Installation Guide
Oracle Banking Corporate Lending Process Management

Release 14.3.0.0.0

Part No. F18230-01

May 2019

ORACLE
FINANCIAL SERVICES

ORACLE

Installation Guide
May 2019
Version 14.3.0.0.0

Oracle Banking Installer
Oracle Park
Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India

Worldwide Inquiries:
Phone: +91 22 6718 3000
Fax:+91 22 6718 3001
www.oracle.com/financialservices/

Table of Contents

1. About this Manual

1.1 Introduction

Purpose:

This document provides the procedure to use the installer product.

Audience:

This guide is primarily intended for Developers for Oracle Banking products. Some information may be relevant to IT decision makers and users of the application are also included. Readers are assumed to possess basic operating system, network, and system administration skills with awareness of vendor/third-party software's and knowledge of Oracle Banking products.

1.2 Scope

1.2.1 Read Sections Completely

Each section should be read and understood completely. Instructions should never be blindly applied. Relevant discussion may occur immediately after instructions for an action, so be sure to read whole sections before beginning implementation.

1.2.2 Understand the Purpose of this Guidance

The purpose of this guidance is to provide procedure to use Oracle Banking Installer Product.

1.2.3 Limitations

This guide is limited in its scope to provide procedure to use Oracle Banking Installer Product.

2. Installer Product

The following sections require to be completed in a sequence to finish the installation.

2.1 Prerequisites

Below listed are the prerequisites which are required to use the Oracle Banking Installer:

1. Node machines must have Oracle Linux 7 installed with Oracle JDK 1.8 Update 202.
2. Install the Chef Client on the node machines to use the Oracle Banking Installer. Refer Installation of Chef Client section for detailed instructions to install chef client.
3. Make sure that the PDB with applicable DB schemas for the product are available.
4. Install the Oracle SQL Client must be installed on the node machine to use the Oracle Banking Installer for the DB SQL execution.
5. Make sure that the required number of WebLogic (12.2.1.3) managed servers and clusters for the product are up & running on SSL with Custom Trust Key Store (jks).
6. Create the data bags required for the product installation. Refer Data bag Creation section for detailed instructions to create the data bags.
7. Set bash shell on node machines to run the Oracle Banking Installer.
8. Configure the proxy on each node machine. Set up the proxy with the following commands. (*Proxy setup is required only to update the yum package*)

```
$ export http_proxy=http://USERNAME:PASSWORD@proxy-server.mycorp.com:port/

$ export https_proxy=https://USERNAME:PASSWORD@proxyserver.mycorp.com:port/
```

9. Update yum using the following command.

```
$ yum update yum
```

10. Make sure that the system date & time is valid and latest.
11. Rabbit MQ server is configured and properties related are compiled into PLATO schema.
12. FQDN should be set properly. You can check the current FQDN using this command.

```
$ hostname -f
```

2.1.1 Installation of Chef Client

Do the following steps to install the chef client:

1. Login to node machine as root user.
2. Copy the product OSDC bundle to the /scratch folder of the Oracle Linux Server i.e. node machine.
3. Extract the product OSDC bundle and navigate to */scratch/<extracted OSDC bundle>/ oracle_banking_installer_pack/softwares/chef-client/14.8.12.*

```
$ cd /scratch/<<extracted OSDC bundle>>/oracle_banking_installer_pack/softwares/chef-client/14.8.12
```

4. Execute the below command to install the chef-client

```
$rpm -ivh chef-14.8.12+20190122094824-1.el7.x86_64.rpm
```

5. Verify chef 14.8.12 is installed in the machine using this command.

```
$ chef-client --version
```

Chef: 14.8.12

2.1.2 Data bag Creation

The data bag contains the encrypted passwords for the WebLogic, DB and SSL. These passwords will be used by the Oracle Banking Installer during the product installation.

Chef requires a secret key to encrypt and decrypt the data bag items. Install the openssl in the node machine and use the following command to generate a secret key:

```
$ openssl rand -base64 512 > /scratch/<extracted OSDC
bundle>/oracle_banking_installer_pack/chef-repo/secrets/secret_key
```

Note: Open ssl is one of the option to generate the secret key.

2.1.2.1 WebLogic

1. Navigate to */scratch/<extracted OSDC bundle>/oracle_banking_installer_pack/chef-repo/data_bags/fsgbu_weblogic_deploy/* folder.

2. Edit the json file “datasource_credentials.json”. This json file will contain the data source passwords.

```
$ cd fsgbu_weblogic_deploy
$ vi datasource_credentials.json
```

3. Update plain data source passwords against the corresponding data source name in the json file.

```
{
 "id": "datasource_credentials",
  "<datasource name>": "<password>",
  "<datasource name>": "<password>",
  "<datasource name>": "<password>",
  "<datasource name>": "<password>"
}
```

4. For reference, see the below example datasource_credentials.json with sample content.

```
{
 "Id": "datasource_credentials",
 "PLATO": "welcome1",
 "PLATO_SECURITY": "welcome1",
```

```
 “PLATO_UI”:  "welcome1",
 "SMS": "welcome1",
 "COMMONCORE": "welcome1",
 "OBCLPMBACKOFF": "welcome1",
 "OBCLPMBUSSOVR": "welcome1",
 "OBCLPMBUSSPRC": "welcome1",
 "OBCLPMBUSSPRD": "welcome1",
 "OBCLPMHANDOFF": "password1",
 "OBCLPMLOANAPP": "password1",
 "OBCLPMLOANCUS": "password1",
 "OBCLPMMAINTCE": "password1",
 "OBCLPMORCHESN": "password1",
 "OBCLPMSYNDAPP": "password1",
 "OBCLPMTXNORCH": "password1"
}
```

5. Execute the below knife command from chef-repo directory to encrypt the passwords.

```
knife data bag from file fsgbu_weblogic_deploy datasource_credentials.json --
secret-file /scratch/<extracted OSDC
bundle>/oracle_banking_installer_pack/chef-repo/secrets/secret_key --local-
```

6. Encrypted data bag file look like below (for reference).


```
 {

   "id": "datasource_credentials",

   "PLATO": {

      "encrypted_data": "yIaX16MzESqLycvLbtKqs59wpnlf56I=\n",

      "iv": "GfYNB5gfB9D2TzNE\n",

      "auth_tag":

      "6mSbTWq8NSEuXLrG0K4MEA==\n",

      "version": 3,
```

},

"PLATO_SECURITY":

{

"encrypted_data":

"igH5nzbCW074e2ML6Ma84Akv2nzb+HE=\n", "iv":

"WQb0M53oYkUcYX24\n",

"auth_tag": "D4r1JeyO8Gp9yIUEFTS/1Q==\n",

"version": 3,

"cipher": "aes-256-gcm"

}, "PLATO_UI":

{

"encrypted_data":

"g1sNKfkOKskKyMdKqO/VUhcgoTsfq1g=\n", "iv":

"BywFi5W16xC+cWsW\n",

"auth_tag": "+oazn8cKEz0g7hUeXFte+w==\n",

"version": 3,

"cipher": "aes-256-gcm"

}, "SMS":

{

"encrypted_data": "aJHzpnrPYbAI9EJvhv+tKTcLrEUFK0o=\n",

"iv": "iDccRYy3xF/+JuRN\n",

"auth_tag": "v/0DeRg0Soyqg3FjdpusUw==\n",

"version": 3,

"cipher": "aes-256-gcm"

},

"COMMONCORE

": {

"encrypted_data":

"F1N/JQ8fwWuCB1ZgkMMpT1vZM5gDRtU=\n", "iv":

"5eDpiTklyacMdqV8\n",

"auth_tag": "ZDM58Q2rj5FzTMZkIa4ESQ==\n",

"version": 3,

"cipher": "aes-256-gcm"

},

"OBCLPMBACKOFF

": {

"encrypted_data": "ED4rzRM/nez2Rihpjluq+AGg0+bNhkY=\n",

"iv": "BBwF+aRduA40eF8K\n",

"auth_tag": "FNDDrnMNCCml6jB0jrqi4Q==\n",

"version": 3,

"cipher": "aes-256-gcm"

},

"OBCLPMBUSSOVR

": {

"encrypted_data":

"Mm5PFgbwOePbI6cYks/L6gG/Pt5dMm0=\n", "iv":

"3NNlwA3k6NrSD6af\n",

"auth_tag": "YKRWYQuqVYNljAEZcIty6A==\n",

"version": 3,

"cipher": "aes-256-gcm"

},

"OBCLPMBUSSPRC

": {

"encrypted_data":

"z8A1PRubiB0zlE7CH+YOwK+BwzKLcwY=\n", "iv":

"fhjRx/YkQN6ztFqF\n",

"auth_tag": "ycSI+7lPAjDbwOgf0olvOA==\n",

"version": 3,

"cipher": "aes-256-gcm"

},

"OBCLPMBUSSPRD

": {

"encrypted_data":

"nnY6O2F1GA63im+QSs7K3kh2ZCkwSlU=\n", "iv":

"zBqWrenS1dghpIyI\n",

"auth_tag": "VDHJJ+EdsjVIKiTtuuxABQ==\n",

"version": 3,

"cipher": "aes-256-gcm"
},

"OBCLPMHANDOFF

": {

"encrypted_data": "DT5g8JEaDtqpDoLqjIoBm2MJoyvB138=\n",

"iv": "g+kxIPzJlDYwck3q\n",

"auth_tag":

"g+tQW2ZLxQEdSOWtLmurtw==\n", "version":

3,

"cipher": "aes-256-gcm"

},

"OBCLPMLOANAPP

": {

"encrypted_data":

"XmgPdnwscQ+CXc3TdQ+zt/FesaxAh3Y=\n", "iv":

"4XyZJ7VIIdwNCfD2\n",

"auth_tag": "7HM9ZVWiJFHaneyj0LGz6g==\n",

"version": 3,

"cipher": "aes-256-gcm"

},

"OBCLPMLOANCUS

": {

"encrypted_data":

```
                "JCB0DTQBMIMzWAsL0fU+mVGz4FV6Igg=\n",
                "iv": "dNPO5L4t+axCe9e7\n",
                "auth_tag":
                "qoYwJdO91ithgkEUTfYxSQ==\n"
                , "version": 3,
                "cipher": "aes-256-gcm"
},
"OBCLPMMAINTC
E": {
                "encrypted_data":
                "zYcLpYUT/EmhXURwZEIZDutwJS3+9+0=\n",
                "iv": "I+b6DkWFNSitGZKK\n",
                "auth_tag":
                "59xJqTtYTDp6sQZkT/wI7g==\n",
                "version": 3,
                "cipher": "aes-256-gcm"

  },
  "OBCLPMORCHESN
  ": {
    "encrypted_data": "Mi38tbFr6ctQO3OEkcc3MzUL229m394=\n",
    "iv": "shAzq/F9O3nCSGBm\n",
    "auth_tag":
```


7. Edit the json file “ssl_credentials.json”. This json file will contain the Custom TrustKeyStore Passphrase.

```
$ cd fsgbu_weblogic_deploy
$ vi ssl_credentials.json
```

8. Update plain Custom TrustKeyStore Passphrase against “ssl_password” key in the json file.

```
{
```

```
  "id": "ssl_credentials_",

  "ssl_password": "<Custom TrustKeyStore Passphrase>"

}
```

9. For reference, see the below example ssl_credentials.json with sample content.

```
{

  "id": "ssl_credentials",

  "ssl_password": "welcome1"

}
```

10. Execute the below knife command from chef-repo directory.

```
knife data bag from file fsgbu_weblogic_deploy ssl_credentials.json--secret-
file /scratch/<extracted OSDCbundle>/oracle_banking_installer_pack/chef-
repo/secrets/secret_key --local-mode
```

11. Encrypted data bag file look like below (for reference).

```
{

  "id": "ssl_credentials",

  "ssl_password": {

    "encrypted_data":

    "gucJa+S8uYf9oFNTFRej9gXB6Mr/z8hOH30FdKdO\n", "iv":

    "kggQfDkHJNKx/ddP\n",

    "auth_tag": "vZZx+dQQrop96X5Znwyoxg==\n",

    "version": 3,

    "cipher": "aes-256-gcm"

  }
```

12. Edit the json file “weblogic_credentials.json”.This json file will contain the WebLogic admin username and password.

```
$ cd fsgbu_weblogic_deploy
$  vi weblogic_credentials.json
```

13. Update plain text WebLogic admin username and password against “wl_admin_username” and “wl_admin_password” keys respectively in the json file.

```
{

  "id": "weblogic_credentials",
```

```
  "wl_admin_password": "<weblogic admin user password>",

  "wl_admin_username": "<weblogic admin user name>"

}
```

14. For reference, see the below example weblogic_credentials.json with sample content.

```
{

  "id": " weblogic_credentials",

  "wl_admin_password": "welcome1",

  "wl_admin_username": "weblogic"

}
```

15. Execute the below knife command from chef-repo directory.

```
knife data bag from file fsgbu_weblogic_deploy weblogic_credentials.json--
secret-file /scratch/<extracted OSDC
bundle>/oracle_banking_installer_pack/chef-repo/secrets/secret_key--local-
mod
e
```

16. Encrypted data bag file look like below (for reference)

```
{

  "id": "weblogic_credentials",

  "wl_admin_password": {

    "encrypted_data":

    "vauqQm/Imoig1u7XW8ciWnRDkVk7LM+p8Hs98g==\n", "iv":

    "BoN0RFK7KBnVqcSM\n",

    "auth_tag": "ozvF+Weo7sfVIEOa/tgfMA==\n",

    "version": 3,

    "cipher": "aes-256-gcm"

  },

  "wl_admin_username": {

    "encrypted_data":

    "nlVnxPa1GcUp9jEheZKX4CgkDw6hP7hXwghk\n", "iv":

    "ibZRT+G1JTqKsTS4\n",

    "auth_tag": "FXurd63q3yfFwU1xUOJSXA==\n",
```

```
  }

}
```

17. Make sure all json files are on 755 permission.

2.1.2.2 Database

1. Navigate to /scratch/*<extracted OSDC bundle>/* oracle_banking_installer_pack/chef-repo/data_bags/fsgbu_db_deploy folder.

2. Edit the json file “schema_credentials.json”. This json file contains the db login passwords.

```
$ cd fsgbu_db_deploy

$ vi schema_credentials.json
```

3. Update the database passwords against their respective place holders in the json file. Schema Passwords should be updated against their respective schema names.

```
{

  "id": "schema_credentials",

  "<schema name>_schema_password": "<schema

  password>", "<schema name>_schema_password":

  "<schema password>"
```

4. For reference, see the below example schema_credentials.json with sample content.

```
{

  "id": " schema_credentials",

  "passwd_blade": "fsgbu"

}
```

5. Execute the below knife command from chef-repo directory.

knife data bag from file fsgbu_db_deploy schema_credentials.json --secret-file /scratch/*<extracted OSDC bundle>*/oracle_banking_installer_pack/chef-repo/secrets/secret_key --local-mode

6. Encrypted data bag file look like below (for reference).

```
{

  "id": "schema_credentials",

  "passwd_blade": {

    "encrypted_data": "ijshvhrLsWQp70Mg9U+GCw9gQKv5AVc=\n",

    "iv": "0VxnQpRSLzxMWOaE\n",
```

```
    "auth_tag":

    "CRHfn70y7KT5w6nCMVo+hg==\n", "version":

    3,

    "cipher": "aes-256-gcm"

  }
```


7. Make sure all json files are on 755 permission.

2.2 Database Installation

Note: Make sure that all the DB schemas which are required for the product are available before installation.

1. Login to Linux server as root user

2. Go to /scratch/<*extracted OSDC bundle*>/oracle_banking_installer_pack/chef-repo folder.

3. Update Parameter file "***db_deploy_properties.rb***" with relevant values as described in below table.

Attribute	**Description**
ORACLE_SERVER_HOSTNAME	Oracle Database Server Hostname
ORACLE_CLIENT_INSTALL_USER	User that owns the Oracle home directory.
ORACLE_CLIENT_HOME	Path to Oracle database Client home
PDB_SID	PDB SID
PDB_PORT	PDB Port
PRODUCT_CUSTOM_SCRIPTS_HOME	Path to the product directory inside the OSDC folder. Eg:- /scratch/OSDC/<product directory>
PLATO_SCHEMA_USERNAME	Plato schema username
PLATO_SECURITY_SCHEMA_USERNAME	Plato security schema username
ORACLE_SERVER_VERSION	Oracle server version
PLATO_UI_SCHEMA_USERNAME	Plato UI schema username
SMS_SCHEMA_USERNAME	SMS schema username
COMMONCORE_SCHEMA_USERNAME	Commoncore schema username
MOCORE_SCHEMA_USERNAME	Mid office commoncore schema username
OBCLPM_SCHEMA_USERNAME	OBCLPM schema username

4. Execute the shell script ***db_deploy_installer.sh*** under /oracle_banking_installer_pack/chef-repo

5. Tail the log file to see the progress

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

2.3 WebLogic Installation

Note: Make sure that all the DB schemas which are required for the product are available and make sure that the required number of WebLogic managed servers and clusters for the product are up & running on configured SSL before installation.

1. Login to Linux server as root user

2. Go to /scratch/*<extracted OSDC bundle>*/oracle_banking_installer_pack/chef-repo folder.

3. Make sure that the Weblogic install user (user that owns weblogic home directory) has write permission on the extracted OSDC bundle.

4. Update Parameter file "*pre_installation_properties.rb*" with relevant values as described in below table

Attribute	**Description**
PRODUCT_BUNDLE_HOME	Path to the product directory inside the OSDC folder. E.g.:- /scratch/OSDC/<product directory>.
APP_STARTERS_PATH_URL	Path to the destination folder inside which App starters files should be copied.
TASKS_PATH_URL	Path to the destination folder inside which Task files should be copied.

5. Execute the shell script ***pre_installation_package_mw_deploy.sh*** under /oracle_banking_installer_pack/chef-repo

6. Tail the log file to see the progress

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

7. Update Parameter file "*mw_deploy_properties.rb*" with relevant values as described in below table.

Attribute	**Description**

JAVA_HOME	Path to the directory where java is installed.
WLS_HOME	Path to the WebLogic home directory.
WLS_INSTALL_USER	User who owns the WebLogic home directory
WLS_INSTALL_USER_GROUP	Group that owns the WebLogic home directory.
WLS_DOMAIN_NAME	Name of the WebLogic domain
IS_TARGET_CLUSTER	Whether the targets to which the deployment is to be done are clusters (true/false)
WLS_SSL_CUSTOM_TRUST_FILE	Location of the SSL custom trust file
WLS_SSL_CUSTOM_TRUST_KEYSTORE_TYPE	SSL Trust KeyStore Type(JCEKS or JKS)
WEBLOGIC_ADMIN_HOST	Weblogic Admin Host
WEBLOGIC_ADMIN_LISTEN_PORT	WebLogic Admin listen port
WEBLOGIC_ADMIN_SSL_PORT	WebLogic Admin SSL port
ORACLE_PDB_SID	Oracle PDB SID for data source configuration
ORACLE_PDB_HOSTNAME	Oracle PDB hostname for data source configuration
ORACLE_PDB_PORT	Oracle PDB port for data source configuration
PRODUCT_BUNDLE_HOME	Path to the product directory inside the OSDC folder. E.g.:- /scratch/OSDC/<product directory>
PLATO_DS_USERNAME	Plato datasource username
PLATO_SECURITY_DS_USERNAME	Plato security datasource username
PLATO_UI_CONFIG_DS_USERNAME	Plato UI Config datasource username

PLATO_DS_TARGET	Plato datasource target
PLATO_SECURITY_DS_TARGET	Plato security datasource target
PLATO_UI_CONFIG_DS_TARGET	Plato UI config datasource target
DISCOVERY_SVCS_TARGET	Discovery services target
CONFIG_SVCS_TARGET	Config services target
API_GATEWAY_TARGET	API gateway target
UI_CONFIG_TARGET	UI config target
TASK_SERVER_TARGET	Task server target
SMS_CORE_SVCS_TARGET	Sms core services target
COMMON_CORE_TARGET	Commoncore services target
CMNCORE_DS_USERNAME	Commoncore datasource username
MOCORE_SCHEMA_USERNAME	Mid office commoncore schema username
SMS_DS_USERNAME	Sms datasource username
CMN_CORE_SVCS_TARGET	Commoncore services target
OBCLPM_SVCS_TARGET	OBCLPM services target
UI_APPSHELL_TARGET	UI appshell target
CMN_CORE_SVCS_TARGET	Commoncore services target
CMN_CORE_DS_TARGET	Commoncore datasource target
SMS_CORE_DS_TARGET	Sms core datasource target
OBCLPM_DS_TARGET	OBCLPM datasource target

OBCLPM_DS_USERNAME	OBCLPM datasource username

8. Execute the shell scripts under /oracle_banking_installer_pack/chef-repo in the following order

    1) plato_mw_deploy_installer.sh
    2) sms_mw_deploy_installer.sh
    3) commoncore_mw_deploy_installer.sh
    4) domain_services_mw_deploy_installer.sh
    5) midoffice_commoncore_mw_deploy_installer.sh
    6) ui_mw_deploy_installer.sh

Note: Before running each shell script, update the properties in "*mw_deploy_properties.rb"* as described in the above table.

9. Tail the log file to see the progress

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