Security Guide
Oracle FLEXCUBE Investor Servicing
Release 12.3.0.0.0
[September] [2016]

ORACLE
FINANCIAL SERVICES

ORACLE

Table of Contents

1. About this Manual

1.1 Introduction

Purpose:

This document provides security-related usage and configuration recommendations for Oracle FLEXCUBE Investor Servicing. This guide may outline procedures required to implement or secure certain features, but it is also not a general-purpose configuration manual.

Audience:

This guide is primarily intended for IT department or administrators deploying FLEXCUBE and third party or vendor software's. 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 FLEXCUBE application.

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 the guidance is to provide security-relevant configuration recommendations. It does not imply the suitability or unsuitability of any product for any particular situation, which entails a risk decision.

1.2.3 Limitations

This guide is limited in its scope to security-related issues. This guide does not claim to offer comprehensive configuration guidance. For general configuration and implementation guidance refer to other sources such as Vendor specific sites.

1.2.4 Test in Non-Production Environment

To the extent possible, guidance should be tested in a non-production environment before deployment.

Ensure that any test environment simulates the configuration in which the application will be deployed as closely as possible.

2. General Principles

The following general principles motivate much of the advice in this guide and should also influence any configuration decisions that are not explicitly addressed.

2.1 Encrypt Transmitted Data Whenever Possible

Data transmitted over a network, whether via wire or wirelessly, is susceptible to passive monitoring. Whenever practical mechanisms exist for encrypting this data-in-transit, they should be applied. Even if data is expected to be transmitted only over a local network, it should still be encrypted if possible. Encrypting authentication data, such as passwords, is particularly important.

2.2 Encrypt Stored Data Whenever Possible

Data on mobile devices or system is particularly susceptible to compromise due to loss of physical control. Whenever practical solutions exist, they should be employed to protect this data.

2.3 Minimize Software to Minimize Vulnerability

The easiest and simplest way to avoid the vulnerabilities in a particular piece of software is to avoid installing the unwanted software altogether.

2.4 Leverage Security Features, Never Disable Them

Security features should be effectively used to improve a system's resistance to attacks. These features can improve a system's robustness against attack for only the cost of a little effort spent doing configuration.

2.5 Grant Least Privilege

Grant the least privilege necessary for users to perform tasks. The more privileges (or capabilities) that a user has, the more opportunities he or she will have to enable the compromise of the system (and be a victim of such a compromise). Similarly, it is possible to restrict the installation of third party apps, and this may be the right balance between security and functionality for some environments.

About *Oracle Software Security assurance* refer below link:

http://www.oracle.com/us/support/assurance/overview/index.html

2.6 Warnings:

- As with any other information system, do not attempt to implement any of the recommendations in this guide without first testing in a non-production environment.
- This document is only a guide containing recommendations. It is not meant to replace well-structured policy or sound judgment. Furthermore this guide does not address site-specific optimization, configuration concerns.
- Care must be taken when implementing this guide to address local operational and policy concerns.

- The configuration settings described in this document apply only to the limited scope, version etc. The guidance may not translate gracefully to other systems or versions, although applying vendor updates is always recommended.
- For further details on each suggested setting always refer the vendor specific sites

2.7 Organization of the document

The document addresses the areas of installation, configuration, deployment and operation of Oracle FLEXCUBE Investor Servicing, in the below described manner:

Chapter 1	*About this Manual* gives information on the intended audience. It also lists the various chapters covered in this User Manual.
Chapter 2	*General Principles* gives information on general points that needs to be followed in this User Manual
Chapter 3	*Pre-Installation* gives information on guidelines that are recommended to secure the host servers (Application Server, Database Server and others) in an installation of Oracle FLEXCUBE Investor Servicing.
Chapter 4	*Installation* gives information on guidelines served to secure the Oracle FLEXCUBE Investor Servicing application deployed on Oracle WebLogic Server
Chapter 5	*Post-Installation* gives information on the guidelines to be followed post installation
Chapter 6	*General Information* provides information about practices to be employed for client workstations.

3. Pre-Installation

3.1 Data Center Practices

3.1.1 Overview

The following guidelines are recommended to secure the host servers (Application Server, Database Server and others) in an installation of Oracle FLEXCUBE Investor Servicing.

3.1.2 Physical System Security

It is highly recommended to operate servers in a secured data center to prevent unauthorized users or operating personnel from tampering with the machines.

3.1.3 Minimize the Server Footprint

Each logical software component (Application Server, Database Server etc.) in the installation should preferably operate in a dedicated server. It is not recommended to operate multiple services like mail, FTPS, LDAP etc. on the same server, unless absolutely necessary.

It is preferable to customize the operating system installation so that only the minimum set of software components is installed.

Development tools should not be installed on the production servers. In cases where a software package should be compiled and built before installation, it is advisable to perform the build process on a separate machine, following which installation of the binary can be performed on the server.

Samples and demos should not be deployed on a production server, since they are bound to be developed without considering security. Any bugs in such software can be exploited by an attacker resulting in a security incident.

3.1.4 Operating System Users and Groups

It is recommended to minimize the number of user accounts on the host, for easier auditing and management. Besides, it reduces the risk of unauthorized personnel accessing the server.

It is recommended to create user accounts with names that are not easily guessable. There should be at least two system administrator accounts for a server, to ensure backup in the eventuality of one account being locked.

Passwords for all accounts should be strong passwords – this should be enforced by the operating system, for instance, via the ***pam*** configuration in UNIX. Passwords should not be easy to guess, and neither should they be stored in an insecure media, or written down for easy remembrance.

Passwords should be set to expire periodically; 60-90 days is the recommended period. Passwords for privileged accounts may have a shorter lifecycle.

3.1.5 Restrict File System Access

It is recommended to use a file system that allows maintenance of access rights.

In Windows, NTFS allows for ACLs to be maintained at the most granular level; however, due care should be exercised when granting file system privileges to the “Everyone” group. Similarly, in UNIX like operating systems, privileges should not be granted to the “Nobody” user and group, unless absolutely required.

3.1.6 Network Perimeter Protection

Firewall rules should be established to ensure that only a required set of services is accessible to machines outside the data center. Network access can be further restricted to ensure that only certain subnets with trusted machines, and not all machines, can access machines in the data center.

Oracle Financial Services does not recommend exposing the application server hosting Oracle FLEXCUBE Investor Servicing to the Internet.

3.1.7 Network Service Protection

Network services installed on the server should be enabled only to serve the primary business function(s) that the server must provide. Disable all services that are not needed to serve a justified business need.

Review the network services (like mail and directory services) running on the servers to ensure that they are adequately protected from abuse by an attacker.

Also review and limit the network file shares on the servers, to reduce the risk of an attack on the file system. It is recommended to share files and directories on servers only to trusted machines in the network.

3.1.8 Usage of Protected Ports

It is not recommended to execute long processes like application servers and database servers under the root account, since a compromise of such processes will result in an attacker gaining elevated privileges.

Therefore, limit the use of protected ports (port numbers less than 1024 on UNIX like operating systems), since they require the use of a privileged user account (in most cases, this is only the root account). Consider the use of NAT to map protected ports to unprotected ones.

3.1.9 Installation of Software in Production Mode

It is highly recommended to install production builds of any software on production servers. For example, Oracle WebLogic Server should be installed in the production mode, as opposed to the default of development mode. The Oracle Database Server should be installed with options required for production usage (for instance, do not install the sample schemas).

Moreover, it is highly recommended to refer to the manuals and documentation provided by the software supplier, for installing and operating such software securely in a production environment.

3.1.10 Software Updates and Patches

It is recommended to subscribe to security bulletins and advisories published by software vendors to ensure that critical servers are always up to date.

Oracle Financial Services recommends that patches be tested to ensure that they do not conflict with the normal operation of the system.

3.1.11 Usage of Security Appliances and Software

Consider the usage of security appliances and software to monitor and ensure that the production environment continues to be secure after the process of server preparation.

Intrusion Detection Systems can be employed to monitor for security sensitive changes in the system and alert personnel. Antivirus scanners can be used to prevent the server(s) from being compromised. Note that, although UNIX like operating systems may have better defenses against viruses (and other malware), consider running antivirus scanners on servers regardless of the OS.

3.1.12 Configure Security Auditing

Most server operating systems (Linux OS with kernel version 2.6 onwards, Microsoft Windows Server 2003 etc.) allow for auditing file and directory access. Oracle Financial Services recommends enabling this feature in order to track file system access violations. It is not recommended to enable audit for normal file access operations; audits should preferably contain records of violations to reduce the amount of noise in the logs.

Administrators should ensure sufficient disk space for the audit log. Additionally, administrators should factor the increase on server load due to auditing being enabled.

3.1.13 Separation of Concerns

It is not recommended to perform development of any kind on a production machine. The standard practice is to establish a separate development environment for developers, isolated from the testing/staging and production environments. Additional environments can be created for other purposes (for instance, a post-production support environment).

3.1.14 Backup Controls

Back-ups should be taken regularly. This will minimize downtime if there is an emergency. Access to the application areas should not be at the operating system level. On-line archival of redologs should be set up from the date of going live. It is recommended that:

- Backup of all database related files viz., data files, control files, redologs, archived files, init.ora, config.ora etc should be taken at the end of the day.
- The tape can be recycled every week by having day-specific tapes.
- On-line backup of archived redo-log files onto a media to achieve to the point recovery in case of crash, shutdown etc.(recycled every day)
- Complete export of database and softbase should be done atleast once in a week and this can be stored off-site (media can be recycled in odd and even numbers).
- Complete backup of the Oracle directory (excluding the database related files) to be taken once in a month. This media can be recycled bimonthly.

- When the database is huge, incremental exports and on-line tablespace backups are recommended.

The above strategy may be improvised by the Oracle DBA, depending on the local needs. The backup operations are to be logged and tapes to be archived in fireproof storages.

3.2 Oracle Database Security

3.2.1 Overview

This section contains security recommendations for the Database.

3.2.2 Hardening

Review database links in both production and development environments. Unwanted links need to be dropped.

3.2.3 Authentication

Middle-tier applications logon to the database through application schemas rather than end-user accounts. Some individuals (IT Administrators) may require direct access to the application database via their own schema.

This setting prevents the database from using an insecure logon protocol. Make sure init.ora contains:

REMOTE_OS_AUTHENT=FALSE

Following an installation, the application database instance contains default, open schemas with default passwords. These accounts and corresponding passwords are well-known, and they should be changed, especially for a database to be used in a production environment.

Use the SQL*Plus PASSWORD command to change a password:

SQL> PASSWORD <SCHEMA>

Metalink Patch note 4926128 contains a SQL script that will list all open accounts with default password in your database.

In addition, the password to the default accounts like SYS, SYSTEM etc. should be complex and securely stored by the bank.

3.2.4 Authorization

The init.ora parameter _TRACE_FILES_PUBLIC grants file system read access to anyone who has activated SQL tracing. Set this to its default value of *False*.

_TRACE_FILES_PUBLIC=FALSE

Set the init.ora parameter REMOTE_OS_ROLES to *False* to prevent insecure remote roles.

REMOTE_OS_ROLES=FALSE

Set O7_DICTIONARY_ACCESSIBILITY to *False* to prevent users with Select ANY privilege from reading data dictionary tables. False is the default for the 10g database.

O7_DICTIONARY_ACCESSIBILITY = FALSE

3.2.5 **Audit**

This section describes the auditing capabilities available in Oracle database. These recommendations should not have a measurable performance impact.

In init.ora, set AUDIT_TRAIL to DB, OS or TRUE. Consult with the Applications Database Administrator before setting this value to TRUE. When set to OS, the database stores its audit records on the file system:

AUDIT_TRAIL = OS

Set parameter AUDIT_FILE_DEST to the directory where the audit records should be stored. When not set, AUDIT_FILE_DEST defaults to $ORACLE_HOME/rdbms/audit. In this example, the database places audit records in directory E:\logs\db\audit.

AUDIT_FILE_DEST = E:\logs\db\audit

Restart the database for these parameters to take effect.

Note: The database generates some audit records by default, whether or not AUDIT_TRAIL is enabled. For example, Oracle automatically creates an operating system file as an audit record when a user logs in as SYSDBA or as INTERNAL.

Monitoring and auditing database sessions, provides valuable information on database activity and is the only way to identify certain types of attacks (for example, password guessing attacks on an application schema). By auditing database sessions, suspicious connections to highly privileged schemas may be identified.

To audit sessions, login through sqlplus as SYSTEM and issue the following command:

SQL> audit session;

Audit any changes to the standard FCIS database schema or creation of new schemas. As rare events, these changes may indicate inappropriate or malicious activity.

To audit schema changes, login through sqlplus as SYSTEM and issue the following command:

SQL> audit user;

To complete the recommended auditing, enable three other audit events: *create database link, alter system* and *system audit*. The remaining audit options generate significant entries of little value. Auditing these other actions provides little meaningful information.

To audit the other events, login through sqlplus as SYSTEM and issue the following commands:

SQL> AUDIT DATABASE LINK; -- Audit create or drop database links

SQL> AUDIT PUBLIC DATABASE LINK; -- Audit create or drop public database links

SQL> AUDIT SYSTEM AUDIT; -- Audit statements themselves

SQL> AUDIT ALTER ANY ROLE by ACCESS; -- Audit alter any role statements

SQL> AUDIT ALTER DATABASE by ACCESS; -- Audit alter database statements

SQL> AUDIT ALTER SYSTEM by ACCESS; -- Audit alter system statements

SQL> AUDIT CREATE ROLE by ACCESS; -- Audit create role statements

SQL> AUDIT DROP ANY ROLE by ACCESS; -- Audit drop any role statements

SQL> AUDIT PROFILE by ACCESS; -- Audit changes to profiles

SQL> AUDIT PUBLIC SYNONYM by ACCESS; -- Audit public synonyms statements

SQL> AUDIT SYSDBA by ACCESS; -- Audit SYSDBA privileges

SQL> AUDIT SYSOPER by ACCESS; -- Audit SYSOPER privileges

SQL> AUDIT SYSTEM GRANT by ACCESS; -- Audit System grant privileges

Connections to the database as well as SYSDBA and SYSOPER actions (instance startup/shutdown) are always logged to the directory $ORACLE_HOME/rdbms/audit (unless AUDIT_FILE_DEST property is overridden). This file contains the operating system user and terminal ID.

If AUDIT_TRAIL is set to OS, review audit records stored in the file name; in AUDIT_FILE_DEST.

If AUDIT_TRAIL is set to DB, retrieve audit records from the SYS.AUD$ table. The contents can be viewed directly or via the following views:

- DBA_AUDIT_EXISTS
- DBA_AUDIT_OBJECT
- DBA_AUDIT_SESSION
- DBA_AUDIT_STATEMENT
- DBA_AUDIT_TRAIL
- DBA_OBJ_AUDIT_OPTS
- DBA_PRIV_AUDIT_OPTS
- DBA_STMT_AUDIT_OPTS

The audit trail contains a lot of data; begin by focusing on the following:

- Username: Oracle Username.
- Terminal: Machine from which the user originated.
- Timestamp: Time the action occurred.
- Object Owner: The owner of the object that the user touched.
- Object Name: The name of the object that the user touched.
- Action Name: The action that occurred against the object (INSERT, UPDATE, DELETE, SELECT, EXECUTE).

Archive and purge the audit trail on a regular basis, at least every 90 days. The database connection entries take up significant space. Backup the audit file before purging.

Audit data may contain confidential or privacy related data. Restrict audit trail access appropriately.

It must be noted that auditing features can impose a significant performance overhead. Auditing should thus be limited to the set of items outlined above. Auditing application schema objects should be strictly avoided.

3.2.6 Secure Database Backups

RMAN secure backup should be used to ensure that the backups stolen from your system cannot be restored in another remote system. Additionally, data masking - a feature offered by Oracle Enterprise Manager – can be used to move data from your production environment to a test environment. Both these are very crucial steps towards securing confidential customer data.

The database backups should be stored for the required period as per the regulations and bank's history retention policies. These backups should be securely stored and access should be controlled to authorized users only.

3.2.7 Separation of Roles

It is vital to ensure that roles and responsibilities of database administrators and application users/administrators are clearly segregated. Database administrators should not be allowed to view or access customer data. Oracle Database vault helps to achieve this separation of duty by creating different realms, factors and rule sets. It can enforce policies that prevent a DBA from accessing an application realm. The product has a set of configuration policies that can be directly implemented with database vault. Implementation specific requirements can be imposed over and above these.

3.2.8 Securing Audit Information

Oracle Audit vault is an audit solution that consolidates, detects, monitors, alerts and reports n audit data for security auditing an compliance. Oracle Audit vault provides mechanisms to collect audit data from various oracle database. It helps to consolidate audit data from multiple systems into a single centralized repository. Thus, DBA's of individual systems will not be able to tamper with audit information of their respective databases.

3.2.9 Advanced Security

Oracle Advanced Security provides industry standards-based data privacy, integrity, authentication, single sign-on, and access authorization in a variety of ways. Sensitive information that is stored in your database or that travels over enterprise networks and the Internet can be protected by encryption algorithms. An encryption algorithm transforms information into a form that cannot be deciphered without a decryption key. Oracle Advanced Security supports multiple industry standard encryption algorithms such as RC4, DES3 and Triple-DES. To ensure the integrity of data packets during transmission, Oracle Advanced Security can generate a cryptographically secure message digest using SHA- hashing algorithms and include it with each message sent across a network.

3.3 Database Operating Environment Security

3.3.1 Overview

The environment in which Oracle Applications run contributes to or detracts from overall system security. This section contains security recommendations for tightening Oracle file system security along with more general advice for overall system hardening.

3.3.2 Hardening

- The directory $ORACLE_HOME/bin contains Oracle executables. Check that the operating system owner of these executables matches the operating system user under which the files have been installed. A typical mistake is to install the executables in user oracle's directory but owned by root.
- Prevent remote login to the Oracle (and root) accounts. Instead, require that legitimate users connect to their own accounts and su to the Oracle account. Better yet, use sudo to restrict access to executables.

Refer to the product installation documentation for the complete instructions on setting file permissions.

On UNIX systems:

- Set the permissions on $ORACLE_HOME/bin to 0751 or less. Set all other directories in $ORACLE_HOME to 0750 or less. Note, this limits access to the Oracle user and its groups (probably DBA).
- Set file permissions for listener.ora and sqlnet.ora to 0600.
- Set file permissions for tnsnames.ora to 0644.
- Ensure that the owner, group and modes of the Oracle files created upon installation are set to allow minimum privilege. The following commands make this change. Note, the group and owner are for illustration only, the correct group and owner should be substituted.

```
$chgrp -R    <dba>      $ORACLE_HOME
$chown -R    <oracle>   $ORACLE_HOME
```

- Review owners and groups when cloning a database
- Protect the $ORACLE_HOME/rdbms/admin directory including catalog.sql, catproc.sql and backup scripts.
- Secure scripts containing usernames and passwords
- Verify that set user id (SUID) and set group id (SGID) are not set on binaries. In general, Oracle recommends that the SUID and SGID bits to be removed from binaries shipped by Oracle.

On windows systems, NTFS must be used. The FAT/FAT32 file system provides no security.

The database and applications require that the underlying operating system provide certain services.

- Electronic Mail

FCIS may require access to a SMTP Mail Transfer Agent (SMTP MTA) typically send mail. This is required for outbound emails, typically notifications from FCIS (if this feature is desired by the financial institution). If possible, restrict access to the operating system users who absolutely need the mail facility from the shell.

- Remote Access

Use secure shell (ssh) to access middle-tier and database hosts. This replaces telnet, rsh, rlogin, rcp and ftp.

The following services may provide operational convenience:

- NTP (Network Time Protocol) – for synchronizing the clock on the UNIX hosts to provide accurate audit records and simplify trouble-shooting.
- CRON – for operating system cleanup and log file rotation

3.3.3 Authentication

Good security requires secure accounts.

- Make sure that all OS accounts have a non-guessable password. To ensure that the passwords are not guessable, use crack or john-the-ripper (password cracking tools) on a regular basis. Often, people use passwords associated with them: license plate numbers, children's names or a hobby. A password tester may check for these. In addition, change passwords from time to time.
- Automatically disable accounts after several failed login attempts.
- .netrc files weaken security.
- The fewer people with root access, the easier it is to track changes.
- The root password must be a strong, non-guessable password. In addition, change the root password every three (3) months and whenever an administrator leaves company. Always logout of root shells; never leave root shells unattended.
- Limit root to console login, only (specified in /etc/security).
- Root, and only root, should have UID 0.
- Check root '.*' files for security holes. The root '.*' files SHOULD have 700 or 600 permissions
- umask for root is 022 (rwxr-xr-x). A umask of 077 (rwx------) is best, but often not practical
- To avoid trojan horse programs, always use full pathnames including aliases. Root should NEVER have "." in path.
- NEVER allow non-root write access to any directories in root's path.
- If possible, do not create root's temporary files in publicly writable directories.

Do not share user accounts. Remove or disable user accounts upon termination. Disable login for well known accounts that do not need direct login access (bin, daemon, sys, uucp, lp, adm). Require strong passwords and, in some cases, a restricted shell.

It is hard to imagine what kind of guests should have access to a production system. For this reason do not allow guest access.

3.3.4 Authorization

Only run NFS as needed, apply latest patches. When creating the /etc/exports file, use limited access flags when possible (such as readonly or nosuid). By using fully qualified hostnames, only the named host may access the file system.

Device files /dev/null, /dev/tty and /dev/console should be world writable but NEVER executable. Most other device files should be unreadable and non-writable by regular users.

Always get programs from a known source. Use a checksum to verify they have not been altered.

Create minimal writable file systems (esp. system files/directories). Limit user file writes to their own directories and /tmp. Add directories for specific groups. Limit important file access to authorized personnel. Use setuid/setgid only where absolutely necessary.

3.3.5 Maintenance

Good security practice does not end after installation. Continued maintenance tasks include:

- Install the latest software patches.
- Install latest operating system patches.
- Verify user accounts - delete or lock accounts no longer required.

- Run security software and review output.
- Keep up to date on security issues by subscribing to security mailing lists, reading security news groups and following the latest security procedures.
- Implement trusted file systems like NIS, NIS+ or others such as HP-UX trusted system.
- Test the system with tools like NESSUS (network security) and CRACK (password checker).
- Install Tripwire to detect changes to files
- Monitor log files including btmp, wtmp, syslog, sulog, etc. Consider setting up automatic email or paging to warn system administrators of any suspicious behaviour. Also check the snort logs.

The environment in which Oracle Applications run contributes to or detracts from overall system security. This section contains security recommendations for tightening Oracle file system security along with more general advice for overall system hardening.

3.3.6 Access Prevention

Authorized Access: Only people with a “need to know” or a legitimate administrative purpose should be allowed any form of access to production database.

Access Logging: All access to production databases must be logged with a specific user ID that maps to a specific individual, either staff or a vendor, including administrator login. There should be separate/dedicated DB user created for Flexcube application which should not be shared or used for any other purpose.

Access Monitoring: Monitoring should be done to track non application sessions into the database and activities performed in that session. Guidelines on monitoring tool can be referred from Oracle documentation or Oracle DBA team.

Providing Access: While providing production access to any individual, process of authorization by higher level management with proper justification should be followed. The access should be controlled by timelines that certain user access will be expired after specific period and should go through renewal process to reactivate.

Restricted Access for support: The support consultants (OFSS / OFSS partner / Third party vendor / Bank IT) should have individual IDs created on database with strictly Read-Only access. Any consultant demanding for updateable/full access should be reported to senior management of respective vendor.

Restricted Access for reporting tools: All third party tools using Production database for reporting purpose (Like BO reports) should access it via a Read-only access ID meant for specific reporting tool.

Reports from Backup Schema : As much as possible, reports should be generated on backup schema and not on production schema.

Restricted Access for interfaces: If third party tool is writing into production database for processing, it should be restricted via APIs wherever applicable. There should be dedicated user created for each distinct interface and that user activity should be tracked to ensure authenticated sessions/activities.

Change Password: Database passwords should be changed at regular intervals at scheduled frequency or event basis. Bank can refer to Oracle FLEXCUBE Password change document attached in appendix section to understand the steps. However, bank is requested to approach to Global Support team before taking up password change activity for the first time.

Production ID restrictions: Inactive DB User Id which are not used for a specific period (say one month) should be disabled and deleted in case of say 3 months of inactivity. Any activation/recreation of such ID should follow standard process/mechanism. Password profile should be created which will automatically take care of disabling the user ids after inactivity for specified time. These are standard recommendations, bank can have their own timelines defined for these activities.

Awareness: The awareness must be created within bank's teams, whoever are having any form of access to production database, to ensure that they do not share their password and do not leave their database session unattended.

Restrict DB Sessions during EOD: Database Sessions using Toad, sql*navigator etc and running heavy queries from those sessions should be avoided during production End of Day process as it creates additional load and may lead to locks if not used properly.

3.3.7 Data Protection

Data Masking: Any production data shared with support consultant (OFSS / OFSS partner / Third Party vendor) should only be shared in masked form. The vital & sensitive information like Customer Name, Customer's personal details, SWIFT address, account title, address, email id should be masked. (Sample data masking scripts are attached in appendix section for reference. Global Support team can provide actual data masking scripts on request that are applicable for your installation). Vendors should be indicated/informed to delete the shared data once the incident is resolved.

Printing of Production data: Printing of production data should be avoided as much as possible and should be printed only when necessary. Printed version of production data should be kept only for required period and destroyed using standard mechanism to avoid it falling into wrong hands. Whenever customer statements are printed, the delivery should be concluded within stipulated period and should be securely stored until then.

Adopt Standard Data Protection Policies: Standard corporate policies like Clean Desk Policy help in strengthening the Data protection. Forming of data controller team to ensure sanity/masking of data before it is handed over for any purpose.

Protected backup: The Backups and storages should ensure labeling and encryption wherever required. The media recycle policy can be adopted to ensure that old unwanted backup tapes/media are not misplaced.

Data Sharing: Ensure NOT TO share data on personal email ids. No part of data should be uploaded through non official web sites. Sharing data with third party vendor, partners, business teams should be done in protected and encrypted form by ensuring key customer data is masked.

3.4 Application Server Security

3.4.1 Overview

This section describes how to secure the Oracle WebLogic Server production environment that hosts the Oracle FLEXCUBE Investor Servicing environment.

3.4.2 Installation of Oracle WebLogic Server

By default, Oracle WebLogic Server is installed with a JDK and several development utilities. These are not required in a production environment.

The installation footprint of Oracle WebLogic Server can be reduced via the following measures:

- During installation of Oracle WebLogic Server, customize the components to be installed. The following components are not required by Oracle FLEXCUBE Investor Servicing in a production environment:
- Oracle WebLogic Workshop
- Web 2.0 HTTP Pub-Sub Server
- Third Party JDBC Drivers (for MySQL and Sybase)
- WebLogic Server examples
- Delete the Pointbase database which is not required for production usage.

3.4.3 Securing the WebLogic Server installation

Once installed, the measures listed below can be employed to secure the WebLogic Server installation.

3.4.3.1 Network perimeter protection

It is highly recommended to employ the use of a firewall (as hardware or software) to lockdown the network access to the WebLogic cluster.

For additional information on planning the firewall configuration for a WebLogic Cluster, refer to the section “Security Options for Cluster Architectures” in the “Using Clusters” guide of the Oracle WebLogic Server documentation.

3.4.3.2 Operating System Users and Groups

It is highly recommended to run the WebLogic Server as a limited user process. The root user account in Unix/Linux and the Administrator account in Windows should not be used to run WebLogic Server since they are privileged user accounts. Other privileged accounts should also not be used to run the WebLogic server.

Hence, it is preferable to create a limited user account say “WebLogic Owner” for running the application server. Additional user accounts are not recommended; in the eventuality, that an additional account is required (say, if the WebLogic owner account is locked out), one of the system administrator accounts can be used to remedy the situation. Having two system administrator accounts is recommended, as it always ensures backup.

3.4.3.3 File System Access to OS Users

Access rights to the Oracle Home, WebLogic Server product directory, and the WebLogic domain directories should be provided only to the “WebLogic Owner” user. Privileged users will anyway have access to the WebLogic Server installation, by default.

Users in the “Others” category can be restricted from reading the afore-mentioned directories.

Ensure that the following files in the WebLogic installation are available only to the WebLogic owner:

- The security LDAP database which is usually located in the WL_HOME\user_projects\domains\ DOMAIN_NAME\servers\SERVER_NAME\data\ldap\ldapfiles directory
- The keystore used in the keystore configuration of the server(s)
- The Root Certificate Authority keystore

Oracle WebLogic Server provides persistent stores for several subsystems, some of which are utilized by Oracle FLEXCUBE Investor Servicing. Ensure that access to the persistent file stores based on files is restricted to the WebLogic owner OS user. The default persistent file store is located in the *data**store**default* directory under the *servername* subdirectory under the WebLogic domain’s root directory. If custom (user-defined) persistence stores have been created, the same restrictions should be applied on the files and directories used by such stores.

3.4.3.4 Usage of Protected Ports

In the case of Oracle WebLogic Server

- Operate WebLogic Server using an unprivileged account, bind to unprotected ports, and use NAT to map protected ports to the unprotected ports.
- Configure WebLogic Server to start with a privileged account, bind to protected ports, and then change the user account to an unprivileged user account. For this purpose, Oracle WebLogic Server on UNIX needs to be configured to have a post-bind user ID or group ID. For additional details, refer to the section “Create and configure machines to run on UNIX” in the “Administration Console Online Help”.

3.4.3.5 Choice of the SSL cipher suite

Oracle WebLogic Server allows for SSL clients to initiate a SSL connection with a null cipher suite. The null cipher suite does not employ any bulk encryption algorithm thus resulting in transmission of all data in clear text, over the wire.

The default configuration of Oracle WebLogic Server is to disable the null cipher suite. Ensure that the usage of the null cipher suite is disabled, preventing any client from negotiating an insecure SSL connection.

Furthermore, for installations having regulatory requirements requiring the use of only ‘high’ cipher suites, Oracle WebLogic Server can be configured to support only certain cipher suites. The restriction can be done in config.xml of the WebLogic domain. Provided below is an example config.xml restricting the cipher suites to those supporting 128-bit symmetric keys or higher, and using RSA for key exchange.

```
....

<ssl>
```

```
      <enabled>true</enabled>

      <ciphersuite>TLS_RSA_WITH_AES_256_CBC_SHA</ciphersuite>

      <ciphersuite>TLS_RSA_WITH_AES_128_CBC_SHA</ciphersuite>



</ssl>

....
```


Note the following:

- Configuration of WebLogic Server to support the above defined cipher suites might also require an additional command line argument to be passed to WebLogic Server, so that a FIPS 140-2 compliant crypto module is utilized. This is done by adding **-Dweblogic.security.SSL.nojce=true** as a JVM argument.
- The restriction on cipher suites needs to be performed for every managed server.
- The order of cipher suites is important – Oracle WebLogic Server chooses the first available cipher suite in the list, that is also supported by the client.
- Cipher suites with RC4 are enabled despite it being second best to AES. This is primarily for older clients that do not support AES (for instance, Microsoft Internet Explorer 6, 7 and 8 on Windows XP).
- Cipher suites using Triple DES (3DES) are not listed since the maximum effective security provided by the algorithm is 112 bits.

3.4.3.6 Usage of WebLogic Connection Filters

Although firewalls restrict the ability of machines to communicate with the WebLogic Server, machines in the data center can still access network services provided by the WebLogic Server.

Configure the Oracle WebLogic Server installation to use connection filters to ensure that only certain machines in the data center can access the WebLogic Server services like HTTP, LDAP, RMI-IIOP etc.

3.4.3.7 Usage of Domain-wide Administration Port for Administrative Traffic

When Oracle WebLogic Server is configured to enable administrative access via the administration port, data is exchanged over SSL, preventing any attacker from sniffing sensitive information about the WebLogic Server configuration.

Furthermore, once the Administration port is enabled, WebLogic Server will serve administration requests on a dedicated port with dedicated resources. A denial service attack mounted on the HTTP/HTTPS channels will not prevent administrators from logging into the WebLogic Server administration console to take corrective actions.

Hence, it is recommended to enable the use of the administration port. Additionally, employ firewall rules or WebLogic Connection Filters to restrict access to the Administration Port to trusted machines from where administrators can log in.

Do note that the Administration Port requires that SSL be enabled on every Managed Server. Additionally, the administration port will be common across all servers in the domain

Further details on configuring the administration port can be found in the “Administration Console Online Help” guide in the Oracle WebLogic Server documentation.

3.4.3.8 Secure the Embedded LDAP port

In a WebLogic Server cluster, restrict access to the embedded LDAP server port only to machines in the WebLogic Server cluster, through the user of connection filters.

3.4.3.9 Disable Remote Access to the JVM Platform MBean Server

The MBean server provided by the JDK (from JDK 5 onwards) provides details in MBeans, containing information about the JVM that is useful for monitoring the JVM process.

Besides the JVM’s platform MBean server, Oracle provides three other MBean servers – the Domain Runtime MBean server, Runtime MBean server and the Edit MBean server. It is possible to configure the Runtime MBean server (that is available on each managed server) as the platform MBean server, allowing JMX clients to access not only the JVM MBeans, but also the WebLogic Server MBeans.

If the Runtime MBean server of WebLogic has been configured as the platform MBean server, enabling remote access creates an access path that is no longer secured by the WebLogic Server Security Framework, but instead by the security features of the Java platform alone. In such a case where remote access to the platform MBean server (and the runtime MBean server) is required, it is recommended that JMX clients access the MBeans via the Runtime MBean server.

Oracle Financial Services recommends that changes once done in this regard, be tested thoroughly for impact on business continuity.

3.4.3.10 Precautions when using SNMP

It is recommended to refer the WebLogic SNMP Management Guide to configure SNMP agents in Oracle WebLogic Server. Due care must be observed over the usage of SNMP v1 and v2 since passwords are sent over clear text in these older version of the protocol. Additional steps required for securing SNMP v3 communication are outlined in the guide.

Oracle Financial Services recommends that changes once done in this regard, be tested thoroughly for impact on business continuity.

3.4.4 Securing the WebLogic Security Service

Address as recommend ensuring the following

3.4.4.1 Enable SSL, but avoid using Demonstration Certificates

Enable the use of SSL so that the servers can be accessed via the SSL listen ports for all supported protocols (including HTTPS).

Oracle WebLogic Server includes demonstration private keys, certificates and trusted certificate authorities that are not intended for use in production. Usage of these keys in production is a security risk due to the free availability of private keys; anyone who has a copy of the WebLogic Server has knowledge of the private keys and can compromise SSL/TLS traffic.

Therefore,

- Use a local CA to issue certificates, or
- Use a root or intermediate CA like VeriSign, Thawte etc. to issue certificates

Oracle Financial Services does not recommend the use of self-signed certificates in production.

Consider avoiding the use of certificates with a MD5 signature; usage of certificates with SHA-1 signatures is recommended. Most root and intermediate CAs have begun phasing out the use of MD5 for signing certificates.

3.4.4.2 Enforce Security Constraints on Digital Certificates

Oracle WebLogic Server performs certificate validation whenever it establishes an outbound SSL connection, or when a two-way SSL connection is established. As part of certificate validation, WebLogic Server checks if the certificate contains the Basic Constraints extension. Ensuring the presence of the Basic Constraints extension will prevent attackers from generating new certificates to aid in website spoofing.

Ensure the check for Basic Constraints extension is enabled, by verifying whether the following line is absent in the WebLogic Server startup command.

```
-Dweblogic.security.SSL.enforceConstraints=off
```

Also verify if any messages have been logged at WebLogic server boot, providing information about the presence of certificates that could be rejected by clients.

3.4.4.3 Ensure that Host Name Verification is Enabled

Oracle WebLogic Server implements host name verification when it acts as a SSL client; this prevents main-in-the-middle attacks from being performed against SSL itself.

It should be noted that the Oracle FLEXCUBE Investor Servicing application deployed on WebLogic Server will establish outbound SSL connections in certain scenarios, for instance, when requests are made to the Oracle BI Publisher server. In such an event, Oracle WebLogic Server will behave as a SSL client.

Oracle WebLogic Server will behave as a SSL client in several scenarios besides the outbound SSL requests made by applications deployed on Oracle WebLogic Server. For instance, managed servers will establish SSL connections with the Admin server at boot time. Hence, it is recommended to ensure that host name verification is enabled in Oracle WebLogic Server, which happens to be the secure default.

Oracle Financial Services highly recommends the usage of certificates that will pass verification. Oracle Financial Services also recommends against the usage of demonstration certificates in production. It should be noted that usage of demonstration certificates in a testing or development environment containing a multi-server WebLogic cluster, will result in boot failures for managed servers.

3.4.4.4 Impose Size and Time Limits on Messages

Consider enforcing constraints on size and on the amount of time taken for a message to arrive at the server. This will ensure protection against denial-of-service attacks against WebLogic Server. Additional details are provided in the Oracle WebLogic Server documentation, in the guide “Securing a Production Environment”, and also in the “Administration Console Online Help”.

Oracle Financial Services recommends that changes, once done in this regard, be tested thoroughly for impact on business continuity – it is quite possible that WebLogic Server receive valid messages that are large enough to be considered as an attack, when such is not the case.

3.4.4.5 Restrict the Number of Open Sockets

Consider limiting the number of sockets opened by WebLogic Server, to prevent some forms on denial-of-service attacks. Further details are available in the Oracle WebLogic Server documentation, in the guide “Securing a Production Environment”, and also in the “Administration Console Online Help”.

Oracle Financial Services recommends that changes, once done in this regard, be tested thoroughly for impact on business continuity – the number of sockets opened is dependent entirely on system load, which is bound to vary across time, and also across installations.

3.4.4.6 Configure WebLogic Server to Manage Overload

Oracle WebLogic Server can be configured to detect, avoid and recover from overload conditions. Configuring WebLogic Server to manage overload conditions allows for WebLogic Server administrators to connect to it, and take remedial actions. Further details on this topic are available in the Oracle WebLogic Server documentation, in the guide “Securing a Production Environment”, and also in the “Administration Console Online Help”.

Oracle Financial Services recommends that changes, once done in this regard, be tested thoroughly for impact on business continuity – the definition of an overload condition depends on the system capabilities; therefore, overload conditions are bound to be defined differently for machines of differing capabilities.

3.4.4.7 User Lockouts and Login Time Limits

The Oracle WebLogic Server guide on “Securing a Production Environment” has a section on configuring user lockouts and login time limits to prevent attacks on user accounts. In general, Oracle FLEXCUBE Investor Servicing does not utilize the WebLogic Security Service for managing FLEXCUBE Investor Servicing user accounts.

Therefore, changes recommended by the WebLogic Server guide should be applied only after assessing the impact on production. The changes applied would be suitable for accounts managed by Oracle WebLogic Server. Note that the WebLogic Server Online Console guide will reference “Compatibility Security” which is deprecated in Oracle WebLogic Server 10.3.

Generally, Oracle FLEXCUBE Investor Servicing employs its own protection mechanisms with respect to user lockouts.

3.4.4.8 Enable Configuration Auditing

Configuration auditing can be enabled to ensure that changes to any WebLogic resource configuration in the WebLogic domain are audited. Enabling this option also allows for auditing of management operations performed by a user on any WebLogic resource.

For additional details, refer to the “Administration Console Online Help”, and the “Configuring WebLogic Security Providers” section in the “Securing WebLogic Server” guide of the Oracle WebLogic Server documentation.

Note that enabling configuration auditing will affect the performance of the system, even though auditing may be enabled for auditing a few events (including configuration changes).

3.4.4.9 System Administrator Accounts

Create at least two system administrator accounts (WebLogic user accounts) for administration of the WebLogic server. The first administrator account will be created when the WebLogic domain is created. Create the second account with the Admin security role.

Provide unique names to the administrator accounts that cannot be easily guessed. Oracle Financial Services discourages naming the WebLogic administrator account as ‘weblogic’ with a password of ‘weblogic’.

Again, having two system administrators ensures that at least one system administrator has access to the WebLogic server in the event of the other being locked out.

4. Installation

4.1 Securing the Oracle FLEXCUBE Investor Servicing Application

The following guidelines serve to secure the Oracle FLEXCUBE Investor Servicing application deployed on Oracle WebLogic Server.

4.1.1 Enforce the Usage of SSL

The FLEXCUBE Installer allows a deployer to configure FLEXCUBE Investor Servicing such that all HTTP connections to the FLEXCUBE Investor Servicing application are over SSL/TLS. In other words, all HTTP traffic in the clear will be prohibited; only HTTPS traffic will be allowed. It is highly recommended to enable this option is a production environment, especially when WebLogic Server acts as the SSL terminator.

Ensure that the following snippet of code is present in the web.xml file of the FLEXCUBE IS web module i.e. in FCJNeoWeb.war.

```
<security-constraint>

	<web-resource-collection>

		<web-resource-name>FLEXCUBE UBS</web-resource-name>

		<description>All endpoints secured</description>

		<url-pattern>/*</url-pattern>

	</web-resource-collection>

	<user-data-constraint>

		<transport-guarantee>CONFIDENTIAL</transport-guarantee>

	</user-data-constraint>

</security-constraint>
```

4.1.2 Setting up Secure Flag for Cookies

If the secure flag is set on a cookie, then browsers will not submit the cookie in any requests that use an unencrypted HTTP connection, thereby preventing the cookie from being trivially intercepted by an attacker monitoring network traffic.

Below configuration has to be ensured in weblogic.xml within the deployed application ear.

1. Cookies are set with Http only as true
2. Cookie secure flag set to true
3. Cookie path to refer to deployed application

<wls: session-descriptor>

<wls: cookie-http-only>true</wls: cookie-http-only>

</wls: session-descriptor>

<wls: session-descriptor>

<wls: cookie-secure>true</wls: cookie-secure>

<wls: url-rewriting-enabled>false</wls: url-rewriting-enabled>

</wls: session-descriptor>

<session-descriptor>

<cookie-name>JSESSIONID</cookie-name>

<cookie-path>/<DeployedApplicationPath></cookie-path>

<cookie-http-only>true</cookie-http-only>

<cookie-secure>true</cookie-secure>

<url-rewriting-enabled>false</url-rewriting-enabled>

</session-descriptor>

Always make sure Cookies are set with always Auth Flag enabled by default for WebLogic server and also recommended to apply the weblogic patch 10.3.5 for versions using below weblogic 10.3.5 to reflect the above changes.

4.1.3 Two-way SSL Connection

A two-way SSL is used when the server needs to authenticate the client. In a two-way SSL connection the client verifies the identity of the server and then passes its identity certificate to the server. The server then validates the identity certificate of the client before completing the SSL handshake.

In order to establish a two-way SSL connection, need to have two certificates, one for the server and the other for client.

For Oracle FLEXCUBE Investor Servicing Solutions, need to configure a single connector. This connector is related to SSL/TLS communication between host or browser and the branch which uses two-way authentication.

For details on implementation of Two-way SSL process, refer to the document available for FLEXCUBE < ***SSL_OR_TLS_ Configuration.doc****>.*

4.1.4 Ensure the Servlet Servlet is Disabled

Oracle FLEXCUBE Investor Servicing does not use the ServletServlet to create default mappings for servlets. All servlets are directly mapped to the required URLs.

Ensure that the following code snippet (or a similar one that uses the weblogic.servlet.ServletServlet) *does not exist* in the web.xml of the Oracle FLEXCUBE Investor Servicing web application.

```
<servlet>
  <servlet-name>ServletServlet</servlet-name>
  <servlet-class>weblogic.servlet.ServletServlet</servlet-class>
</servlet>

<servlet-mapping>
  <servlet-name>ServletServlet</servlet-name>
  <url-pattern>/myservlet/*</url-pattern>
</servlet-mapping>
```

4.1.5 Session time out and Token Management

Session timeout represents the event occurring when a user do not perform any action on a web site during a interval (defined in application). The event, on server side, change the status of the user session to 'invalid' (ie. "not used anymore") and instruct the Application/web server to destroy it (deleting all data contained into it). Application allows defining the session time out.

The default value for session time out is 30 minutes.

The entire subsequent request within the session will be having the Authenticated and Cross-site request forgery tokens .Every request send to the application from the browser is validated against the IsAuthenticated attribute and Cross-site request forgery token.

4.2 Securing the Gateway Services

4.2.1 Overview

Different applications deployed on disparate platforms and using different infrastructure need to be able to communicate and integrate seamlessly with Oracle FLEXCUBE in order to exchange data. The Oracle FLEXCUBE Integration Gateway will cater to these integration needs.

The integration needs supported by the Gateway can be broadly categorized from the perspective of the Gateway as follows:

- Inbound application integration – used when any external system needs to add, modify or query information within Oracle FLEXCUBE
- Outbound application integration – used when any external system needs to be notified of the various events that occur within Oracle FLEXCUBE.

4.2.2 Inbound Application Integration

Oracle FLEXCUBE Inbound Application Gateway provides XML based interfaces thus enhancing the need to communicate and integrate with the external systems. The data exchanged between Oracle FLEXCUBE and the external systems will be in the form of XML messages. These XML messages are defined in FCIS in the form of XML Schema Documents (XSD) and are referred to as 'FCIS formats'

FCIS Inbound Application Integration Gateway uses the Synchronous and Asynchronous Deployment Pattern for addressing the integration needs.

The Synchronous Deployment Pattern is classified into the following:

- Oracle FLEXCUBE EJB Based Synchronous Inbound Application Integration Deployment Pattern
- Oracle FLEXCUBE Web Services Based Synchronous Inbound Application Integration Deployment Pattern
- Oracle FLEXCUBE HTTP Servlet Based Synchronous Inbound Application Integration Deployment Pattern

Asynchronous Deployment Pattern is:

- Oracle FLEXCUBE MDB Based Asynchronous Inbound Application Integration Deployment Patten

4.2.2.1 EJB Based Synchronous Deployment Pattern

The Enterprise Java Beans (EJB) deployment pattern will be used in integration scenarios where the external system connecting to Oracle FLEXCUBE is 'EJB literate', i.e., the external system is capable of interacting with Oracle FLEXCUBE based upon the EJB interface. In this deployment pattern, the external system will use the RMI/IIOP protocol to communicate with the Oracle FLEXCUBE EJB.

In this deployment pattern the EJB displayed by Oracle FLEXCUBE will be a stateless session bean. The actual request will be in the form of an XML message. After the necessary processing is done in Oracle FLEXCUBE based on the request, the response is returned to the external system as an XML message. The transaction control for the processing will stay with the Oracle FLEXCUBE EJB.

4.2.2.2 Web Services Based Synchronous Deployment Pattern

The web services deployment pattern will be used in integration scenarios where the external system connecting to Oracle FLEXCUBE wants to connect using standards-based, inter-operable web services.

This deployment pattern is especially applicable to systems which meet the following broad guidelines:

- Systems that are not 'EJB literate', i.e., such systems are not capable of establishing connections with Oracle FLEXCUBE based upon the EJB interface; and/or
- Systems that prefer to use a standards-based approach

In this deployment pattern, the external system will use the SOAP (Simple Object Access Protocol) messages to communicate to the Oracle FLEXCUBE web services.

The services displayed by Oracle FLEXCUBE are of a 'message based' style, i.e., the actual request will be in the form of an XML message, but the request will be a 'payload' within the SOAP message. After the necessary processing is done in Oracle FLEXCUBE based on the request, the response is returned to the external system as an XML message which will be a 'payload' within the response SOAP message. The transaction control for the processing will stay with the Oracle FLEXCUBE.

4.2.2.3 HTTP Servlet Based Synchronous Deployment Pattern

The HTTP servlet deployment pattern will be used in integration scenarios where the external system connecting to Oracle FLEXCUBE wants to connect to Oracle FLEXCUBE using simple HTTP messages.

This is especially applicable to systems such as the following:

- Systems that are not 'EJB literate', i.e., are not capable establishing a connections with Oracle FLEXCUBE based upon the EJB interface; and/or
- Systems that prefer to use a simple http message based approach without wanting to use SOAP as the standard

In this deployment pattern, the external system will make an HTTP request to the Oracle FLEXCUBE servlet.

For this deployment pattern, Oracle FLEXCUBE will display a single servlet. The actual request will be in the form of an XML message. This XML message is embedded into the body of the HTTP request sent to the Oracle FLEXCUBE servlet. After the necessary processing is done in Oracle FLEXCUBE based on the request, the response is returned to the external system as an XML message which is once again embedded within the body of the response HTTP message. The transaction control for the processing will stay with the Oracle FLEXCUBE.

4.2.2.4 MDB Based Asynchronous Deployment Pattern

The MDB deployment pattern is used in integration scenarios where the external system connecting to Oracle FLEXCUBE wants to connect to Oracle FLEXCUBE using JMS queues.

This is especially applicable to systems such as the following:

- Systems that prefer to use JMS queues based approach without wanting to wait for the reply

Here external system sends messages in XML format to request queue on which an MDB is listening. When a message arrives on the queue, it is picked up for processing. After the necessary processing is done in Oracle FLEXCUBE, based on the request, the response is sent to the response queue as an XML message.

Refer Resource_Creation documents for more information.

4.2.3 Outbound Application Integration

The Outbound Application Integration is also called the Oracle FLEXCUBE Notify Application Integration layer. This application layer sends out notification messages to the external system whenever events occur in Oracle FLEXCUBE.

The notification messages generated by FCIS on the occurrence of these events will be XML messages. These XML messages are defined in FCIS in the form of XML Schema Documents (XSD) and are referred to as 'FCIS formats'

4.2.4 External System Maintenance

An external system needs to be defined that will communicate with the Oracle FLEXCUBE Integration Gateway. Below are the details requiring inputting while creating the external system.

External System-- Specify a name for the external system. This should be the same as the Source in an incoming message.

Description - Specify a brief description for the External System.

Request-- A way needs to be defined in which the external system should correlate its request message with the response message. Message ID can be chosen of a request message as the Correlation ID in the response message. Alternatively, user can choose Correlation ID of a request message and maintain it as the Correlation ID of the corresponding response message.

Request Message--User can choose the Request message to be 'Full Screen' or 'Input Only'. If you select 'Full Screen' as the request message, the response message will also display 'Full Screen'.

Response Message--User can choose the Response message to be 'Full Screen' or 'Record Identification Msg'.
Default Response Queue-- You can define a response queue for each of the In Queue's through which the External System will communicate with Oracle FLEXCUBE. Define a valid queue name as the Default Response Queue.

Dead Letter Queue--If the messages received are non-readable, such messages are directed to Dead Letter Queue defined for the external system.

XSD Validation Required-- Check this box to indicate if the request message should be validated against its corresponding XSD.

Register Response Queue Message ID--Check this box to indicate if the message ID provided by the Response Queue should be logged when a response message is posted into the queue.

4.2.5 Accessing Services and Operations

In a message it is mandatory to maintain a list of Service Names and Operation Codes. This information is called Gateway Operations.

A combination of every such Service Name and Operation Code is mapped to a combination of Function ID and Action. Every screen in Oracle FLEXCUBE is linked with a function ID. This information is called Gateway Functions.

User can gain access to an external system using the Gateway Functions. The Function IDs mapped in Gateway Functions should be valid Function IDs maintained in Oracle FLEXCUBE. Hence, for every new Service or Operation being introduced, it is important that you provide data in Gateway Operations and Gateway Functions.

4.2.6 Gateway Password Generation Logic for External System Authentication

As a secure configuration password authentication should be enabled for the external system maintained. The same can be verifying in External system detail screen level.

Once these features enable, system will validate for Encrypted password as part of every request sent by the External System.

The Message ID which is present as part of the header in Request XML, is considered as hash. External System generates an unique Message ID, which is functional mandatory field in the header. Create a Message Digest with SHA-512 algorithm.

The hash created from the previous step and the password in clear text together is encrypted in DESede encryption method. Apply Base64 encoding to encrypted value and send to the Oracle FLEXCUBE gateway.

4.3 Securing the Web Services by using OWSM

Oracle FLEXCUBE Investor Servicing supports to the WebLogic Server WS-Policies for enforcing security for Web services. Customer can implement any Oracle WSM WS-Security policies and use them with WebLogic Web services.

The Oracle WSM policies are documented in the Oracle Fusion Middleware Security and Administrator's Guide for Web Services < http://docs.oracle.com/cd/E21764_01/web.1111/b32511/toc.htm>

5. Post-Installation

5.1 Desktop Security

5.1.1 Application of Security Patches

Oracle Financial Services highly recommends the following:

- Browsers should be upgraded whenever newer versions are released, for they often include new security features. Additionally, in-built security features of browsers should not be turned off.
- Security patches issued by the Operating System vendor should be applied regularly.
- Updates to anti-virus software and anti-spyware programs should be applied regularly.
- Security Updates to other environmental software like Microsoft Core XML Services (MSXML) should be applied regularly.

Additionally, it is recommended that major upgrades such as browser upgrades and Operating System service packs be tested for impact on business continuity.

5.1.2 Hardening the Browser

Oracle FLEXCUBE Investor Servicing is certified for usage in different browsers. Pls refer the respective release documents on the versions of browsers on which FLEXCUBE Investor Servicing has been certified Each of these browsers provide recommendations from a security perspective and customers are encouraged to employ the recommendations provided by them.

In all browsers, it is recommended to enable the popup blocker with a specific rule to disable popup-blocking for the FLEXCUBE Investor Servicing web application.

Please refer < Client_Browser_Settings.doc > for more information. Hardening Internet Explorer

For Internet Explorer specifically, Microsoft has provided guidance for enhancing Internet Explorer security in the following documents for the respective versions of the browsers:

- Internet Explorer 7 Desktop Security Guide
- Internet Explorer 8 Desktop Security Guide

Customers are encouraged to employ the recommendations provided by Microsoft in the above mentioned guides.

Among the guidelines provided in these documents, Oracle Financial Services specifically recommends the following settings to all customers of FLEXCUBE Investor Servicing:

- Certificate Security - Ensure the usage of SSL 3.0 and TLS 1.0. Disable SSL 2.0 as it is an insecure protocol.
- Privacy Settings - Set Form autocomplete options to Disabled. This will prevent inadvertent caching of data keyed by users.

Application of the following recommendations from Microsoft is not recommended:

- Privacy Settings - Empty Temporary Internet Files Folder When Browser is closed – Oracle FLEXCUBE Investor Servicing relies heavily on client-side caching performed by Internet Explorer using this folder. The application will behave slowly after this setting is enabled, since the browser will download resources from the server after every browser restart. Hence, it is not recommended to enable this setting. It should be noted that the details of transactions performed by the FLEXCUBE Investor Servicing users are not cached in the Temporary Internet Files folder (irrespective of this setting).
- Other Security Recommendations - Do not Save encrypted pages to disk – By default, Internet Explorer stores both encrypted and unencrypted content in the Temporary Internet Files folder. Enabling this setting is bound to cause performance issues (especially when FLEXCUBE Investor Servicing is accessed over HTTPS), since the browser will no longer cache resources. As stated before, details of transactions performed by users will not be cached in the Temporary Internet Files folder (irrespective of this setting).

5.1.3 Terminal Lockout Policy

Oracle Financial Services recommends that a terminal lockout policy be put in place to automatically lockout unattended PC sessions after a certain duration. This is primarily because Oracle FLEXCUBE Investor Servicing will not lock out the browser session, although it does expire the browser session after certain period of inactivity. Users may however be able to access unattended sessions while the FLEXCUBE Investor Servicing user is still logged in. Hence, organizations are expected to set a corporate policy for handling unattended PC sessions; it is recommended to enable the feature to lock workstations, or to enable password-protected screensavers.

5.2 Oracle FLEXCUBE Investor Servicing Controls

5.2.1 Overview

This chapter describes the various programs available within Oracle FLEXCUBE, to help in the maintenance of security.

Access to the system is possible only if the user logs in with a valid ID and the correct password. The activities of the users can be reviewed by the Security Officer in the Event Log and the Violation Log reports.

5.2.2 Disable Logging

It is recommended that the debug logging facility of the application be turned off, once the system is in production. This is achieved by updating the property file of the application via the Oracle FLEXCUBE Investor Servicing Installer.

The above described practice does not disable logging performed by the application in the database tier. This can be disabled by running the lockdown scripts provided. The lockdown scripts will disable logging across all modules and across all users in the system.

5.2.3 Audit Trail Report

A detailed Audit Trail is maintained by the system on all the activities performed by the user from the moment of login. This audit trail lists all the functions invoked by the user, along with the date and time. The program reports the activities, beginning with the last one. It can be displayed or printed. The records can be optionally purged once a printout is taken. This program should be allotted only to the Security Officer.

5.2.4 Security Violation Report

This program can be used to display or print the Violation Report. The report gives details of exceptional activities performed by a user during the day. The difference between the Violation Report and the Audit Trail is that the former gives details of all the activities performed by the users during the day, and the latter gives details of exceptional activities, for e.g. forced password change, unsuccessful logins, User already logged in, etc. The details given include:

- Time
- The name of the operator
- The name of the function
- The ID of the terminal
- A message giving the reason for the login

The system gives the Security reports a numerical sequence. The Security Report includes the following messages:

5.2.4.1 Sign-on Messages

Message	Explanation
User Already Logged In	The user has already logged into the system and is attempting a login through a different terminal.

Message	Explanation
User ID/Password is wrong	An incorrect user ID or password was entered.
User Status is Locked. Please contact your System Administrator	The user profile has been disabled due to an excessive number of attempts to login, using an incorrect user ID or password. The number of attempts could have matched either the successive or cumulative number of login failures (configured for the system).

5.2.5 Display/Print User Profile

This function provides an on-line display / print of user profiles and their access rights. The information includes:

- The type (customer / staff)
- The status of the profile - enabled or disabled or on-hold
- The time of the last login
- The date of the last password /status change
- The number of invalid login attempts
- The language code / home module of the user

5.2.6 Clear User Profile

A user ID can get locked into the system due to various reasons like an improper logout or a system failure. The Clear User Profile function can be run by another user to reset the status of the user who got locked in. This program should be used carefully and conditionally.

5.2.7 Change User Password

Users can use this function to change their passwords. A user password should contain a minimum of six characters and a maximum of twelve characters (both parameterizable). It should be different from the current and two previous passwords. The program will prompt the user to confirm the new password when the user will have to sign-on again with the new password.

5.2.8 List of Logged-in Users

The user can run this program to see which users are in use within Oracle FLEXCUBE at the time the program is being run. The information includes the following:

- The ID of the terminal
- The ID of the user
- The login time

5.2.9 Change Time Level

Time levels have to be set for both the system and the users. Ten time levels are available, 0 to 9. Restricted Access can be used to set the Users time level. The Change Time Level function can be used to do the same for the module. A user will be allowed to sign-on to the system only if his/her time level is equal to or higher than the system time level. This concept is useful because timings for system access for a user can be manipulated by increasing the system time level. For e.g. the End of Day operators could be allotted a time level of 1, and the users could be allotted a time level of 0. If the application time-level is set at 1 during End of Day operations, only the End of Day operators will have access to the application. The other users will be denied access.

5.2.10 Authentication & Authorization

First, only authorized users can access the system with the help of a unique User ID and a password. Secondly, a user should have access rights to execute a function.
The user profile of a user contains the User ID, the password and the functions to which the user has access. FLEXCUBE operation such as new, copy, query, unlock etc will be enabled based on function rights available for the user. The function rights will be checked for each operation performed by the user.

Administrator can define the maximum number of unsuccessful attempts after which a User ID should be disabled. When a User ID has been disabled, the Administrator should enable it. The password of a user can be made applicable only for a fixed period. This forces the user to change the password at regular intervals thus reducing security risks. Further, Administrator can define passwords that could be commonly used by a user as Restrictive Passwords at the user, user role and bank level. A user cannot use any password that is listed as a Restrictive Password at any of these levels.

Provided user has opted for the SSO Enabled option at bank level, user can log in from an LDAP (Oracle Internet Directory) external system into Oracle FLEXCUBE Investor Servicing Solutions.
After successful authentication and authorization of the user is carried out by the LDAP (Oracle Internet Directory), a request is forwarded to gain access into Oracle FLEXCUBE Investor Servicing Solutions without specifying Oracle FLEXCUBE user id and password.

5.2.11 Role Based Access Controls

Application level access has implemented via the Security Management System (SMS) module. SMS supports “ROLE BASED” access of Screens and different types of operations.

5.2.12 Access controls like module level

User can indicate the modules from where a user can operate in the Restricted Access screen (function-ID).

5.2.13 Maker – Checker

Application supports dual control methodology, wherein every operation performed has to be authorized by another user with the requisite rights.

5.2.14 User Management

FLEXCUBE Investor Servicing enables creation of users through SMDUSRDF UI. On authorization of the newly created user, the credentials are automatically mailed to the user's email id. This reduces the risk of password been known to the administrator, who creates users for the bank.

User is forced to change the password on first login. The password supplied is hashed iteratively after being appended with a randomly generated salt value. Hashing algorithm used is of SHA-2 family and above.

User privileges are maintained by Roles. Roles definition is captured via another UI. These roles are mapped to a user in the SMDUSRDF UI. Basing on these user- roles mapping the user will have access to different modules in FLEXCUBE.

5.2.14.1 Credential Over mail

To enable this feature mail server details needs to be provided at the time of property file creation .Below are the required parameters

- Host Server
- User ID
- User Password
- JNDI Name

Also mail session configuration required in Application server. Sample details for creating a mail session are listed below:

Name: FCISMailSession

JNDI Name: mail/FCISMail (The same need to be maintained in property file creation.)

Java Mail Properties for SMTPS protocol:

mail.host=<HOST_MAIL_SERVER>

mail.smtps.port=<SMTPS_SERVER_PORT>

mail.transport.protocol=smtps

mail.smtps.auth=true

mail.smtps.host==<HOST_SMTPS_MAIL_SERVER>

For details on configuration of Mail Session *process, refer to the document < Resource_Creation_WL.doc for weblogic or Resource_Creation_WAS.doc for websphere >.*

5.2.15 Access Enforcement

Access management in FLEXCUBE can be done in four steps.

1. Module level— in such a case the user cannot view even the menu list of the FCIS when he tries to login into the restricted module. Thus, no transactions could be performed

2. Roles wise—as described above basing on the user-roles mapping, the user can access different modules in FCIS. For an example, a bank clerk will have access to customer creation, account opening, term-deposits opening and liquidation screens, but he will not have access to SMDUSRDF UI, which is for user creation.
3. Function-ID wise—here, the user can be restricted to launch even the UI on clicking on the menu list.
4. Product/ Account class wise— here, the user can be prevented access to certain account classes or products. This will disable him from creating any accounts or transactions using those prevented account class and product respectively.

5.2.16 Information Flow Enforcement

Information flows from GUI to the application server in form of request XMLs. There are validations in place to validate the request XMLs. Malicious data entry are filtered off from further processing when found in the body of the XML. Frontend Java classes calls the backend PLSQL packages for further processing. This, PLSQL level validations are in place in the database server. Exclusive use of bind variables and calls to Oracle's DBMS_ASSERT package does the sanitization of the data.

All request URL's are sanitized properly and responses encoded so as to avoid any scripting injection.

5.2.17 Separation of Duties

Login into the FCIS application is majorly controlled by enforcing Roles for the user. As per the roles assigned, the user is able to access the functionalities. As for an example, if the user is assigned the SMS role, he will be able to access the SMS related GUIs only. He cannot create or view a customer or customer accounts or even a monetary transaction.

5.2.18 Least Privilege

FLEXCUBE by default assigns no roles to a user. This is zero privilege from a functional perspective. With no roles assigned, a user cannot view the menu list as everything will appear to be blank. The system administrator has to explicitly map the roles to a user he creates. If it's a copied user then the new user will have the roles assigned to the user from which it has been copied. Nevertheless, it's recommended to map users to specific roles as per his job grade, designation etc.

5.2.19 Continuous Monitoring

FLEXCUBE has history tables to record /archive every time a user logs in and logs out (session expires). It also has history table to record all internal monetary transactions. Also it has message browsers, a user-friendly GUI, to capture/monitor both incoming and outgoing transaction messages.

More on the security front, FLEXCUBE in conjunction with Oracle Access Manager (OAM) has been tested to be working extremely fine. The login was successful using the SSO implementation. Also, the use of HP Web Inspect, a dynamic application security testing software for assessing security vulnerabilities is a good option. This tool is an automated and configurable web-application security-testing tool that mimics real-world hacking techniques and attacks and works fairly good on FCIS.

Customers even have used CA Site Minder which provides the enterprise-class secure Single Sign-On (SSO) and Web access management one needs to authenticate users and control access to Web applications and portals. Across Internet and intranet applications, it enables the secure delivery of essential information and applications to users, partners, suppliers, and customers via secure SSO.

5.2.20 Information System Backup

The below mentioned following points are true for the topic. These recommendations may yield better results.

1. Database related files viz., data files, control files, redo-logs, archived files, init.ora, config.ora etc should be taken at the end of the day.
2. On-line backup of archived redo-log files to be done periodically.
3. Complete export of database and the application should be done atleast once in a week and this can be stored off-site
4. Complete backup of the Oracle directory (excluding the database related files) periodically.
5. When the database is huge, incremental exports(that is delta or differential exports) and on-line tablespace backups are recommended.
6. RMAN secure backup should be used to ensure that the backups stolen from the production/deployed system cannot be restored in another remote system. Additionally, data masking - a feature offered by Oracle Enterprise Manager – can be used to move data from production environment to a test environment. Both these are very crucial steps towards securing confidential customer data.
7. The database backups should be stored for the required period as per the regulations and bank's history retention policies. These backups should be securely stored and access should be controlled to authorized users only.

5.2.21 User Identification and Authentication

For implementation of authentication process please refer the document<SSO configuration doc>

For authorization, Please refer the document on access control.

5.2.22 Privacy controls

Tokenization mechanism is implemented in FCIS, where the token is created for every request that hit server for avoiding forgery attacks. Also, to avoid Clickjacking and frame spoofing attack FCIS have respective header and code configuration. Proper privacy control and content type has been placed.

5.2.23 Transmission Integrity and Confidentiality

Communication is over Transport Layer Security (TLS) and, Secure Sockets Layer (SSL), which are cryptographic protocols that provide communication security over the Internet. These transport protocols use asymmetric cryptography for authentication of key exchange, symmetric encryption for confidentiality and message authentication codes for message integrity. FLEXCUBE users are highly recommended to use SSL.

Furthermore, Http Only flag is included in a Set-Cookie HTTP response header. This tells the browser that this particular cookie should only be accessed by the server. Any attempt to access the cookie from client script is strictly forbidden.

5.2.24 Password Management

Certain user password related parameters should be defined at the bank level.

These parameters will apply to all the users of the system. Examples of such parameters are the number of invalid login attempts after which a user-id should be disabled, the maximum and minimum length for a password, the number of previous passwords that should not be used, the interval at which the password should be changed by every user, etc.

5.2.24.1 Invalid Logins

In FLEXCUBE user should specify the allowable number of times an invalid login attempt is made by a user. Each user accesses the system through a unique User ID and password. While logging on to the system, if either the User Id or the Password is wrong, it amounts to an invalid login attempt.
By default, the allowable number of cumulative invalid attempts is six, and the allowable number of consecutive invalid attempts is three. These default values can be changed and specify the allowable number of attempts in each case. An allowable number for cumulative attempts are between 6 and 99, and for consecutive (successive) attempts are between 3 and 5.

When authentication of credentials is unsuccessful due to an incorrect user ID, then the user id will not be logged in the audit logs. In case the user id is correct and the password is wrong,
the attempt is logged in the audit log and the successive and cumulative failure count is incremented. When the user id and password are correct, this is logged into the audit logs.

5.2.24.2 Specifying Parameter

Dormancy Days

Oracle FLEXCUBE allows you to automatically disable the profile of all the users who have not logged into the system for a pre-defined period of time. A user ID is considered dormant if the difference between the last login date and the current date is equal to or greater than the number of 'Dormancy Days' that has been specified. This is reckoned in calendar days i.e. inclusive of holidays. All dormant users (whose home module is same as the current module) are disabled during the end of day run at the current module.

5.2.24.3 Specifying Parameters for User Passwords

Password Length (characters)

The range of length (in terms of number of characters) of a user password can be set. The number of characters in a user password is not allowed to exceed the maximum length, or fall below the minimum length that has been specified.

The minimum length defaults to 8, and the maximum length to 15. The defaults values can be changed and specify the required range. The length can specify a minimum length between 6 and 15 characters, and a maximum length between 10 and 15 characters. The minimum length that specified must not exceed the maximum length that have specified.

Force Password Change after

The password of a user can be made valid for a fixed period after which a password change should be forced. After the specified number of days has elapsed for the user's password, it is no longer valid and a password change is forced. The number of calendar days defined will be applicable for a password change of any nature - either through the 'Change Password' function initiated by the user or a forced change initiated by the system. The system defaults to a value of 30, which can be changed. The number of days can be between 15 and 180 days,

Password Repetitions

The number of previous passwords that cannot be set as the new current password can be configured, when a password change occurs. The system defaults to a value of three (i.e., when a user changes the user password, the user's previous three passwords cannot be set as the new password). The default value can be changed and it can specify a number between one and five.

Minimum Days between Password Changes

The minimum number of calendar days that must elapse between two password changes can be configured. After a user has changed the user password, it cannot be changed again until the minimum numbers of days you specify here have elapsed.

Intimate Users (before password expiry)

The number of working days before password expiry can be configured, which is used to display a warning message to the user. When the user logs into the system (the stipulated number of days before the expiry date of the password), a warning message will continue to be displayed till the password expires or till the user changes it. By default, the value for this parameter is two (i.e., two days before password expiry).

5.2.24.4 Placing Restrictions on User Passwords

Application allows placing restrictions on the number of alpha and numeric characters that can be specified for a user password.

Maximum Consecutive Repetitive Characters

The maximum number of allowable repetitive characters occurring consecutively, in a user password can be specified. This specification is validated whenever a user changes the user password, and is applicable for a password change of any nature - either through the 'Change Password' function initiated by the user or a forced change initiated by the system.

Minimum Number of Special Characters in Password

Application allows defining minimum number of special characters allowed in a user password. The system validates these specifications only when a user chooses to change the password. Following is the default value application used:

Minimum No of Special Characters = 1

Minimum Number of Numeric Characters in Password

Likewise, application allows defining the minimum number of numeric characters allowed in a user password. The system validates the password only when a user chooses to change his password. Following is the default value used:

Minimum No of Numeric Characters = 1

Minimum Number of Lower Case Characters in Password

The minimum number of lowercase characters allowed in a user password also can be configured. The allowed lower case characters are from the US-ASCII character set only. The system validates these specifications only when a user chooses to change the password

Following is the default value used:

Minimum No of Lower Case Characters = 1

Minimum Number of Upper Case Characters in Password

The minimum number of upper case characters allowed in a user password can be configured. The allowed upper case characters are from the US-ASCII character set only. The system validates these specifications only when a user chooses to change the password.

Following is the default values used:

Minimum No of Upper Case Characters = 1

5.2.24.5 Password Restrictions

Application allows defining a list of passwords that cannot be used by any user of the system in the bank. This list, called the Restrictive Passwords list can be defined at three levels:

- At the bank level (applicable to all the users of the system)
- At the user role level (applicable for all the users assigned the same role)
- At the user level (applicable for the user)

The list of Restrictive Passwords should typically contain those passwords the users are most likely to use: the name of your bank, city, country, etc. For a user role, it could contain names, or terms, that are commonly used in the department. At the user level, it could contain the names of loved ones, etc. By disallowing users from using such common passwords, you can reduce the risk of somebody other than the user knowing the password.

6. General Information

6.1 Cryptography

FLEXCUBE uses cryptography to protect the sensitive data. It uses algorithm like SHA-2 and above family, TripleDES, AES.

6.2 Oracle Database Security suggestions:

- Install only what is required.
- Lock and expire default user account.
- Changing default user password.
- Change password for Administrative Accounts
- Change default password for all users
- Enforce password management
- Secure batch jobs
- Manage access to SYSDBA and SYSOPER roles
- Enable the Oracle data dictionary Protection
- Follow the principle of the least privilege

6.3 Security patch

Security patches needs to be applied whenever it's available for the applicable product version.

6.4 Oracle Software Security Assurance - Standards

Every acquired organization must complete the Mergers and Acquisitions (M&A) Security Integration process. The issues identified during this review must be addressed according to the agreed upon M&A remediation plan. The acquired organization must complete SPOC assignments and plan integration of OSSA methodologies and processes into its SDLC.

For more information visit https://gps.oracle.com/doku.php?do=search&id=OSSA

6.5 References

6.5.1 Datacenter Security considerations

Please refer to the following links to understand Datacenter Security considerations

http://docs.oracle.com/cd/B14099_19/core.1012/b13999/rectop.htm

http://www.sas70.us.com/industries/data-center-colocations.php

http://www.anixter.com/content/dam/Anixter/White%20Papers/12F0010X00-Four-Layers-Data-Center-Security-WP-EN-US.pdf

6.5.2 Database Security considerations

Please refer the below links to understand more on Database Security considerations recommended to be followed

http://www.oracle.com/us/products/database/security/overview/index.html

http://www.oracle.com/technetwork/products/secure-backup/overview/index.html

http://www.oracle.com/technetwork/database/security/twp-security-checklist-database-1-132870.pdf

http://www.red-database-security.com/wp/sentrigo_webinar.pdf

http://www.databasesecurity.com/oracle/twp_security_checklist_db_database.pdf

http://www.checklist20.com/pdfs/Databases/Oracle%20Database.pdf

http://www.applicure.com/blog/database-security-best-practice

6.5.3 Security recommendations / practices followed for Database Environment

Please refer the below mentioned links to understand more on Security recommendations / practices followed for Database Environment

http://docs.oracle.com/cd/B28359_01/network.111/b28531/guidelines.htm

http://ecomputernotes.com/database-system/adv-database/security-in-database-environment

https://security.berkeley.edu/node/138?destination=node/138

6.5.4 Common security considerations

Please refer below links to understand some of the common security considerations to be followed

http://docs.oracle.com/cd/B14099_19/core.1012/b28654.pdf

http://docs.oracle.com/cd/E14899_01/doc.9102/e14761/tuningforappserver.htm

http://docs.oracle.com/cd/E13222_01/wls/docs81b/lockdown/practices.html

http://docs.oracle.com/cd/E23943_01/web.1111/e14529/security.htm

http://www.oracle.com/us/solutions/oos/weblogic-server/overview/index.html

http://isu.ifmo.ru/docs/IAS904/core.904/b10377/arch.htm#1005544

http://www.ibm.com/developerworks/websphere/library/techarticles/0209_oberlin/oberlin.html

http://cnc.ucr.edu/security/desktop.html

http://makeitsafe.missouri.edu/best-practices/windows.html

https://security.tennessee.edu/pdfs/sdlbp.pdf

Security Practices Guide
[September] [2016]
Version 12.3.0.0.0

Oracle Financial Services Software Limited
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/