yes	Yes	Yes	Yes
ORE Interpreter Settings **NOTE:** This section is applicable only if ORE interpreter is to be used.					

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
RSERVE_USERNAME	Indicates the RServe username.	No	No	No	No
RSERVE_PASSWORD	Indicates the RServe password.	No	No	No	No
HTTP_PROXY	Indicates the proxy for the host where FCC Studio is installed.	No	No	No	No
HTTPS_PROXY	Indicates the proxy for the host where FCC Studio is installed.	No	No	No	No
REPO_CRAN_URL	Indicates the URL from where the R packages are obtained. The format for the REPO_CRAN_URL is as follows: *https://cran.r-project.org/*.	No	No	No	No
USERS_LIB_PATH	Indicates the path where the R packages are installed.	No	No	No	No
RSERVE_CONF_PATH	Indicates the path where the `Rserve.conf` file is present.	No	No	No	No
DB Details for Studio Schema					
STUDIO_DB_HOSTNAME	Indicates the hostname of the database where Studio schema is created.	Yes	Yes	Yes	Yes
STUDIO_DB_PORT	Indicates the port number where Studio schema is created.	Yes	Yes	Yes	Yes
STUDIO_DB_SERVICE_NAME	Indicates the service name of the database where Studio schema is created.	Yes	Yes	Yes	Yes
STUDIO_DB_SID	Indicates the SID of the database where Studio schema is created.	Yes	Yes	Yes	Yes
STUDIO_DB_USERNAME	Indicates the username of the Studio Schema (newly created Oracle Schema).	Yes	Yes	Yes	Yes
STUDIO_DB_PASSWORD	Indicates the password of the Studio schema.	Yes	Yes	Yes	Yes

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
Studio DB Wallet Details For more information on creating wallet, see Appendix - Setting Up Password Stores with Oracle Wallet.					
STUDIO_ALIAS_NAME	Indicates the Studio alias name. **NOTE:** Enter the alias name that was created during wallet creation.	Yes	Yes	Yes	Yes
STUDIO_WALLET_LOCATION	Indicates the Studio wallet location.	Yes	Yes	Yes	Yes
STUDIO_TNS_ADMIN_PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `STUDIO_ALIAS_NAME` is present.	Yes	Yes	Yes	Yes
DB Details for BD Config Schema					
BD_CONFIG_HOSTNAME	Indicates the hostname of the database where BD or ECM config schema is installed.	Yes	Yes	Enter NA	Enter NA
BD_CONFIG_PORT	Indicates the port of the database where BD or ECM config schema is installed.	Yes	Yes	Enter NA	Enter NA
BD_CONFIG_SERVICE_NAME	Indicates the service name of the database where BD or ECM config schema is installed.	Yes	Yes	Enter NA	Enter NA
BD_CONFIG_SID	Indicates the SID of the database where BD or ECM config schema is installed.	Yes	Yes	Enter NA	Enter NA
BD_CONFIG_USERNAME	Indicates the username for the BD or ECM config schema.	Yes	Yes	Enter NA	Enter NA
BD_CONFIG_PASSWORD	Indicates the password for the BD or ECM config schema.	Yes	Yes	Enter NA	Enter NA

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
BD Config Wallet Details For more information on creating wallet, see Appendix - Setting Up Password Stores with Oracle Wallet.					
BD_CONFIG_ALIAS_NAME	Indicates the BD or ECM config alias name. **NOTE:** Enter the alias name that was created during wallet creation.	Yes	Yes	Enter NA	Enter NA
BD_CONFIG_WALLET_LOCATION	Indicates the BD or ECM config wallet location.	Yes	Yes	Enter NA	Enter NA
BD_CONFIG_TNS_ADMIN_PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `BD_CONFIG_ALIAS_NAME` is present.	Yes	Yes	Enter NA	Enter NA
DB Details for BD Atomic Schema					
BD_ATOMIC_HOSTNAME	Indicates the BD or ECM atomic schema hostname.	Yes	Yes	Enter NA	Enter NA
BD_ATOMIC_PORT	Indicates the BD or ECM atomic schema port number.	Yes	Yes	Enter NA	Enter NA
BD_ATOMIC_SERVICE_NAME	Indicates the BD or ECM atomic schema service name.	Yes	Yes	Enter NA	Enter NA
BD_ATOMIC_SID	Indicates the BD or ECM atomic schema SID.	Yes	Yes	Enter NA	Enter NA
BD_ATOMIC_USERNAME	Indicates the username of the BD or ECM atomic schema.	Yes	Yes	Enter NA	Enter NA
BD_ATOMIC_PASSWORD	Indicates the password of the BD or ECM atomic schema.	Yes	Yes	Enter NA	Enter NA

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
BD Atomic Wallet Details For more information on creating wallet, see Appendix - Setting Up Password Stores with Oracle Wallet.					
BD_ATOMIC_ALIAS_NAME	Indicates the BD or ECM atomic alias name. **NOTE:** Enter the alias name that was created during wallet creation.	Yes	Yes	Enter NA	Enter NA
BD_ATOMIC_WALLET_LOCATION	Indicates the BD or ECM atomic wallet location.	Yes	Yes	Enter NA	Enter NA
BD_ATOMIC_TNS_ADMIN_PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `BD_ATOMIC_ALIAS_NAME` is present.	Yes	Yes	Enter NA	Enter NA
SQL Scripts					
FSINFODOM	Indicates the name of the BD or ECM Infodom.	Yes	Yes	Enter NA	Enter NA
FSSEGMENT	Indicates the name of the BD or ECM segment.	Yes	Yes	Enter NA	Enter NA
DATAMOVEMENT_LINK_NAME	• If the Studio schema is in a different database host, you must create a DB link and provide the same link in this parameter. If no DB link is present, provide the BD or ECM Atomic schema name in this parameter. • If the Studio schema is in the same database host, the value for this parameter is the user name of the BD or ECM atomic schema.	Yes	Yes	Yes	Yes

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
DATAMOVEMENT_LINK_TYPE	If the DB link is used, enter DBLINK in this field. If the DB link is not used, enter SCHEMA in this field.	Yes	Yes	Yes	Yes
PGX Setup Details					
PGX_INSTALATION_PATH	Indicates the installation path of the PGX server. Example: `<STUDIO_INSTALLATION_PATH>/pgx/server/`	Yes	Yes	Yes	Yes

PGX_PGB_PATH	Indicates the path where you want to obtain the output graph PGB file.
	Example for Hive Installation:
	`hdfs:/user/ofsaa`
	Example for Oracle DB Installation:
	`/scratch/ofsaa`
		Yes	Yes	Yes	Yes

Cloudera Setup Details Contact System Administrator to obtain the required details for these parameters.					
HADOOP_CREDENTIAL_PROVIDER_PATH	Indicates the path where Hadoop credential is stored.	Yes	Yes	Enter NA	Enter NA
HADOOP_PASSWORD_ALIAS	Indicates the Hadoop alias given when creating the Hadoop credentials. **Note**: Indicates the Hadoop alias given when creating the Hadoop credentials. For information on creating credential keystore, see Creating the Credential Keystore.	Yes	Yes	Enter NA	Enter NA
Hive_Host_Name	Indicates the Hive hostname.	Yes	Yes	Enter NA	Enter NA

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
Hive_Port_number	Indicates the Hive port number. Contact System Administrator to obtain the port number.	Yes	Yes	Enter NA	Enter NA
HIVE_PRINCIPAL	Indicates the Hive Principal. Contact System Administrator to obtain HIVE_PRINCIPAL.	Yes	Yes	Enter NA	Enter NA
HIVE_SCHEMA	Indicates to create a schema in HIVE.	Yes	Yes	Enter NA	Enter NA
JAAS_CONF_FILE_PATH	Created for future use.	No	No	No	No
Krb_Host_FQDN_Name	Indicates the Krb host FQDN name.	Yes	Yes	Enter NA	Enter NA
Krb_Realm_Name	Indicates the Krb realm name.	Yes	Yes	Enter NA	Enter NA
Krb_Service_Name	Indicates the Krb service name. Example: Hive	Yes	Yes	Enter NA	Enter NA
KRB5_CONF_FILE_PATH	Created for future use.	No	No	No	No
security_krb5_kdc_server	Created for future use.	No	No	No	No
security_krb5_realm	Created for future use.	No	No	No	No
server_kerberos_keytab_file	Indicates the Kerberos keytab file.	Yes	Yes	Enter NA	Enter NA
server_kerberos_principal	Indicates the Kerberos Principal.	Yes	Yes	Enter NA	Enter NA
SQOOP_HOSTMACHINE_USER_NAME	Indicates the user name of the Host machine where sqoop will run.	Yes	Yes	Enter NA	Enter NA

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
SQOOP_PARAMFILE_PATH	1. Create a file with the name `sqoop.properties` in the Big Data server and add the following entry: `oracle.jdbc.mapDateToTimestamp=false` 2. 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 '/'.	Yes	Yes	Enter NA	Enter NA
SQOOP_PARTITION_COL	Indicates the column in which the HIVE table is partitioned. The value must be `SNAPSHOT_DT`.	Yes	Yes	Enter NA	Enter NA
SQOOP_TRG_HOSTNAME	Indicates the hostname of the Big Data server where SQOOP will run. Example: `<HostName>`	Yes	Yes	Enter NA	Enter NA
SQOOP_WORKDIR_HDFS	Indicates the SQOOP working directory in HDFS. Example: `/user/ofsaa`	Yes	Yes	Enter NA	Enter NA
Elastic Search Cluster details					
ELASTIC_SEARCH_PORT	Indicates the port number where the elastic search service is installed.	Yes	Yes	Yes	Yes
ELASTIC_SEARCH_HOSTNAME	Indicates the hostname of the database where the elastic search service is installed.	Yes	Yes	Yes	Yes
Matching Service					
MATCHING_SERVICE_HOSTNAME	Indicates the hostname where the matching service is installed.	Yes	Yes	Yes	Yes

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
MATCHING_SERVICE_PORT	Indicates the port number where the matching service is installed.	Yes	Yes	Yes	Yes
EXECUTOR_THREADS	Indicates the number of threads to run in parallel during one scroll. For example: 10	Yes	Yes	Yes	Yes
ELASTICRESPONSE_BUFFERLIMIT_BYTE	Indicates the buffer size of the response obtained from the elastic search service. For example: 200	Yes	Yes	Yes	Yes
SCROLL_TIME	Indicates the duration for which the scroll_size output is active. For example: 5	Yes	Yes	Yes	Yes
SCROLL_SIZE	Indicates the amount of data that must be obtained in one attempt when a query is fired on an index in the elastic search service. For example: 1000	Yes	Yes	Yes	Yes
Entity Resolution					
ER_SERVICE_PORT	Indicates the port number where the entity resolution service is installed. Default Value: 7051 **NOTE:** This value is already set. Do not change the value.	Yes	Yes	Yes	Yes
ER_SERVICE_URL	Indicates the URL of the entity resolution service. For example: `http://<HostName>:7051`	Yes	Yes	Yes	Yes
Graphs					
HDFS_GRAPH_FILES_PATH	Indicates the filepath in the HDFS where the `graph.json` is formed.	Yes	Yes	Yes	Yes

Table C–1 (Continued)config.sh File

Parameter	Significance	Installing with OFSAA (Mandatory)	Upgrading with OFSAA (Mandatory)	Installing with non-OFSAA (Mandatory)	Upgrading with non-OFSAA (Mandatory)
GRAPH_FILES_PATH	Indicates the directory in the Big Data server for graph files.	Yes	Yes	Yes	Yes
GRAPH_NAME	Indicates the name you want to assign to the global graph at the end of ETL.	Yes	Yes	Yes	Yes

12 Appendix - Starting/Stopping FCC Studio Services

This section information on how to start and stop the services required for the FCC Studio application. It includes the following sections:

Start/Stop FCC Studio Services	Applicable for FCC Studio installed with OFSAA	Applicable for FCC Studio installed with Non-OFSAA
Starting/Stopping PGX Service	Yes	No
Starting/Stopping FCC Studio	Yes	Yes

12.1 Starting/Stopping PGX Service

- To start the PGX service, follow these steps:
  a. Navigate to the path where 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 the following command:

     `./start-pgx.sh`

  NOTE Ensure to execute the Graph_Alive notebook after each time you start/restart the PGX service. For more information, Appendix - Executing the Graph_Alive Notebook.

- To stop the PGX service, kill the process.

12.2 Starting/Stopping FCC Studio

- To start FCC Studio, follow these steps:
  a. Navigate to the `<Studio_Installation_Path>/bin/` directory.
  b. Run the following command:

     `./fcc-studio.sh`

- To stop FCC Studio, kill all the FCC Studio processes.

Once all the services are up and running, you can access the FCC Studio application. For more information, see Appendix - Accessing the FCC Studio Application.

13 Appendix - Tables and Sequences

The list of tables and sequences to be dropped during reinstallation of FCC Studio are as follows:

13.1 Studio Schema Tables

The following table lists the Studio Schema tables that must be dropped during reinstallation of FCC Studio.

Table 1: Studio Schema Tables

DS_PARAGRAPH	DS_NOTEBOOK_TAGS	DS_TASK_RESULTS
DS_ENTITY_PERMISSIONS	DS_ROLE	DS_PERMISSION_ACTIONS
DS_GROUP	DS_IS_PERMITTED	DS_PERMISSION_MAPPING
DS_USER_PERMS_MAP	DS_USER_ROLES	DS_NOTEBOOK
DS_INTERPRETER_RESULT_MSGS	DS_USER	DS_PERMS_MAP_ACTIONS
DS_ENTITY_PERMS_MAP	DS_TASK	DS_GRAPH
DS_INTERPRETER_RESULT	DS_GROUP_PERMS_MAP	DS_NOTEBOOK_RELATIONS
DS_INTERPRETER_PROPS	DS_JOB	DS_PERMISSION
DS_ROLE_PERMS_MAP	DS_VISUALIZATION_TEMPLATE	DS_RESULT_MESSAGE
DS_INTERPRETER_ABILITIES	DATABASECHANGELOG	DATABASECHANGELOGLOCK
DS_USER_GROUPS	DS_INTERPRETER_VARIANT	DS_COMMENT
DS_PARAGRAPH_RELATIONS		

13.2 Studio Schema Sequences

The following table lists the Studio Schema sequences that must be dropped during reinstallation of FCC Studio.

Table 2: Studio Schema Sequences

SEQ_COMMENT	SEQ_ENTITY_PERMISSIONS	SEQ_GRAPH
SEQ_GROUP	SEQ_INTERPRETER_RESULT	SEQ_INTERPRETER_VARIANT

Table 2: Studio Schema Sequences

SEQ_JOB	SEQ_NOTEBOOK	SEQ_PARAGRAPH
SEQ_PERMISSION	SEQ_PERMISSION_MAPPING	SEQ_RESULT_MESSAGE
SEQ_ROLE	SEQ_TASK	SEQ_USER
SEQ_VISUALIZATION_TEMPLATE		

14 Appendix - Accessing the FCC Studio Application

To access FCC Studio, follow these steps:

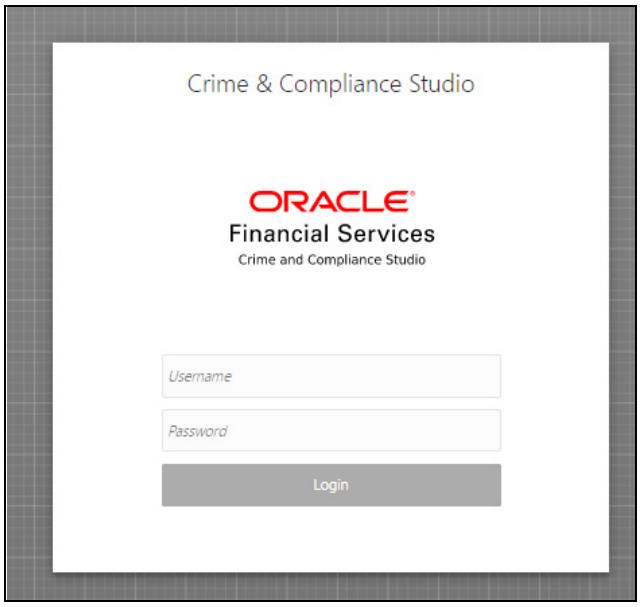
1. Enter the URL in the following format in the web browser:

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

   Here `<Port_Number>` is 7008 for the FCC Studio application installed on-premise.

   The FCC Studio application login page is displayed.

   Figure 1: Crime and Compliance Studio Login Page

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

   The FCC Studio application's landing page is displayed with the list of all the out-of-the-box notebooks packaged with FCC Studio.

15 Appendix - Enabling a Second Spark/PySpark Interpreter

To set up a second Spark/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 The PySpark interpreter does not have or need its own start-script.

   a. Navigate to the `<Studio_Installation_Path>/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 `<Studio_Installation_Path>/interpreters/bin/` directory to update the port number to a new port number that is not in use (for example, 7030) and rename the log file (for example, `spark2.sh`).

      In the `start-spark2-interpreter.sh` file,

      — Line 13 becomes:

      ```
      ${SPARK_SUBMIT} --class
      oracle.datastudio.interpreterserver.ZeppelinRemoteInterpreterServe
      r --driver-class-path ${CLASSPATH} --driver-java-options
      "${JAVA_OPTS} ${SPARK_INTERPRETER_OPTS} " --files
      ${py4j[0]},${SPARK_HOME}/python/lib/pyspark.zip ${SPARK_CONF}
      ${SPARK_APP_JAR} ${1:-7030} > $DIR/../../logs/spark2.log
      ```

      — Line 24 becomes:

      ```
      java -DlogFileName=spark -Dfile.encoding=UTF-8 ${JAVA_OPTS}
      ${SPARK_INTERPRETER_OPTS}
      oracle.datastudio.interpreterserver.ZeppelinRemoteInterpreterServe
      r ${1:-7030} > $DIR/../../logs/spark2.log
      ```

   c. Edit the `start-all-interpreters.sh` file in the `<Studio_Installation_Path>/interpreters/bin/` directory as follows:

      Insert the following code:

      ```
      sh "$DIR"/start-spark2-interpreter.sh &
      ```

      Below the following code:

      ```
      sh "$DIR"/start-spark-interpreter.sh &
      ```

2. Create the interpreter JSON for the second Spark interpreter.

   a. Navigate to the `<Studio_Installation_Path>/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 `<Studio_Installation_Path>/interpreters/conf/` directory as follows:

      Update the following parameter values:

```
group, name, groupSettings.initialCodeCapability, port,
capabilities.name, capabilities.button.label
```

to the following:

```
<new-spark-interpreter-name> ,<new-spark-interpreter-name>, <new-
spark-interpreter-name>, 7030 (the port chosen in the first step),
<new-spark-interpreter-name>, <new-spark-interpreter-name>
```

The first 23 lines of the new JSON file must be similar to the following code, where Spark2 is used for `<new-spark-interpreter-name>`:

```
[
  {
    "group": "spark2",
    "name": "spark2",
    "className": "org.apache.zeppelin.spark.SparkInterpreter",
    "groupSettings": {
      "initialCode": "True\n",
      "initialCodeCapability": "spark2"
    },
    "host": "localhost",
    "port": 7030,
    "capabilities": [
      {
        "name": "spark2",
        "highlightLanguage": "scala",
        "formEscapeCharacter": "@",
        "button": {
          "defaultCode": "println(\"Hello, world\")",
          "icon": "fa fa-fw fa-building-o",
          "label": "Spark2"
        }
      }
    ],
```

3. Create the interpreter JSON for the second PySpark interpreter.
   a. Navigate to the `<Studio_Installation_Path>/interpreters/conf` directory and create the new interpreter JSON called `pyspark2.json` using following command:

      ```
      cp pyspark.json pyspark2.json
      ```

   b. Edit the `pyspark2.json` file in the `<Studio_Installation_Path>/interpreters/conf/` directory as follows:

      Update the following parameter values:

```
group, name, port, capabilities.name, capabilities.button.label
```

to the following:

```
<new-spark-interpreter-name> ,<new-pyspark-interpreter-name>, 7030
(the port chosen in the first step), <new-pyspark-interpreter-name>,
<new-pyspark-interpreter-name>
```

The first 19 lines of the new JSON file must be similar to the following code, where spark2 is used for `<new-spark-interpreter-name>` and pyspark2 is used for `<new-pyspark-interpreter-name>`:

```
[
{
"group": "spark2",
"name": "pyspark2",
"className": "org.apache.zeppelin.spark.PySparkInterpreter",
"host": "localhost",
"port": 7030,
"capabilities": [
{
"name": "pyspark2",
"highlightLanguage": "python",
"button": {
"defaultCode": "print('Hello World')",
"icon": "icon-python",
"label": "pyspark2"
},
"formEscapeCharacter": "$"
}
],
```

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 Yarn Mode, To prepend the declaration of those environment variables to the respective Spark interpreter start-scripts, so that they don't interfere, rather than setting them system wide.

4. Restart FCC Studio. For more information, see Starting/Stopping FCC Studio.

16 Appendix - Executing the Graph_Alive Notebook

In an Investigation Hub notebook, the graph is lost whenever a session is reset and this occurs as part of the session clean-up. You must execute the Graph_Alive notebook to retain the link to the graph even when a session is reset.

To execute the Graph_Alive notebook, follow these steps:

1. Access the FCC Studio application. For more information, see Appendix - Accessing the FCC Studio Application.
2. Click the Graph_Alive notebook.

   The Graph_Alive notebook is displayed.
3. Click **Run Paragraphs**  to execute all the paragraphs in the notebook in a sequential order.

17 Appendix - Post-installation Interpreter Settings

After starting FCC Studio, the interpreter settings can be performed from the ***Interpreters*** page of the FCC Studio application UI. For information on configuring interpreters, see the *Configuring Interpreters* chapter in the *Oracle Financial Services Crime and Compliance Studio Administration Guide*.

17.1 Configuring the fcc-python Interpreter

17.1.1 Installing the Python 3.6 Libraries

To install the Python 3.6 libraries, follow these steps:

1. Navigate to the `<Studio_Installation_Path>/python-packages/bin` directory.
2. Execute the following command:

   ```
   chmod 0755 *
   ```

3. Install the Python 3.6 library packages that are packaged with the FCC Studio Installer using the following command:

   ```
   ./install.sh
   ```

   The Python 3.6 libraries that are packaged with the FCC Studio Installer are installed.

NOTE

- Execute the `./remove.sh` command to remove the Python 3.6 libraries that are already installed.
- To install additional Python libraries, contact System Administrator to install the same on the Processing Server (Studio Notebook Server).
- To change the default Python version in the Interpreters, see the *Changing Python Version in the fcc-python Interpreter* section in the *OFS Crime and Compliance Studio Administration Guide*.

OFSAA Support Contact Details

Raise a Service Request (SR) in My Oracle Support (MOS) for queries related to the 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.