Oracle® Financial Services Crime and Compliance Studio Application

Installation Guide

Release 8.0.7.2.0

E91246-01

February 2020

ORACLE®

Installation Guide, Release 8.0.7.2.0

E91246-01

Contents

B Tables and Sequences

C Setting Up Password Stores with Oracle Wallet

D Installing RServe Manually

E Creating Required Index Files

F Providing Spark Libraries

G Using Python Virtual Environments with PySpark

H Enabling a Second Spark/PySpark Interpreter

Document Control

Version Number	Revision Date	Changes Done
8.0.7.0.0	Created: November 2019	Created the first version of Oracle Financial Services Crime and Compliance Studio Deployment Guide for v8.0.7.0.0 Release.
8.0.7.1.0	Updated: October 2019	Updated the guide with the following newly introduced Studio services: • ETL • Data Forwarding • Cross Language Name Matching Data Forwarding service has been removed. For more information, see Installing ETL Services.
8.0.7.2.0	Updated: February 2020	• Install Studio application with or without BD. For more information, see Chapter 3, "Installing Studio With BD" and Chapter 6, "Installing Studio Without BD" • The support for the Data Forward service has been deprecated. • Upgrade an existing instance of the Studio application (v8.0.7.x) to the v8.0.7.2.0 version. For more information, see Chapter 5, "Upgrading Studio With BD" and • Configuration for the newly introduced interpreters such as Spark and PySpark interpreters. For more information, see Configuring Interpreters

Preface

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.

Audience

This document is intended for System Engineers who are responsible for installing and configuring or upgrading the OFS Crime and Compliance Studio application.

The document assumes that you have experience in installing Enterprise components. The basic knowledge of the following is recommended:

- UNIX commands
- Database concepts
- Big Data

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

Installation Overview

This chapter provides the information required to understand the installation of the Oracle Financial Services (OFS) Crime and Compliance Studio application with or without BD (Behavior Detection).

Introduction

This release (v8.0.7.2.0) of the OFS Crime and Compliance Studio application pack can be used for the following:

- Install a new instance of the FCC Studio application (v8.0.7.2.0) with or without BD.
- Upgrade an existing instance of the FCC Studio application (v8.0.7.x) to v8.0.7.2.0 version.

Quick Start Steps to Install Studio with BD

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

Table 1–1 Quick Start Steps to Install Studio with BD

Sl. No.	Steps	Reference Links

1.	Prepare for installation	1. Prerequisites
		2. Hardware and Software Requirements
		3. Prerequisite Environmental Settings
		4. Performing Common Pre-installation Tasks
		  1. Obtaining the Software
		  2. Extracting the Software
		5. Required File Structure
		6. Interpreter Settings

Table 1–1 Quick Start Steps to Install Studio with BD

Sl. No.	Steps	Reference Links

2.	Install the Studio application with BD	1. Installing the Studio Application With BD
		  1. Placing Required Files
		  2. Configuring the config.sh File
		  3. Running the Installer
		2. Verifying the Installation
		3. Installing ETL Services
		  1. Configuring ETL Services
		  2. Configuring FILEPATH for ICIJ
		4. Installing PGX
		5. Launching the FCC Studio Application

3	Post-installation configuration	• Configuring SSH Connection • Configuring Interpreters • Performing OFSAA Configuration for Batch Execution • Performing Hive Data Movement

Quick Start Steps to Install Studio without BD

This section provides the quick start steps to install a new instance of the FCC Studio application without BD.

Table 1–2 Quick Start Steps to Install Studio without BD

Sl. No.	Steps	Reference Links

1.	Prepare for installation	1. Prerequisites
		2. Hardware and Software Requirements
		3. Prerequisite Environmental Settings
		4. Performing Common Pre-installation Tasks
		  1. Obtaining the Software
		  2. Extracting the Software
		5. Interpreter Settings

2.	Install the Studio application without BD	1. Installing the Studio Application Without BD
		  1. Configuring the config.sh File
		  2. Running the Installer
		2. Verifying the Installation
		3. Launching the FCC Studio Application

Table 1–2 Quick Start Steps to Install Studio without BD

Sl. No.	Steps	Reference Links
3	Post-installation configuration	• Configuring SSH Connection • Configuring Interpreters

2

Preparing for Installation

This chapter provides the necessary information to review before installing the OFS Crime and Compliance Studio application. It includes the following sections:

Prerequisites

A new instance of the Studio application can be installed either with or without BD. To install the Studio application with BD, ensure the BD application pack is installed.

Hardware and Software Requirements

The hardware and software required to install Studio are as follows:

Table 2–1 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+

Table 2–1 (Cont.) Hardware and Software Requirements

Hadoop Cluster	• HDP Version 2.5 • Hadoop-2.7.3+hdp2.5+844 • Hive-1.2.1+hdp2.5+350 • Sqoop1 V 1.4.4+hdp2.5+67 • Oracle Loader For Hadoop (OLH) V 3.2 • Hive JDBC Connectors V 2.5.15 • Spark 2.x
ETL Namematching Server	RHEL 7.4+
BIG DATA	
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

Prerequisite Environmental Settings

The prerequisite environmental settings to be performed before beginning the installation of Studio are as follows:

Table 2–2 Prerequisite Information

Category	Expected Value
PGX Settings	PGX version 19.4.x 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. • `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 2–2 (Cont.) Prerequisite Information

Category	Expected Value
ETL Services	The following packages must be installed or present in the server where ETL services are deployed: • `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` For information on installing ETL Services, see Installing ETL Services.
Oracle Database Settings	**Note**: This setting is required only if the Wallet has to be created on the same server as that of the 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 download 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 installation 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` To check the locale installed, execute the following command: `locale -a | grep -i 'en_US.utf'` The installed 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 2–2 (Cont.) Prerequisite Information

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 Schema User>; GRANT execute on dbms_rls TO <BD Schema User>; GRANT ALL privileges TO <Studio Schema User>;
Wallet Settings	Set a password store with Oracle Wallet. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".

Performing Common Pre-installation Tasks

The common pre-installation tasks that you must perform before installing Studio are as follows:

- Obtaining the Software
- Extracting the Software

Obtaining the Software

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

1. Login to My Oracle Support with a valid Oracle account and search for the Patch ID **30868218** under the *Patches & Updates* tab.
2. Download the installer archive `OFS_FCCM_STUDIO_8.0.7.2.0_LINUX.zip` file to the download directory (in Binary Mode) on the setup identified for Studio installation.

Extracting the Software

1. Extract the contents of the `OFS_FCCM_STUDIO_8.0.7.2.0_Linux.zip` installer archive file in the download directory using the following command:

   ```
   unzip -a OFS_FCCM_STUDIO_8.0.7.2.0_LINUX.zip
   ```

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

Note:

Do not rename the Application installer folder name on 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
   ```

Required File Structure

The Studio application must be installed with certain additional services such as Cross Language Name Matching, ETL, and Hadoop/Spark services.

To install the additional services, you must obtain the required configuration files as follows from the Big Data installation path.

> **Note:**
>
> These files must be kept ready and provided in the following file structure which is used during Studio installation.

- Hadoop Cluster
  - `core-site.xml`
  - `hadoop-env.sh`
  - `hdfs-site.xml`
  - `hive-env.sh`
  - `hive-site.xml`
  - `log4j.properties`
  - `mapred-site.xml`
  - `redaction-rules.json`
  - `spark-defaults.conf`
  - `spark-env.sh`
  - `ssl-client.xml`
  - `topology.map`
  - `topology.py`
  - `yarn-site.xml`
- Kerberos Files
  - `krb5.conf`
  - `ofsaa.keytab`

  > **Note**:
  >
  > Rename your `.keytab` file to `ofsaa.keytab`.

- Certificates
  - `key.store.jks`
  - `trusted.store.jks`

- 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 existing jars of Cloudera installation.
- The `HiveJDBC4.jar` file is not available in the Cloudera setup. You must download the same from the Cloudera website.

Interpreter Settings

Note:

Perform the following pre-requisite settings only for the interpreters that you need.

Table 2–3 Interpreter Settings

Interpreter	Prerequisite Settings
fcc-jdbc	No additional configuration is required. **Note:** A new interpreter called vanilla jdbc interpreter is provided, which will connect to the Studio schema. Studio with non-BD can use this interpreter instead of the fcc-jdbc interpreter.
fcc-ore	For the required configuration, see Appendix D, "Installing RServe Manually"
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	Install the py4j package.
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.

Table 2–3 Interpreter Settings

Interpreter	Prerequisite Settings
pyspark	For the required configuration, see Configuring PySpark Interpreter.
spark	For the required configuration, see Configuring Spark Interpreter

Configuring PySpark Interpreter

Prerequisites

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

Configuration

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

- In the ***Spark Interpreter Settings*** page of the 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 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.

To ensure that the two Python versions match, in the case where your components run on different machines, see Appendix G, "Using Python Virtual Environments with PySpark".

Configuring Spark Interpreter

Prerequisites

To operate the Spark interpreter in local mode or Yarn mode, the `spark.master` property must be set accordingly. For information on setting the Spark Master properties, see Appendix F, "Providing Spark Libraries".

Local Mode

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

Yarn Mode

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

- Provide custom Spark libraries. For more information, see Appendix F, "Providing Spark Libraries".

- The cluster's Hadoop client-side configuration files that include XML files such as `yarn-site.xml` are required and must be supplied with the Spark libraries. These files are available in the Hadoop configuration directory, `HADOOP_CONF_DIR` of the cluster or can be downloaded from the cluster manager's UI if you are using a Cloudera cluster.

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 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 can be found in the Spark's 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 Studio application UI.

3

Installing Studio With BD

This chapter provides the necessary information to install the Crime and Compliance Studio application. It includes the following topics:

1. Installing the Studio Application With BD
2. Verifying the Installation
3. Installing ETL Services
4. Installing PGX
5. Launching the FCC Studio Application

Installing the Studio Application With BD

To install the Studio application with BD, follow these steps:

1. Placing Required Files
2. Configuring the config.sh File
3. Running the Installer

Placing Required Files

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

1. Place the additional jar files.
    1. Navigate to the `<Studio_Installation_Path>/batchservice/user/lib` directory.
    2. 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 existing jars of Cloudera installation.
- The `HiveJDBC4.jar` file is not available in the Cloudera setup. You must download the same from the Cloudera website.

2. Place the Kerberos Files.
    1. Navigate to the `<Studio_Installation_Path>/batchservice/user/conf` directory.
    2. Place the following Kerberos files:
        - `krb5.conf`
        - `ofsaa.keytab`

Configuring the config.sh File

To configure the `config.sh` file, 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 mentioned in Table 3–1, " config.sh Parameters".

Note:

- You must manually set the InteractionVariable parameter value. 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 enter the correct hostname for URL of ETL and PGX services in the `config.sh` file.
- Do not alter the parameter values that are already set in the `config.sh` file

---

Table 3–1 config.sh Parameters

InteractionVariable Name	Significance	Used for Hive DataBase	Mandatory
FCC_STUDIO_ INSTALLATION_PATH	Indicates the path where Studio is extracted.	Yes	Yes
NON_OFSAA	Enter "false" to install Studio application with BD.	Yes	Yes
REALM	For example: `com.oracle.ofss.fccm.studio.datastudio.auth.FCCMRealm`	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 OFSAAAI, the value must be set in the following format: `https://<HostName>:<PortNo>/<ContextName>/rest-api`	Yes	Yes
PGX_SERVER_URL	Indicates the URL of the PGX server. Example: `http://<HostName>:<PortNo>/` The value for PortNo must be 7007.	Yes	Yes

Table 3–1 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Used for Hive DataBase	Mandatory
LIVY_HOST_URL	Indicates the URL of the Livy application. The format for the URL is as follows: `http://<HostName>:<PortNo>`	Yes	Yes
Internal Services			
AUTH_SERVICE_URL	Indicates the AUTH service URL which gets activated after the `fcc-studio.sh` file runs. The format for the Auth service URL is as follows: `http://<HostName>:7041/authservice`	Yes	Yes
BATCH_SERVICE_URL	Indicates the batch service URL which gets activated after the `fcc-studio.sh` file runs. The format for the batch service URL is as follows: `http://<HostName>:7043/batchservice`	Yes	Yes
META_SERVICE_URL	Indicates the metaservice URL which gets activated after the `fcc-studio.sh` file runs. The format for the metaservice URL is as follows: `http://<HostName>:7045/metaservice`	Yes	Yes
SESSION_SERVICE_URL	Indicates the session service URL which gets activated after the `fcc-studio.sh` file runs. The format for the session service URL is as follows: `http://<HostName>:7047/sessionservice`	Yes	Yes
ETL_SERVICE_URL	Indicates the ETL Service URL.	Yes	Yes
FCC_STUDIO_URL	Indicates the Studio URL. The format for the FCC Studio URL is as follows: `http://<HostName>:7008`	Yes	Yes
ORE Interpreter Settings			
RSERVE_USERNAME	Indicates the RServe username.	No	No
RSERVE_PASSWORD	Indicates the RServe password.	No	No
HTTP_PROXY	Indicates the HTTP proxy.	No	No
HTTPS_PROXY	Indicates the HTTPS proxy.	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
USERS_LIB_PATH	Indicates the path where the R packages are installed.	No	No
RSERVE_CONF_PATH	Indicates the path where the `Rserve.conf` file is present.	No	No
DB Details for Studio Schema			
STUDIO_DB_HOSTNAME	Indicates the hostname of the database where Studio schema is created.	Yes	Yes
STUDIO_DB_PORT	Indicates the port number where Studio schema is created.	Yes	Yes

Table 3–1 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Used for Hive DataBase	Mandatory
STUDIO_DB_SERVICE_NAME	Indicates the service name of the database where Studio schema is created.	Yes	Yes
STUDIO_DB_SID	Indicates the SID of the database where Studio schema is created.	Yes	Yes
STUDIO_DB_USERNAME	Indicates the username of the Studio Schema (newly created Oracle Schema).	Yes	Yes
STUDIO_DB_PASSWORD	Indicates the password for the newly created schema.	Yes	Yes
Studio DB Wallet Details			
STUDIO_ALIAS_NAME	Indicates the Studio alias name. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes	Yes
STUDIO_WALLET_LOCATION	Indicates the Studio wallet location. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes	Yes
STUDIO_TNS_ADMIN_PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `STUDIO_ALIAS_NAME` is present. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes	Yes
DB Details for BD Config Schema			
BD_CONFIG_HOSTNAME	Indicates the hostname of the database where BD config schema is installed.	Yes	Yes
BD_CONFIG_PORT	Indicates the port of the database where BD config schema is installed.	Yes	Yes
BD_CONFIG_SERVICE_NAME	Indicates the service name of the database where BD config schema is installed.	Yes	Yes
BD_CONFIG_SID	Indicates the SID of the database where BD config schema is installed.	Yes	Yes
BD_CONFIG_USERNAME	Indicates the username for the BD config schema.	Yes	Yes
BD_CONFIG_PASSWORD	Indicates the password for the BD config schema.	Yes	Yes
BD Config Wallet Details			
BD_CONFIG_ALIAS_NAME	Indicates the BD config alias name. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes	Yes
BD_CONFIG_WALLET_LOCATION	Indicates the BD config wallet location. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes	Yes
BD_CONFIG_TNS_ADMIN_PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `BD_CONFIG_ALIAS_NAME` is present. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes	Yes
DB Details for BD Atomic Schema			
BD_ATOMIC_HOSTNAME	Indicates the BD atomic schema hostname.	Yes	Yes
BD_ATOMIC_PORT	Indicates the BD atomic schema port number.	Yes	Yes
BD_ATOMIC_SERVICE_NAME	Indicates the BD atomic schema service name.	Yes	Yes

Table 3–1 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Used for Hive DataBase	Mandatory
BD_ATOMIC_SID	Indicates the BD atomic schema SID.	Yes	Yes
BD_ATOMIC_USERNAME	Indicates the username of the BD atomic schema.	Yes	Yes
BD_ATOMIC_PASSWORD	Indicates the password of the BD atomic schema.	Yes	Yes
BD Atomic Wallet Details			
BD_ATOMIC_ALIAS_NAME	Indicates the BD atomic alias name. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes	Yes
BD_ATOMIC_WALLET_LOCATION	Indicates the BD atomic wallet location. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes	Yes
BD_ATOMIC_TNS_ADMIN_PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `BD_ATOMIC_ALIAS_NAME` is present. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes	Yes
SQL Scripts			
FSINFODOM	Indicates the name of the BD Infodom.	Yes	Yes
FSSEGMENT	Indicates the name of the BD segment.	Yes	Yes
DATAMOVEMENT_LINK_NAME	• If the newly created 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 <SCHEMA_NAME> in this parameter. • If the newly created schema is in the same database host, the value for this parameter is the user name of the BD atomic schema.	Yes	Yes
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
PGX Setup Details			
PGX_INSTALATION_PATH	Indicates the installation path of the PGX server. Example: `<STUDIO_INSTALLATION_PATH>/pgx/server/`	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

Cloudera Setup Details			
HADOOP_CREDENTIAL_PROVIDER_PATH	Indicates the path where Hadoop credential is stored. **Note**: Indicates the Hadoop alias given when creating the Hadoop credentials. For information on how to create a credential keystore, see Creating Credential Keystore.	Yes	Yes

Table 3–1 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Used for Hive DataBase	Mandatory
HADOOP_PASSWORD_ ALIAS	Indicates the Hadoop alias given when creating the Hadoop credentials.	Yes	Yes
Hive_Host_Name	Indicates the Hive hostname.	Yes	Yes
Hive_Port_number	Indicates the Hive port number. Contact System Administrator to obtain the port number.	Yes	Yes
HIVE_PRINCIPAL	Indicates the Hive Principal. Contact System Administrator to obtain HIVE_PRINCIPAL.	Yes	Yes
HIVE_SCHEMA	Indicates to create a schema in HIVE.	Yes	Yes
JAAS_CONF_FILE_ PATH	Created for future use.	No	No
Krb_Host_FQDN_Name	Indicates the Krb host FQDN name.	Yes	Yes
Krb_Realm_Name	Indicates the Krb realm name.	Yes	Yes
Krb_Service_Name	Indicates the Krb service name. Example: Hive	Yes	Yes
KRB5_CONF_FILE_ PATH	Created for future use.	No	No
security_krb5_kdc_server	Created for future use.	No	No
security_krb5_realm	Created for future use.	No	No
server_kerberos_keytab_ file	Created for future use.	No	Yes
server_kerberos_principal	Created for future use.	No	Yes
SQOOP_ HOSTMACHINE_ USER_NAME	Indicates the user name of the Host machine where sqoop will run.	Yes	Yes
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 in the `SQOOP_PARAMFILE_PATH` parameter. Example: `/scratch/ofsaa/` **Note**: Ensure that the location name ends with a '/'.	Yes	Yes
SQOOP_PARTITION_ COL	Indicates the column in which the HIVE table is partitioned. The value must be `SNAPSHOT_DT`.	Yes	Yes
SQOOP_TRG_ HOSTNAME	Indicates the hostname of the SQOOP server. Example: `<HostName>`	Yes	Yes
SQOOP_WORKDIR_ HDFS	Indicates the SQOOP working directory in HDFS. Example: `/user/ofsaa`	Yes	Yes

Running the Installer

To run the 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 `install.sh` command does not generate any log file.

3. Execute the following command in the console:

   ```
   ./fcc_studio.sh
   ```

   The OFS Crime and Compliance Studio application is installed with BD. The Data Studio and all the interpreters are started.

   After the successful completion of Studio, the script displays a URL. You can use this URL to launch the FCC Studio Application. For more information, see Launching the FCC Studio Application.

4. Verify the Installation. See Verifying the Installation

Verifying the Installation

To verify the installation:

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

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

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

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

Installing ETL Services

Installing the ETL service will also install the Cross Language Name Matching service.

To install ETL Services, follow these steps:

1. Configuring ETL Services
2. Configuring FILEPATH for ICIJ

Configuring ETL Services

Note:

- ETL can be installed on the same server where Studio is installed or on a different server.
- The following packages must be installed or present in the server where ETL services are 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
  ```

To install ETL, follow these steps:

1. Navigate to the <STUDIO_INSTALLATION_PATH>/etl/bin/ directory.
2. Perform the following:
   - If ETL is to be installed on the same server where Studio is installed, unzip the OFS_FCCM_STUDIO_ETL-8.0.7.2.0.zip file.
   - If ETL is to be installed on a different server, follow these steps:
   1. Copy the OFS_FCCM_STUDIO_ETL-8.0.7.2.0.zip file to the ETL server.
   2. Unzip the OFS_FCCM_STUDIO_ETL-8.0.7.2.0.zip file.

   Note: The path where the OFS_FCCM_STUDIO_ETL-8.0.7.2.0.zip file is unzipped is referred to as <ETL_INSTALLATION_PATH>.
3. Navigate to <ETL_INSTALLATION_PATH>/etl/bin directory.
4. Configure the config.sh file as described in Table 3–2, "config.sh for ETL Services".

Note: Do not alter the parameter values that are already set in the config.sh file.

Table 3–2 config.sh for ETL Services

Interaction Variable Name	Significance
ETL Services	
CROSSLANGNAMEMATCH_SERVICE_URL	Indicates the URL for the Cross language Name Matching service. For example: http://<HostName>:7023
ETL_SERVICE_URL	Indicates the URL for the ETL service. For example: http://<HostName>:7024
ETL_DRIVER_HOST	Indicates the IP address of the ETL Initiation machine host.

Table 3–2 config.sh for ETL Services

Interaction Variable Name	Significance
ETL_SERVICE_PORT	The port that should be used by the ETL service and Spark driver. For example: 30724.
ETL_DRIVER_ BLOCKMANAGER_PORT	The port that should be used by the Spark driver blockmanager. For example: 30726.
HIVE_SCHEMA	Indicates the Hive schema as configured in the `<Studio_ Installation_Path>/bin/config.sh` file.
ETL_DRIVER_CORES	Indicates the number of cores present on the ETL Initiation machine host (master of CDH server) that is present/accessible on the server where ETL services are deployed.
ETL_DRIVER_MEMORY	Indicates how much memory is to be assigned to the ETL service. **Note**: Ensure that the memory is slightly less than that of the ETL Initiation machine host. For example: 90g.
ETL_EXECUTOR_ INSTANCES	The number of Spark executor instances that should be used on the big data cluster during ETL.
ETL_EXECUTOR_CORES	The number of cores that should be used by each Spark executor instance on the big data cluster during ETL.
ETL_EXECUTOR_MEMORY	The amount of memory that should be used by each Spark executor instance on the big data cluster during ETL.
URL_GLOBAL_GRAPH_ NODES_CSV	Indicates the HDFS URL where the CSV file of the global graph is stored at the end of the ETL. It can either be a local or hdfs path. For example: `hdfs:///user/ofsaa/ETL_Directory/global_ graph_nodes.csv` **Note**: Ensure you have already created the `ETL_Directory` manually and have provided 775 permission. This directory is used to store the CSV file at the end of the ETL.
URL_GLOBAL_GRAPH_ EDGES_CSV	Indicates the HDFS URL where the CSV file of the global graph is stored at the end of the ETL. It can either be a local or hdfs path. For example: hdfs:///user/ofsaa/STUDIO_ETL/global_graph_edges.csv **Note**: Ensure this location is already created and available to store the CSV file at the end of the ETL.
URL_GLOBAL_GRAPH_ CONFIG_JSON	Indicates the HDFS URL where the PGX graph configuration .json file is stored at the end of the ETL. The location can be either local or hdfs path. For example: `hdfs:///user/ofsaa/STUDIO_ETL/config.json` **Note**: Ensure this location is already created and available to store the JSON file at the end of the ETL. If you do not want a graph configuration file written, provide the value as follows: `null://EMPTY`
URL_GLOBAL_GRAPH_ CONFIG_TEMPLATE_JSON	Indicates the information about the `output.config.template.location` parameter in the `etl.properties` file. **Note**: Ensure this location is already created and available to store the JSON file at the end of the ETL.

Table 3–2 config.sh for ETL Services

Interaction Variable Name	Significance
URL_NAMES_CSV	Indicates the HDFS URL where the names CSV file is updated at the end of the ETL. It can either be a local or hdfs path. For example: `hdfs:///user/ofsaa/STUDIO_ETL/name_index.csv` **Note**: Ensure this location is already created and available, and the CSV file is already created and placed in this location. The CSV file values are replaced with the values in the new CSV file created at the end of the ETL. For information on creating the CSV files, see Appendix E, "Creating Required Index Files".
URL_ADDRESS_CSV	Indicates the HDFS URL where the addresses CSV file is updated at the end of the ETL. It can either be a local or hdfs path. For example: `hdfs:///user/ofsaa/STUDIO_ETL/address_index.csv` **Note**: Ensure this location is already created and available, and the CSV file is already created and placed in this location. The CSV file values are replaced with the values in the new CSV file created at the end of the ETL. For information on creating the CSV Files, see Appendix E, "Creating Required Index Files".
Cloudera Setup Details	
KERBEROS_KEYTAB_FILENAME	Indicates the information about `server_kerberos_keytab_file` from studio `config.sh` file.
KERBEROS_PRINCIPAL	Indicates the information about `server_kerberos_principal` from studio `config.sh` file.
KRB5_CONFIG_FILENAME	Indicates the file name of the Kerberos config file. For example: krb5.conf.
ETL Configuration	
KERBEROS_TICKET_RENEWAL_PERIOD	Indicates the interval on which Kerberos ticket is renewed.

5. Navigate to the `<ETL_Installation_Path>/conf/` directory and replace the files present inside the following folders with the required files as per the file structure mentioned in Required File Structure.
   - Hadoop Cluster
   - Certificates
   - Kerberos
6. Navigate to the `<ETL_INSTALLATION_PATH>/bin` directory.
7. Execute the following command in the console:

   ```
   ./install.sh
   ```

   Note: Execution of `install.sh` command does not generate any log file.

   The values configured in the `config.sh` file will auto-populate the values in the following property files for ETL and Cross Language Name Matching services respectively:

   - `<ETL_Installation_Path>/etl/conf/etl.properties`

- `<ETL_Installation_Path>/crosslangnamematch/conf/NameMatchingLocations.properties`

For more information on the `etl.properties` and `NameMatchingLocations.properties` files, see the *Studio Services* chapter in the *OFS Crime and Compliance Studio Administration Guide*.

8. (Optional) Add a new data source. For more information, see *Configuring Data Sources for Graph* chapter in the *OFS Crime and Compliance Studio Administration Guide*.
9. Start the ETL Services. For more information see Starting/Stopping ETL Service.

Configuring FILEPATH for ICIJ

Note: The FCC Studio graph model is configured to include ICIJ watch list files.

To configure FILEPATH for ICIJ, follow these steps:

- If watch list files are present, follow these steps:
  1. Place the watch list file in HDFS, which is accessible by the user.
  2. Update the `FILEPATH` of the watch list files in the `fcc_studio_etl_files` table.

Figure 3–1 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

- If watch list files are absent, follow these steps:
  1. Edit the `<ETL_Installation_Path>/etl/conf/etl.properties` file as follows:

     Change the following:

     ```
     connectors=paradise;bahama;offshore;panama;fcdm
     ```

     to

     ```
     connectors=fcdm
     ```

Note: Ignore the properties that start with parameter values like `bahama`, `offshore`, `paradise`, and `panama`.

Installing PGX

Note: PGX can be installed on the same server where Studio is installed or on a different server.

To install PGX, follow these steps:

1. Navigate to the `<Studio_Installation_Path>/pgx/server/` directory.
2. Perform the following:

- If PGX is to be installed on the same server where Studio is installed, unzip the `pgx-distribution-19.4.x-server.zip` file.
- If PGX is to be installed on a different server, follow these steps:

1. Copy the `pgx-distribution-19.4.x-server.zip` file to the PGX server.
2. Unzip the `pgx-distribution-19.4.x-server.zip` file.

Note: The path where the `pgx-distribution-19.4.x-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:
   - In the `server.conf` file, set the following:

     ```
     enable_tls: false,

     enable_client_authentication: false
     ```

   - In the `pgx.conf` file, set the following:

     ```
     allow_local_filesystem: true
     ```

     Here, true indicates to enable, and false indicates to disable.
5. Copy and paste the following Kerberos Files to the `<PGX_Installation_Path>/pgx/server/conf/kerberos` directory:
   - `krb5.conf`
   - `ofsaa.keytab`
6. Copy and paste the following Hadoop configuration files into 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 3–3 config.sh Parameters

InteractionVariable Name	Significance
KERBEROS_TICKET_RENEWAL_PERIOD	For example: 7200 would mean every 2 hours
KERBEROS_PRINCIPAL	For example: ofsaa@WHFAYK.ORACLE.COM

Table 3–3 (Cont.) config.sh Parameters

InteractionVariable Name	Significance
KERBEROS_KEYTAB_ FILENAME	For example: ofsaa.keytab
KRB5_CONFIG_ FILENAME	For example: krb5.conf
PGX_SERVER_OFF_ HEAP_MB	For example: 10240
PGX_SERVER_ON_ HEAP_MB	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/ofsaa/hive_etl_out_80702_pgx/config.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 commands:
   1. `./config.sh`
   2. `./install.sh`
9. Start the PGX service. For more information, see Starting/Stopping PGX Service.

Note: PGX must be started only after the ETL configuration is completed and the PGX graph configuration .json file located in the `URL_GLOBAL_GRAPH_CONFIG_JSON` path is present.

Launching the FCC Studio Application

1. Enter the URL obtained after successful installation of the Studio application in the following format into the browser:

   `https://<Master_Node>:7008`

   The OFS Crime and Compliance Studio application is launched.

4

Post-installation Configuration for Studio Installed With BD

On successful installation of Studio application with BD, perform the following post-installation configuration:

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

Note: Before running the post-installation steps, an SSH connection to the Big Data server must be configured.

Configuring SSH Connection

To configure SSH connection, follow these steps:

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

       `Generating public/private rsa key pair`
    2. `Enter file in which to save the key (<Linux_Home>/.ssh/id_rsa):` [Press Enter]
    3. `Enter passphrase (empty for no passphrase):` [Press Enter]
    4. `Enter same passphrase again:` [Press Enter]
    5. `ssh-copy-id -i ~/.ssh/id_rsa.pub <BigData_Server>`
    6. `ssh <BigData_Server>`

Configuring Interpreters

After starting the Studio application, the configuration for the interpreters can be performed from the user interface (UI). For information on configuring interpreters, see the *Configuring Interpreters* chapter in the *Oracle Financial Services Crime and Compliance Studio Administration Guide*.

Performing OFSAA Configuration for Batch Execution

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 pack is installed and to the `$FIC_DB_HOME/bin` pat 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 to the `$FIC_DB_HOME/lib` directory.

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

Performing Hive Data Movement

To perform Hive data movement, follow these steps:

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

Configuring Schema Creation

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

Configuring Schema Creation from Studio Server

To configure Schema Creation from 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 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
   ```

   This creates tables in the Hive Schema.

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

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 to the `<OFSAA_FIC_HOME_PATH>/ficdb/lib` path.

2. Copy all the .sh files located in the <Studio_Installation_Path>/ficdb/bin path to 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 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
   ```

   This creates tables in the Hive Schema.
5. Check the <Studio_Installaton_Path>/logs/batchservice.logs for more information.

Creating Credential Keystore

To create a credential keystore, follow these steps:

1. Login as HDFS SuperUser.
2. Create a credential keystore on HDFS by executing the following command:

   ```
   hadoop credential create mydb.password.alias -provider
   jceks://hdfs/user/root/oracle.password.jceks
   ```
3. Verify the credential keystore file by executing the following command:

   ```
   hadoop credential list -provider
   jceks://hdfs/user/root/oracle.password.jceks
   ```
4. Grant Read permission to the keystore file by executing the following command:

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

Note: Ensure the credential keystore file path and the alias are correctly provided in the config.sh file.

Configuring Data Movement and Graph Load

Note: The Big Data System Administrator must place the batchservice-8.0.7.2.0.jar file in all nodes of the Spark cluster. Ensure that the path of the jar file is present in 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 to 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*.

Performing Configuration for Running Scenario Notebooks

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

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

   For information on performing Data Movement and Graph Load, see the *Executing Published Scenario Notebook* section in the *OFS Crime and Compliance Studio Administration Guide*.

5

Upgrading Studio With BD

This chapter provides the information required to understand the upgrade of the Studio application with BD.

Upgrade Overview

This release (v8.0.7.2.0) of the OFS Crime and Compliance Studio application pack can be used for the following:

- Install a new instance of the Studio application (v8.0.7.2.0).
- Upgrade an existing instance of the Studio application (v8.0.7.x) to the v8.0.7.2.0 version.

> **Note:** During the upgrade, ensure to provide the same BD database, Studio schema, Hive schema, wallet related information that you used during the installation of the existing instance of the Studio application.

This section provides quick start steps to upgrade an existing instance of the Studio application (v8.0.7.x) with BD to v8.0.7.2.0 version.

Table 5–1 Quick Start Steps to Upgrade Studio with BD

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 Common Pre-installation Tasks
		  1. Obtaining the Software
		  2. Extracting the Software
		2. Required File Structure
		3. Interpreter Settings

Table 5–1 Quick Start Steps to Upgrade Studio with BD

Sl. No.	Steps	Reference Links

2.	Install the Studio application with BD	1. Placing Required Files
		2. Configuring the config.sh File for Upgrade
		3. Running the Installer
		4. Verifying the Installation
		5. Installing ETL Services
		  1. Configuring ETL Services
		  2. Configuring FILEPATH for ICIJ
		6. Installing PGX
		7. Launching the FCC Studio Application

3	Post-installation configuration	• Configuring SSH Connection • Configuring Interpreters • Performing OFSAA Configuration for Batch Execution • Performing Hive Data Movement

Configuring the config.sh File for Upgrade

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

1. Login to the server as a non-root user.
2. Navigate to the `<Studio_Installation_Path>/bin/config.sh` file.
3. Configure the `config.sh` file as mentioned in Table 5–2, " config.sh Parameters".

Note:

- You must manually set the InteractionVariable parameter value. 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 enter the correct hostname for URL of ETL and PGX services in the `config.sh` file.
- Do not alter the parameter values that are already set in the `config.sh` file
- During the upgrade, ensure to provide the same BD database, Studio schema, Hive schema, and wallet related information in the `config.sh` file, that you used during the installation of the existing instance of the Studio application.

Table 5–2 config.sh Parameters

InteractionVariable Name	Significance	Mandatory
FCC_STUDIO_INSTALLATION_PATH	Indicates the path where Studio is extracted.	Yes

Table 5–2 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Mandatory
NON_OFSAA	Enter "false" to install Studio application with BD.	Yes
REALM	For example: `com.oracle.ofss.fccm.studio.datastudio.auth.FCCMRealm`	Yes
External Services		
OFSAA_SERVICE_URL	Indicates the URL of the OFSAA instance. Do not enter '/' at the end of the URL. **Note:** • For OFSAAAI, the value must be set in the following format: `https://<HostName>:<PortNo>/<ContextName>/rest-api`	Yes
PGX_SERVER_URL	Indicates the URL of the PGX server. Example: `http://<HostName>:<PortNo>/` The value for PortNo must be 7007.	Yes
LIVY_HOST_URL	Indicates the URL of the Livy application. The format for the URL is as follows: `http://<HostName>:<PortNo>`	Yes
Internal Services		
AUTH_SERVICE_URL	Indicates the AUTH service URL which gets activated after the `fcc-studio.sh` file runs. The format for the Auth service URL is as follows: `http://<HostName>:7041/authservice`	Yes
BATCH_SERVICE_URL	Indicates the batch service URL which gets activated after the `fcc-studio.sh` file runs. The format for the batch service URL is as follows: `http://<HostName>:7043/batchservice`	Yes
META_SERVICE_URL	Indicates the metaservice URL which gets activated after the `fcc-studio.sh` file runs. The format for the metaservice URL is as follows: `http://<HostName>:7045/metaservice`	Yes
SESSION_SERVICE_URL	Indicates the session service URL which gets activated after the `fcc-studio.sh` file runs. The format for the session service URL is as follows: `http://<HostName>:7047/sessionservice`	Yes
ETL_SERVICE_URL	Indicates the ETL Service URL.	Yes
FCC_STUDIO_URL	Indicates the Studio URL. The format for the FCC Studio URL is as follows: `http://<HostName>:7008`	Yes
ORE Interpreter Settings		
RSERVE_USERNAME	Indicates the RServe username.	No

Table 5–2 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Mandatory
RSERVE_PASSWORD	Indicates the RServe password.	No
HTTP_PROXY	Indicates the HTTP proxy.	No
HTTPS_PROXY	Indicates the HTTPS proxy.	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
USERS_LIB_PATH	Indicates the path where the R packages are installed.	No
RSERVE_CONF_PATH	Indicates the path where the `Rserve.conf` file is present.	No
DB Details for Studio Schema		
STUDIO_DB_HOSTNAME	Indicates the hostname of the database where Studio schema is created.	Yes
STUDIO_DB_PORT	Indicates the port number where Studio schema is created.	Yes
STUDIO_DB_SERVICE_NAME	Indicates the service name of the database where Studio schema is created.	Yes
STUDIO_DB_SID	Indicates the SID of the database where Studio schema is created.	Yes
STUDIO_DB_USERNAME	Indicates the username of the Studio Schema (newly created Oracle Schema).	Yes
STUDIO_DB_PASSWORD	Indicates the password for the newly created schema.	Yes
Studio DB Wallet Details		
STUDIO_ALIAS_NAME	Indicates the Studio alias name. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes
STUDIO_WALLET_LOCATION	Indicates the Studio wallet location. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes
STUDIO_TNS_ADMIN_PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `STUDIO_ALIAS_NAME` is present. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes
DB Details for BD Config Schema		
BD_CONFIG_HOSTNAME	Indicates the hostname of the database where BD config schema is installed.	Yes
BD_CONFIG_PORT	Indicates the port of the database where BD config schema is installed.	Yes
BD_CONFIG_SERVICE_NAME	Indicates the service name of the database where BD config schema is installed.	Yes
BD_CONFIG_SID	Indicates the SID of the database where BD config schema is installed.	Yes
BD_CONFIG_USERNAME	Indicates the username for the BD config schema.	Yes
BD_CONFIG_PASSWORD	Indicates the password for the BD config schema.	Yes
BD Config Wallet Details		
BD_CONFIG_ALIAS_NAME	Indicates the BD config alias name. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes

Table 5–2 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Mandatory
BD_CONFIG_WALLET_LOCATION	Indicates the BD config wallet location. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes
BD_CONFIG_TNS_ADMIN_PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `BD_CONFIG_ALIAS_NAME` is present. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes
DB Details for BD Atomic Schema		
BD_ATOMIC_HOSTNAME	Indicates the BD atomic schema hostname.	Yes
BD_ATOMIC_PORT	Indicates the BD atomic schema port number.	Yes
BD_ATOMIC_SERVICE_NAME	Indicates the BD atomic schema service name.	Yes
BD_ATOMIC_SID	Indicates the BD atomic schema SID.	Yes
BD_ATOMIC_USERNAME	Indicates the username of the BD atomic schema.	Yes
BD_ATOMIC_PASSWORD	Indicates the password of the BD atomic schema.	Yes
BD Atomic Wallet Details		
BD_ATOMIC_ALIAS_NAME	Indicates the BD atomic alias name. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes
BD_ATOMIC_WALLET_LOCATION	Indicates the BD atomic wallet location. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes
BD_ATOMIC_TNS_ADMIN_PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `BD_ATOMIC_ALIAS_NAME` is present. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes
SQL Scripts		
FSINFODOM	Indicates the name of the BD Infodom.	Yes
FSSEGMENT	Indicates the name of the BD segment.	Yes
DATAMOVEMENT_LINK_NAME	• If the newly created 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 <SCHEMA_NAME> in this parameter. • If the newly created schema is in the same database host, the value for this parameter is the user name of the BD atomic schema.	Yes
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
PGX Setup Details		
PGX_INSTALATION_PATH	Indicates the installation path of the PGX server. Example: `<STUDIO_INSTALLATION_PATH>/pgx/server/`	Yes

Table 5–2 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Mandatory

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

Cloudera Setup Details		
HADOOP_CREDENTIAL_PROVIDER_PATH	Indicates the path where Hadoop credential is stored. **Note**: Indicates the Hadoop alias given when creating the Hadoop credentials. For information on how to create a credential keystore, see Creating Credential Keystore.	Yes
HADOOP_PASSWORD_ALIAS	Indicates the Hadoop alias given when creating the Hadoop credentials.	Yes
Hive_Host_Name	Indicates the Hive hostname.	Yes
Hive_Port_number	Indicates the Hive port number. Contact System Administrator to obtain the port number.	Yes
HIVE_PRINCIPAL	Indicates the Hive Principal. Contact System Administrator to obtain HIVE_PRINCIPAL.	Yes
HIVE_SCHEMA	Indicates to create a schema in HIVE.	Yes
JAAS_CONF_FILE_PATH	Created for future use.	No
Krb_Host_FQDN_Name	Indicates the Krb host FQDN name.	Yes
Krb_Realm_Name	Indicates the Krb realm name.	Yes
Krb_Service_Name	Indicates the Krb service name. Example: Hive	Yes
KRB5_CONF_FILE_PATH	Created for future use.	No
security_krb5_kdc_server	Created for future use.	No
security_krb5_realm	Created for future use.	No
server_kerberos_keytab_file	Created for future use.	Yes
server_kerberos_principal	Created for future use.	Yes
SQOOP_HOSTMACHINE_USER_NAME	Indicates the user name of the Host machine where sqoop will run.	Yes

Table 5–2 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	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 in the `SQOOP_PARAMFILE_PATH` parameter. Example: `/scratch/ofsaa/` **Note**: Ensure that the location name ends with a '/'.	Yes
SQOOP_PARTITION_COL	Indicates the column in which the HIVE table is partitioned. The value must be `SNAPSHOT_DT`.	Yes
SQOOP_TRG_HOSTNAME	Indicates the hostname of the SQOOP server. Example: `<HostName>`	Yes
SQOOP_WORKDIR_HDFS	Indicates the SQOOP working directory in HDFS. Example: `/user/ofsaa`	Yes

6

Installing Studio Without BD

This chapter provides the necessary information to install the Crime and Compliance Studio application without BD. It includes the following topics:

1. Installing the Studio Application Without BD
2. Verifying the Installation
3. Launching the FCC Studio Application

Installing the Studio Application Without BD

To install the Studio application, follow these steps:

1. Configuring the config.sh File
2. Running the Installer

Configuring the config.sh File

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

1. Login to the server as a non-root user.
2. Navigate to the `<Studio_Installation_Path>/bin/config.sh` directory.
3. Configure the `config.sh` file as mentioned in Table 6–1, " config.sh Parameters".

   You must manually set the InteractionVariable parameter values. If a value is not applicable, enter NA and ensure that the value is not entered as NULL.

Note:

- You must manually set the InteractionVariable parameter value. 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 enter the correct hostname for URL of ETL and PGX services in the `config.sh` file.
- Do not alter the parameter values that are already set in the `config.sh` file
- Retain the existing placeholder values for the non-mandatory parameters.

Table 6–1 config.sh Parameters

InteractionVariable Name	Significance	Mandatory
FCC_STUDIO_ INSTALLATION_PATH	Indicates the path where Studio is extracted.	Yes
NON_OFSAA	Enter "true" to install Studio application without BD.	Yes
REALM	For example: `com.oracle.ofss.fccm.studio.datastudio.auth.DemoRealm`	Yes
External Services		
OFSAA_SERVICE_URL	Indicates the URL of the OFSAA instance. Do not enter '/' at the end of the URL. **Note:** • For OFSAAAI, the value must be set in the following format: `https://<HostName>:<PortNo>/<ContextName>/rest-api`	No
PGX_SERVER_URL	Indicates the URL of the PGX server. Example: `http://<HostName>:<PortNo>/` The value for PortNo must be 7007.	Yes
LIVY_HOST_URL	Indicates the URL of the Livy application. The format for the URL is as follows: `http://<HostName>:<PortNo>`	Yes
Internal Services		
AUTH_SERVICE_URL	Indicates the AUTH service URL which gets activated after the `fcc-studio.sh` file runs. The format for the Auth service URL is as follows: `http://<HostName>:7041/authservice`	Yes
BATCH_SERVICE_URL	Indicates the batch service URL which gets activated after the `fcc-studio.sh` file runs. The format for the batch service URL is as follows: `http://<HostName>:7043/batchservice`	Yes
META_SERVICE_URL	Indicates the metaservice URL which gets activated after the `fcc-studio.sh` file runs. The format for the metaservice URL is as follows: `http://<HostName>:7045/metaservice`	Yes
SESSION_SERVICE_ URL	Indicates the session service URL which gets activated after the `fcc-studio.sh` file runs. The format for the session service URL is as follows: `http://<HostName>:7047/sessionservice`	Yes
ETL_SERVICE_URL	Indicates the ETL Service URL.	No
FCC_STUDIO_URL	Indicates the Studio URL. The format for the FCC Studio URL is as follows: `http://<HostName>:7008`	Yes
ORE Interpreter Settings		

Table 6–1 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Mandatory
RSERVE_USERNAME	Indicates the RServe username.	No
RSERVE_PASSWORD	Indicates the RServe password.	No
HTTP_PROXY	Indicates the HTTP proxy.	No
HTTPS_PROXY	Indicates the HTTPS proxy.	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
USERS_LIB_PATH	Indicates the path where the R packages are installed.	No
RSERVE_CONF_PATH	Indicates the path where the `Rserve.conf` file is present.	No
DB Details for Studio Schema		
STUDIO_DB_ HOSTNAME	Indicates the hostname of the database where Studio schema is created.	Yes
STUDIO_DB_PORT	Indicates the port number where Studio schema is created.	Yes
STUDIO_DB_SERVICE_ NAME	Indicates the service name of the database where Studio schema is created.	Yes
STUDIO_DB_SID	Indicates the SID of the database where Studio schema is created.	Yes
STUDIO_DB_ USERNAME	Indicates the username of the Studio Schema (newly created Oracle Schema).	Yes
STUDIO_DB_ PASSWORD	Indicates the password for the newly created schema.	Yes
Studio DB Wallet Details		
STUDIO_ALIAS_NAME	Indicates the Studio alias name. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes
STUDIO_WALLET_ LOCATION	Indicates the Studio wallet location. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes
STUDIO_TNS_ADMIN_ PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `STUDIO_ALIAS_NAME` is present. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	Yes
DB Details for BD Config Schema		
BD_CONFIG_ HOSTNAME	Indicates the hostname of the database where BD config schema is installed.	No
BD_CONFIG_PORT	Indicates the port of the database where BD config schema is installed.	No
BD_CONFIG_ SERVICE_NAME	Indicates the service name of the database where BD config schema is installed.	No
BD_CONFIG_SID	Indicates the SID of the database where BD config schema is installed.	No
BD_CONFIG_ USERNAME	Indicates the username for the BD config schema. **Note:** Ensure to enter the value as NA	Yes
BD_CONFIG_ PASSWORD	Indicates the password for the BD config schema.	No
BD Config Wallet Details		
BD_CONFIG_ALIAS_ NAME	Indicates the BD config alias name. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	No

Table 6–1 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Mandatory
BD_CONFIG_WALLET_LOCATION	Indicates the BD config wallet location. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	No
BD_CONFIG_TNS_ADMIN_PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `BD_CONFIG_ALIAS_NAME` is present. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	No
DB Details for BD Atomic Schema		
BD_ATOMIC_HOSTNAME	Indicates the BD atomic schema hostname.	No
BD_ATOMIC_PORT	Indicates the BD atomic schema port number.	No
BD_ATOMIC_SERVICE_NAME	Indicates the BD atomic schema service name.	No
BD_ATOMIC_SID	Indicates the BD atomic schema SID.	No
BD_ATOMIC_USERNAME	Indicates the username of the BD atomic schema.	No
BD_ATOMIC_PASSWORD	Indicates the password of the BD atomic schema.	No
BD Atomic Wallet Details		
BD_ATOMIC_ALIAS_NAME	Indicates the BD atomic alias name. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	No
BD_ATOMIC_WALLET_LOCATION	Indicates the BD atomic wallet location. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	No
BD_ATOMIC_TNS_ADMIN_PATH	Indicates the path of the `tnsnames.ora` file where an entry for the `BD_ATOMIC_ALIAS_NAME` is present. For more information, see Appendix C, "Setting Up Password Stores with Oracle Wallet".	No
SQL Scripts		
FSINFODOM	Indicates the name of the BD Infodom.	No
FSSEGMENT	Indicates the name of the BD segment.	No
DATAMOVEMENT_LINK_NAME	• If the newly created 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 <SCHEMA_NAME> in this parameter. • If the newly created schema is in the same database host, the value for this parameter is the user name of the BD atomic schema. **Note:** Ensure to enter the value as NA	Yes
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. **Note:** Ensure to enter the value as NA	Yes
PGX Setup Details		
PGX_INSTALATION_PATH	Indicates the installation path of the PGX server. Example: `<STUDIO_INSTALLATION_PATH>/pgx/server/`	Yes

Table 6–1 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Mandatory

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

Cloudera Setup Details		
HADOOP_ CREDENTIAL_ PROVIDER_PATH	Indicates the path where Hadoop credential is stored. **Note**: Indicates the Hadoop alias given when creating the Hadoop credentials. For information on how to create a credential keystore, see Creating Credential Keystore.	No
HADOOP_PASSWORD_ ALIAS	Indicates the Hadoop alias given when creating the Hadoop credentials.	No
Hive_Host_Name	Indicates the Hive hostname.	No
Hive_Port_number	Indicates the Hive port number. Contact System Administrator to obtain the port number.	No
HIVE_PRINCIPAL	Indicates the Hive Principal. Contact System Administrator to obtain HIVE_PRINCIPAL.	No
HIVE_SCHEMA	Indicates to create a schema in HIVE.	No
JAAS_CONF_FILE_ PATH	Created for future use.	No
Krb_Host_FQDN_Name	Indicates the Krb host FQDN name.	No
Krb_Realm_Name	Indicates the Krb realm name.	No
Krb_Service_Name	Indicates the Krb service name. Example: Hive	No
KRB5_CONF_FILE_ PATH	Created for future use.	No
security_krb5_kdc_server	Created for future use.	No
security_krb5_realm	Created for future use.	No
server_kerberos_keytab_ file	Created for future use.	No
server_kerberos_principal	Created for future use.	No
SQOOP_ HOSTMACHINE_ USER_NAME	Indicates the user name of the Host machine where sqoop will run.	No
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 in the `SQOOP_ PARAMFILE_PATH` parameter. Example: `/scratch/ofsaa/` **Note**: Ensure that the location name ends with a '/'.	No

Table 6–1 (Cont.) config.sh Parameters

InteractionVariable Name	Significance	Mandatory
SQOOP_PARTITION_COL	Indicates the column in which the HIVE table is partitioned. The value must be `SNAPSHOT_DT`.	No
SQOOP_TRG_HOSTNAME	Indicates the hostname of the SQOOP server. Example: `<HostName>`	No
SQOOP_WORKDIR_HDFS	Indicates the SQOOP working directory in HDFS. Example: `/user/ofsaa`	No

Running the Installer

To run the 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 `install.sh` command does not generate any log file.

3. Execute the following command in the console:

   ```
   ./fcc_studio.sh
   ```

   The OFS Crime and Compliance Studio application is installed without BD. The Data Studio and all the interpreters are started.

   After the successful completion of Studio, the script displays a URL. You can use this URL to launch the FCC Studio Application. For more information, see Launching the FCC Studio Application.

4. Verify the Installation. See Verifying the Installation

Verifying the Installation

To verify the installation, follow these steps:

- Check the log files located in the `<STUDIO_INSTALLATION_PATH>/logs` directory.
  - Ensure that the metaservice is up and running.
  - The authservice and batchservice will not be up.
- In the 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 the further installation and contact Oracle Support with relevant log files.

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

Launching the FCC Studio Application

1. Enter the URL obtained after successful installation of the Studio application in the following format into the browser:

   `https://<Master_Node>:7008`

   The OFS Crime and Compliance Studio application is launched.

7

Post-installation Configuration for Studio Installed Without BD

On successful installation of the Studio application without BD, perform the following post-installation configuration:

- Configuring SSH Connection
- Configuring Interpreters

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

Configuring SSH Connection

To configure SSH connection, follow these steps:

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

       `Generating public/private rsa key pair`
    2. `Enter file in which to save the key (<Linux_Home>/.ssh/id_rsa):` [Press Enter]
    3. `Enter passphrase (empty for no passphrase):` [Press Enter]
    4. `Enter same passphrase again:` [Press Enter]
    5. `ssh-copy-id -i ~/.ssh/id_rsa.pub <BigData_Server>`
    6. `ssh <BigData_Server>`

Configuring Interpreters

After starting the Studio application, the configuration for the interpreters can be performed from the user interface (UI). For information on configuring interpreters, see the *Configuring Interpreters* chapter in the *Oracle Financial Services Crime and Compliance Studio Administration Guide*.

8

Reinstalling Studio Application

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

To reinstall the Studio application, 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>` folder.
4. Unzip the installer archive file, `OFS_FCCM_STUDIO_.0.7.2.0_Linux.zip` and extract the software. For more information, see Extracting the Software.
5. Perform database cleanup by performing the following:

Table 8–1 Database Cleanup

Schema	Applicable for BD	Applicable for Non-BD
Clean up for Studio Schema	Yes	Yes
Cleanup for BD Atomic Schema	Yes	No
Cleanup for BD Config Schema	Yes	No

6. Reinstall the Studio application. For more information, see Installing Studio With BD.

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 Studio Schema Tables section.

Cleanup for BD Atomic Schema

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

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

```
TRUNCATE TABLE DATABASECHANGELOGLOCK;

TRUNCATE TABLE DATABASECHANGELOG;
```

Cleanup for BD Config Schema

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

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

```
TRUNCATE TABLE DATABASECHANGELOGLOCK;

TRUNCATE TABLE DATABASECHANGELOG;
```

A

Starting/Stopping Studio Services

This section describes how to start and stop the services needed for the Studio application. It includes the following sections:

Table 8–2

Start/Stop Studio Services	Applicable for BD	Applicable for Non-BD
Starting/Stopping PGX Service	Yes	No
Starting/Stopping ETL Service	Yes	No
Starting/Stopping Studio	Yes	Yes

Starting/Stopping PGX Service

- To start the PGX service:
  1. Navigate to the path where 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:

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

- To stop the PGX service, kill the process.

Starting/Stopping ETL Service

- To start the ETL service:
  1. Navigate to the path where ETL service is installed.
  2. Run the following command:

     ```
     ./starter.sh
     ```

- To stop the ETL service:
  1. Navigate to the path where ETL service is installed.
  2. Run the following command:

     ```
     ./stopper.sh
     ```

Starting/Stopping Studio

- To start Studio:
  1. Navigate to the `<Studio_Installation_Path>/bin/` directory.
  2. Run the following command:

     ```
     ./fcc-studio.sh
     ```
- To stop Studio, kill all the Studio processes.

Once all the services are up and running, you can launch the Studio application. For more information, see Launching the FCC Studio Application.

B

Tables and Sequences

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

- Studio Schema Tables
- Studio Schema Sequences

Studio Schema Tables

The following table includes the details of the Studio Schema tables that must be dropped during the reinstallation of the Studio application.

Table 8–3 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	DATABASECHANGELOGLOC K
DS_USER_GROUPS	DS_INTERPRETER_VARIANT	DS_COMMENT
DS_PARAGRAPH_ RELATIONS		

Studio Schema Sequences

The following table includes the details of the Studio Schema sequences that must be dropped during the reinstallation of the Studio application:

Table 8–4 Studio Schema Sequences

SEQ_COMMENT	SEQ_ENTITY_PERMISSIONS	SEQ_GRAPH
SEQ_GROUP	SEQ_INTERPRETER_RESULT	SEQ_INTERPRETER_ VARIANT
SEQ_JOB	SEQ_NOTEBOOK	SEQ_PARAGRAPH
SEQ_PERMISSION	SEQ_PERMISSION_MAPPING	SEQ_RESULT_MESSAGE
SEQ_ROLE	SEQ_TASK	SEQ_USER
SEQ_VISUALIZATION_ TEMPLATE		

C

Setting Up Password Stores with Oracle Wallet

This section includes the following topics:

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.

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>` folder 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>:

Table 8–5

Schema	Applicable for BD	Applicable for Non-BD
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.

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>"
   ```

D

Installing RServe Manually

Overview

You must install Rserve manually on a host to expose the local R installation on that host to the network, so that remote Rserve clients such as the R interpreter can use the local R installation.

The R interpreter always connects to an Rserve instance and runs the R code remotely. The interpreter needs to be configured with the hostname or IP and the port of the remote instance (where Rserve is running). When the interpreter is initialized, it connects to the remote instance.

This section includes the following topics:

Prerequisites

The following is a list of prerequisites required before beginning the installation of RServe:

- Ensure that Oracle Linux 7.x and Oracle JDK 8 are validated against Oracle Linux 7.4 and Oracle JDK 8u161.
- The user must be a root user.
- 800 MB disk space is required for package installation.

The following subsections provide more details for prerequisites:

Installing Oracle R Distribution

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

```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.

Installing Dependencies

Rserve has certain dependencies to run correctly. The **openssl-devel** is required for SSL support. The dependencies change based on the libraries you have installed. For example, to let **knitr** send plots as **base64 encoded** strings, you require **pango-devel**.

The following dependent packages must be installed for Rserve to support SSL:

```bash
(root)# yum install openssl openssl-devel pango-devel
```

Installing ORE Client

To connect to ORE through Rserve, follow these steps:

- Install the corresponding client libraries. For more information, see https://docs.oracle.com/cd/E83411_01/OREAD/installing-ORE-client.htm#OREAD167 from the ORE project.
- Install the knitR and PrintR packages.

Installing Rserve

To install Rserve, call the following code in your R shell:

```R
> install.packages('Rserve', repos='https://www.rforge.net/')
```

If you are behind a proxy, ensure that R is communicated about it when you start the R shell.

For example, you can start R shell as shown in the following before installing any package:

```bash
$ http_proxy=http://your-proxy:80 R
```

Configuring Rserve

You can base your config on the following example configuration, which you should store in an `Rserve.conf` file. You will require `Rserve.conf` as a reference when you start Rserve.

```
auth required

plaintext disabled

pwdfile /path/to/Rserve.pwd

remote enable

switch.qap.tls enable

tls.port 6311

qap disable

interactive no

rsa.key /path/to/server.key

tls.key /path/to/server.key

tls.cert /path/to/server.crt
```

This configuration tells Rserve to encrypt the communication with TLS and listen for incoming connections on port **6311**. The `Rserve.pwd` file appears as shown in the following example:

```
user $5baa61e4c9b93f3f0682250b6cf8331b7ee68fd8
```

The file contains one line per user, where the first part is the username and the second part is the password. The password can either be plain text or a **MD5/SHA1** hash. In this example, the password `password` is hashed with SHA1. If you use hashed passwords, the password must start with a `$` sign.

The `rsa.key, tls.key`, and `tls.cert` settings point to the private key files you require for TLS. These keys can be generated using the **openssl** command-line tool as shown in the following example:

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

$ openssl req -new -key server.key -out server.csr

$ openssl x509 -req -days 265 -in server.csr -signkey server.key -out
server.crt
```

The preceding sample is an example and for a production deployment, you should use relevant certificates. You can find more information about configuration options on the Rserve homepage - https://www.rforge.net/Rserve/doc.html.

Starting Rserve

After installing Rserve and placing configuration files in the correct location, start the Rserve as given here:

```bash
```

```
$ R CMD Rserve --no-save --RS-conf /path/to/Rserve.conf
```

Rserve starts in the background. After it starts, the R Interpreter is able to connect to it. The Rserve process is running before you start the R Interpreter.

Adding the Certificate to the Keystore

The certificates that were generated in the previous step to configure Rserve to encrypt the communication must be added to the Java keystore in order to be used by the R interpreter. The add procedure depends on your setup.

Add the certificate to a keystore as given here:

```bash
$ $JAVA_HOME/bin/keytool -import -alias rserve -file /path/to/server.crt -keystore /path/to/keystore-storepass storepassword -noprompt
```

The certificate must be imported correctly and the correct keystore is used by the Java process you use to start the R interpreter. Else you will get SSL related exceptions when the interpreter attempts to connect to Rserve.

You can specify the keystore when starting the R interpreter as shown in the following example:

```bash
$ $JAVA_HOME/bin/java -Djavax.net.ssl.trustStore=/path/to/keystore -
Djavax.net.ssl.trustStorePassword=storepassword <additional paramters>
```

Installing Additional Libraries

Depending on your use case, you must install other R libraries. For example, you can install **knitr** or **ggplot2**, in the same manner, that you installed Rserve previously. You can use the *package.install* within your R shell to perform the installation.

For example:

```R
> install.packages('knitr')
```

E

Creating Required Index Files

Creating Required Index Files

To create the required index files, follow these steps:

1. Create the index files, `name_index.csv` and `address_index.csv` with the column names are per the configuration mentioned in the `<ETL_Installation_Path>/crosslangnamematch/conf/NameMatchingLocations.properties` file with dummy values.

   For more information on `NameMatchingLocations.properties` file, see the *Cross Language Name Matching Service* section in the *Studio Services* chapter in the *OFS Crime and Compliance Studio Administration Guide*.

   Note: The entries in the CSV file must be tab-separated.

   For example:

Table E–1 name_index.csv

node_id	name	source
1	dummy	dummy

Table E–2 address_index.csv

node_id	address	source
1	dummy	dummy

F

Providing Spark Libraries

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

1. Place your unarchived Spark libraries in a directory accessible by FCC Studio.

   You can download the Spark libraries from the Spark's Official Release Archive.

2. Set the environment variable `SPARK_HOME` to the absolute path of your Spark libraries.

3. Place your Hadoop client-configuration files to a directory accessible by FCC Studio.

   You can 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 configuration files directly from the cluster manager's UI if you are using a Cloudera cluster.

4. Set the system environment variable `HADOOP_CONF_DIR` to the absolute path of the directory that contains the configuration files.

G

Using Python Virtual Environments with PySpark

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

1. Creating a Virtual Environment with Conda
2. Updating Interpreter Properties

Creating a Virtual Environment with Conda

Note: You can also use virtualenv to create your virtual environment instead of Conda.

1. Ensure that you have Conda and Conda-Pack installed.
2. Create your virtual environment by executing 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 by executing 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 by executing the following command:

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

Updating Interpreter Properties

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

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

H

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 require its own start-script.

   1. Navigate to the `<Studio_Installation_Path>/interpreters/bin` directory and execute the following command to create a new start-script called `start-spark2-interpreter.sh`:

      ```
      cp start-spark-interpreter.sh start-spark2-interpreter.sh
      ```

   2. 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 which 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.ZeppelinRemoteInterpreterServer
      --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.ZeppelinRemoteInterpreterServer
      ${1:-7030} > $DIR/../../logs/spark2.log
      ```

   3. 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.

   1. Navigate to the <Studio_Installation_Path>/interpreters/conf directory and execute the following command to create the new interpreter JSON called spark2.json:

      ```
      cp spark.json spark2.json
      ```

   2. 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",
          "classNameName": "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.

1. Navigate to the `<Studio_Installation_Path>/interpreters/conf` directory and execute the following command to create the new interpreter JSON called `pyspark2.json`:

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

2. 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 the two interpreters to different external clusters when setting the environment variables, `SPARK_HOME` and `HADOOP_CONF_DIR` as part of

Providing Spark Libraries

,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 the Studio application. For more information, see Starting/Stopping Studio.