# Oracle® Fusion Middleware

## Upgrading Oracle Access Manager

12*c* (12.2.1.3.0)
F31416-06
November 2021

Oracle Fusion Middleware Upgrading Oracle Access Manager, 12*c* (12.2.1.3.0)

F31416-06

# Contents

# Part I In-Place Upgrade of Oracle Access Manager

## 3 Upgrading Oracle Access Manager Single Node Environments

## 4 Upgrading Oracle Access Manager Highly Available Environments

## B Updating the JDK After Installing and Configuring an Oracle Fusion Middleware Product

# Preface

This document describes how to upgrade an existing Oracle Access Manager environment to 12*c* (12.2.1.3.0).



## Audience

Identify the target audience for your book and learn more about this document intended for.

This document is intended for system administrators who are responsible for installing, maintaining, and upgrading Oracle Access Manager. It is assumed that readers have knowledge of the following:

- Oracle Fusion Middleware system administration and configuration.
- Configuration parameters and expected behavior of the system being upgraded.

## Documentation Accessibility

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

**Access to Oracle Support**

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

## Related Documents

Refer to the Oracle Fusion Middleware Library for additional information.

- For installation information, see Fusion Middleware Installation Documentation.
- For upgrade information, see Fusion Middleware Upgrade Documentation.
- For administration-related information, see Fusion Middleware Administration Documentation.

- For release-related information, see Fusion Middleware Release Notes.

# Conventions

Learn about the conventions used in this document.

This document uses the following text conventions:

| Convention | Meaning |
| --- | --- |
| **boldface** | Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary. |
| *italic* | Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values. |
| `monospace` | Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter. |

1

# Introduction to Upgrading Oracle Access Manager to 12*c* (12.2.1.3.0)

Before you begin, review all introductory information to understand the standard upgrade topologies and upgrade paths for Oracle Access Manager 12*c* (12.2.1.3.0).

> **Note:**
>
> For general information about Fusion Middleware upgrade planning and other upgrade concepts and resources, see the following sections in *Planning an Upgrade of Oracle Fusion Middleware*:
>
> - Planning an Upgrade to Oracle Fusion Middleware 12*c* (12.2.1.3.0)
> - Understanding In-Place versus Out-of-Place Upgrades
> - Understanding the Basic Upgrade Tasks

The following topics describe the concepts related to upgrading Oracle Access Manager:

- About the Starting Points for a Oracle Access Manager Upgrade
  You can upgrade to Oracle Access Manager 12*c* (12.2.1.3.0) from a supported 11*g* release.
- About the Oracle Access Manager Upgrade Scenarios
  The steps to upgrade Oracle Access Manager to 12*c* (12.2.1.3.0) depend on the existing 11*g* Release 2 (11.1.2.3.0) production topology.
- About the New Features for Oracle Access Manager 12c
  Several changes have been made to Oracle Access Manager between 11*g* and 12*c*.
- About Upgrade Restrictions
  If you are using two or more Oracle Fusion Middleware products of the same or different versions in a single, supported, Oracle Fusion Middleware configuration, you must consider the interoperability and compatibility factors before planning the upgrade.
- Terminology Used in this Guide
  For consistency, the following terminology is used in this guide.
- How to Use This Guide
  This guide covers various upgrade scenarios.

## About the Starting Points for a Oracle Access Manager Upgrade

You can upgrade to Oracle Access Manager 12*c* (12.2.1.3.0) from a supported 11*g* release.

Supported starting point for is upgrading Oracle Access Manager to 12*c* (12.2.1.3.0) is Oracle Access Manager 11*g* Release 2 (11.1.2.3.0).

If you are not using the 11.1.2.3.0 version of Oracle Access Manager, you must upgrade to 11.1.2.3.0 before you move to 12*c* (12.2.1.3.0).

For information about upgrading Oracle Access Manager to 11*g* Release 2 (11.1.2.3.0), see Introduction to Oracle Identity and Access Management Upgrade in the *Upgrade Guide for Oracle Identity and Access Management* for 11*g* Release 2 (11.1.2.3.0).

The upgrade procedures in this guide explain how to upgrade an existing Oracle Access Manager 11*g* domain to Oracle Access Manager 12*c* (12.2.1.3.0). If your domain contains other components, you will have to upgrade those components as well. Links to supporting documentation are provided wherever necessary.

For information about upgrade planning recommendations, see Doc ID 2539939.2.

# About the Oracle Access Manager Upgrade Scenarios

The steps to upgrade Oracle Access Manager to 12*c* (12.2.1.3.0) depend on the existing 11*g* Release 2 (11.1.2.3.0) production topology.

Oracle Access Manager can be deployed in a number of different ways. This upgrade documentation provides instructions for the common deployment topologies, it can however be used as a guide for the less common deployment topologies.

Your actual topology may vary, but the topologies described here provide an example that can be used as a guide to upgrade other similar Oracle Access Manager topologies.

> **Note:**
>
> For additional information about the upgrade process and planning resources to ensure your upgrade is successful, see Planning an Upgrade to Oracle Fusion Middleware 12c (12.2.1.3.0) in *Planning an Upgrade of Oracle Fusion Middleware*.

You can upgrade the following topologies or deployments using the procedure described in this guide:

- Single node environments
- Highly available (multinode) environments
- Oracle Access Manager Multi-data center setup
- Oracle Identity Manager and Oracle Access Manager integrated environments that are set up manually in 11.1.2.3.0
- Oracle Identity Manager and Oracle Access Manager integrated environments that are set up using Life Cycle Management (LCM) tool in 11.1.2.3.0

> **Note:**
>
> If you are using Oracle Access Manager Mobile and Social, read about the features not supported in this release of OAM before considering an upgrade to 12*c* (12.2.1.3.0). See Features Not Supported in Access Manager 12.2.1.3.0.

# About the New Features for Oracle Access Manager 12*c*

Several changes have been made to Oracle Access Manager between 11*g* and 12*c*.

To understand what's new in general in Oracle Fusion Middleware 12*c*, see New and Changed Features in *Understanding Oracle Fusion Middleware*.

If your environment includes Oracle WebLogic Server with Oracle ADF, see Key Differences Between Application Developer 11*g* and Infrastructure 12*c*.

For information about Oracle Access Manager 12*c* (12.2.1.3.0), and it's features, refer to the following topics in the *Administrator's Guide for Oracle Access Manager*:

- Features of Access Manager 12.2.1.3.0
- Features Not Supported in Access Manager 12.2.1.3.0
- Understanding Oracle Access Manager Services
- Understanding Oracle Access Management Access Manager

# About Upgrade Restrictions

If you are using two or more Oracle Fusion Middleware products of the same or different versions in a single, supported, Oracle Fusion Middleware configuration, you must consider the interoperability and compatibility factors before planning the upgrade.

**Interoperability**

In the context of Oracle Fusion Middleware products, Interoperability is defined as the ability of two Oracle Fusion Middleware products or components of the same version (or release) to work together (interoperate) in a supported Oracle Fusion Middleware configuration. Specifically, interoperability applies when the first 4 digits of the release or version number are the same. For example, Oracle Fusion Middleware 12*c* (12.2.1.0) components are generally interoperable with other 12*c* (12.2.1.0) components.

To facilitate a graceful upgrade of the components to 12c, during the upgrade, Oracle Access Manager 12c is compatible with OAM WebGate 11g (11.1.2.3), RREG11g Client 11g (11.1.2.3 with latest bundle patch), and ASDK 11g.

After you upgrade Oracle HTTP Server (OHS) from 12.2.1.2.0 (WebGate based on 11.1.2.3.0 OAM architecture) to the OHS 12.2.1.3.0 (WebGate based on 12.2.1.3.0 OAM architecture), you will need to add the User Defined Parameter of `UniqueCookieNames=Legacy` to the WebGate agent configuration through the OAM console. Adding the parameter ensures that the WebGate cookie name format is recognized by the OHS WebGate after the upgrade to 12c (12.2.1.3). For instructions, see Doc ID 2673236.1. For a list of the supported WebGate parameters, see User-Defined WebGate Parameters in *Administering Oracle Access Management*.

> **Note:**
>
> Exporting and importing OAM policies from other releases by using tools such as exportPolicy, importPolicy, and so on, is not certified. An upgrade is the only supported path to move policies from one release to another.

### Compatibility

In the context of Oracle Fusion Middleware products, Compatibility is defined as the ability of two Oracle Fusion Middleware components of different versions (or releases) to interoperate.

For a list of products and features available in Oracle Fusion Middleware Release 12.2.1.3.0, see Products and Features Available in Oracle Fusion Middleware 12*c* (12.2.1.3.0) in *Understanding Interoperability and Compatibility*.

# Terminology Used in this Guide

For consistency, the following terminology is used in this guide.

**Table 1-1 Terminology**

| Information | Example Value | Description |
|---|---|---|
| `JAVA_HOME` | `/home/Oracle/Java/jdk1.8.0_131` | Environment variable that points to the Java JDK home directory. |
| Database host | `examplehost.exampledomain` | Name and domain of the host where the database is running. |
| Database port | `1521` | Port number that the database listens on. The default Oracle database listen port is `1521`. |
| Database service name | `orcl.exampledomain` | Oracle databases require a unique service name. The default service name is `orcl`. |
| DBA username | `FMW` | Name of user with database administration privileges. The default DBA user on Oracle databases is `SYS`. |
| DBA password | *`<dba_password>`* | Password of the user with database administration privileges. |
| *`ORACLE_HOME`* | `/home/Oracle/`*`product`*`/ORACLE_HOME` | 12c directory in which you will install your software.<br>This directory will include Oracle Fusion Middleware Infrastructure and Oracle Access Manager, as needed. |
| Console port | `7001` | Port for Oracle WebLogic Server and Oracle Access Manager consoles. |

**Table 1-1 (Cont.) Terminology**

| Information | Example Value | Description |
|---|---|---|
| *DOMAIN_HOME* | /home/Oracle/config/domains/idm_domain | Location in which your domain data is stored.<br>**Note:** This is the domain where the primary Administration server is configured. |
| *APPLICATION_HOME* | /home/Oracle/config/applications/idm_domain | Location in which your application data is stored. |
| Administrator user name for your WebLogic domain | weblogic | Name of the user with Oracle WebLogic Server administration privileges. The default administrator user is weblogic. |
| Administrator user password | *<admin_password>* | Password of the user with Oracle WebLogic Server administration privileges. |
| RCU | *ORACLE_HOME*/oracle_common/bin | Path to the Repository Creation Utility (RCU). |
| RCU schema prefix | oam | Prefix for names of database schemas used by Oracle Access Manager. |
| RCU schema password | *<rcu_password>* | Password for the database schemas used by Oracle Access Manager. |
| Configuration utility | *ORACLE_HOME*/oracle_common/common/bin | Path to the Configuration Wizard for domain creation and configuration. |

# How to Use This Guide

This guide covers various upgrade scenarios.

Depending on your existing 11.1.2.3.0 deployment, refer to the respective topics for upgrading Oracle Access Manager to 12*c* (12.2.1.3.0):

- **Single Node Environments**:

  For upgrading single node Oracle Access Manager (OAM) setup, see Upgrading Oracle Access Manager Single Node Environments.

- **Multi-node or Highly Available Environments**:
  - For upgrading multi-node Oracle Access Manager setup, see Upgrading Oracle Access Manager Highly Available Environments.
  - For upgrading Oracle Access Manager multi-data center setup, see Upgrading Oracle Access Manager Multi-Data Center Environments.
- **OIM-OAM Integrated Highly Available Environments**:

- For upgrading OIM-OAM integrated highly available deployment, that was set up manually in 11*g*, see Upgrading OIM-OAM Integrated Environments set up Manually.
- For upgrading OIM-OAM integrated highly available deployment, that was set up using Life Cycle Management (LCM) tool in 11*g*, see Upgrading OIM-OAM Integrated Environments set up Using Life Cycle Management Tool.

> **Note:**
>
> Before you begin the upgrade, ensure that you review the Pre-Upgrade Requirements and perform necessary pre-upgrade tasks.

2

# Pre-Upgrade Requirements

Before you begin to upgrade Oracle Access Manager 12*c* (12.2.1.3.0), you must perform pre-upgrade tasks such as backing up, cloning your current environment, and verifying that your system meets certified requirements.

- Oracle Fusion Middleware Pre-Upgrade Checklist
  Perform the tasks in this checklist before you begin any upgrade to ensure you have a successful upgrade and limited downtime.
- Creating a Complete Backup
  Before you start an upgrade, back up all system-critical files, including the Oracle home, Middleware home, and databases that host your Oracle Fusion Middleware schemas.
- Verifying Certification and System Requirements
  Review the certification matrix and system requirements documents to verify that your environment meets the necessary requirements for installation.
- Updating Policy Files when Using Enhanced Encryption (AES 256)
  If you plan to use enhanced encryption, such as Advanced Encryption Standard (AES 256), in your upgraded environment, Oracle recommends that you apply the latest required policy files to the JDK before you upgrade.
- Creating a Non-SYSDBA User to Run the Upgrade Assistant
  Oracle recommends that you create a non-SYSDBA user called `FMW` to run the Upgrade Assistant. This user has the privileges required to modify schemas, but does not have full administrator privileges.
- Identifying Existing Schemas Available for Upgrade
  This optional task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.
- Validating the Keystore Files
- Enabling SSL Server Authentication Only Mode (SSL Mode 2) for Oracle Internet Directory
  If the Identity Store is Oracle Internet Directory (OID) SSL No Authentication Mode (SSL Mode 1), you should enable OID SSL Server Authentication Only Mode (SSL Mode 2) and import the OID certificate to OAM to prevent authentication failure after the upgrade.
- Shutting Down the Node Managers
  Ensure that you have shut down all the local and remote Node Managers before starting the upgrade process.

## Oracle Fusion Middleware Pre-Upgrade Checklist

Perform the tasks in this checklist before you begin any upgrade to ensure you have a successful upgrade and limited downtime.

Upgrades are performed while the servers are down. This checklist identifies important and often time-consuming pre-upgrade tasks that you can perform before the upgrade to limit

your downtime. The more preparation you do before you begin the upgrade process, the less time you will spend offline.

> **Note:**
>
> The pre-upgrade procedures you perform will depend on the configuration of your existing system, the components you are upgrading, and the environment you want to create at the end of the upgrade and configuration process. Complete only those tasks that apply to your configurations or use cases.
>
> Ensure that Oracle Access Manager and Oracle Identity Manager are in different domains. If they are in the same domain, then you need to separate them into multiple domains. For more information, see Separating Oracle Identity Management Applications Into Multiple Domains.

**Table 2-1 Tasks to Perform Before You Upgrade to Oracle Fusion Middleware 12*c***

| Task | Description |
|---|---|
| **Required**<br>Create a complete backup of your existing environment. | Back up all system-critical files, including the Oracle home, Middleware home, and databases that contain any schemas that are to be upgraded. If the upgrade fails, you must restore your pre-upgrade environment and begin the upgrade again.<br>See Creating a Complete Backup.<br>• Make sure that your backup includes the schema version registry table. See Backing Up the Schema Version Registry Table.<br>• If you modified any of the startup scripts in your existing domain, you will need to copy them to temporary directory location (outside of the existing domain) during the upgrade and redeploy them after the upgrade. See Maintaining Customized Domain and Environment Settings. |
| **Required**<br>Verify that you are installing and upgrading your product on a supported hardware and software configuration.<br>**Caution:** Do not attempt an upgrade if you are unable to use the latest supported operating system. As with all supported configurations, failure to comply with these requirements may cause your upgrade to fail. | Verify that your hardware and software configurations (including operating systems) are supported by the latest certifications and requirements. Also make sure to use a supported JDK version before you install the 12*c* product distributions.<br>Oracle recommends that you verify this information right before you start the upgrade as the certification requirements are frequently updated.<br>**Note:**<br>• Make sure that you have applied the latest patches to your components before you upgrade.<br>• Upgrade a component at a time, be it an Oracle Component or a dependent component. For example. Do not upgrade OUD, OIM, OAM, the operating system, the database, and the hardware all at the same time.<br>See Verifying Certification and System Requirements. |

**Table 2-1 (Cont.) Tasks to Perform Before You Upgrade to Oracle Fusion Middleware 12*c***

| Task | Description |
|---|---|
| **Required for 32–bit Operating Systems Only**<br>Migrate to a 64-bit operating system before you upgrade. | This is required only if you are currently running an unsupported 32–bit operating system.<br>See Migrating from a 32-Bit to a 64-Bit Operating System. |
| **Optional**<br>Update security policy files if you are using enhanced encryption (AES 256). | Some of the security algorithms used in Fusion Middleware 12*c* require additional policy files for the JDK.<br>If you plan to use enhanced encryption, such as AES 256, Oracle recommends that you apply the latest required policy files to the JDK before you upgrade.<br>See Updating Policy Files when Using Enhanced Encryption (AES 256). |
| **Optional**<br>Create a Non-SYSDBA user to run the Upgrade Assistant. | Oracle recommends that you create the FMW user to run Upgrade Assistant. User FMW can run the Upgrade Assistant without system administration privileges.<br>See Creating a Non-SYSDBA User to Run the Upgrade Assistant |
| **Optional**<br>Review the list of available schemas. | Query the schema version registry to view schema information.<br>See Identifying Existing Schemas Available for Upgrade. |
| **Required**<br>Ensure that the keystore files are valid. | See Validating the Keystore Files. |
| **Optional**<br>Enable OID SSL Server Authentication Only Mode (SSL Mode 2). | See Enabling SSL Server Authentication Only Mode (SSL Mode 2) for Oracle Internet Directory. |
| **Optional**<br>Shut down all the local and remote Node Managers before starting the upgrade process. | See Shutting Down the Node Managers. |

# Creating a Complete Backup

Before you start an upgrade, back up all system-critical files, including the Oracle home, Middleware home, and databases that host your Oracle Fusion Middleware schemas.

The backup must include the `SYSTEM.SCHEMA_VERSION_REGISTRY$` table so that you can restore the contents back to its pre-upgrade state if the upgrade fails.

> **Note:**
>
> The Upgrade Assistant Prerequisites screen prompts you to acknowledge that backups have been performed before you proceed with the actual upgrade. However, the Upgrade Assistant does not verify that a backup has been created.

See:

- Backing Up Your Environment in *Administering Oracle Fusion Middleware*

- Upgrading and Preparing Your Oracle Databases for 12*c* in *Planning an Upgrade of Oracle Fusion Middleware*
- Backup and Recovery in *Backup and Recovery User's Guide*.
- Oracle Database Documentation for information about upgrading to Oracle Database 18c and 19c.
- Backing Up the Schema Version Registry Table
  Your system backup must include the `SYSTEM.SCHEMA_VERSION_REGISTRY$` table or the `FMWREGISTRY.SCHEMA_VERSION_REGISTRY$` table.
- Maintaining Customized Domain and Environment Settings
  If you have modified any domain-generated, server startup scripts, or configuration files in your pre-upgrade environment, it is important to note that these changes are overwritten during the installation, domain upgrade, and reconfiguration operations. Save your customized files to a shared library location so that you can continue to use them after the upgrade.

## Backing Up the Schema Version Registry Table

Your system backup must include the `SYSTEM.SCHEMA_VERSION_REGISTRY$` table or the `FMWREGISTRY.SCHEMA_VERSION_REGISTRY$` table.

Each Fusion Middleware schema has a row in the `SYSTEM.SCHEMA_VERSION_REGISTRY$` table. If you run the Upgrade Assistant to update an existing schema and it does not succeed, you must restore the original schema before you can try again. Before you run the Upgrade Assistant, make sure you back up your existing database schemas and the schema version registry.

> **Note:**
>
> Before you upgrade a schema using the Upgrade Assistant, you must perform a complete database backup. During the upgrade, you are required to acknowledge that backups have been performed.

## Maintaining Customized Domain and Environment Settings

If you have modified any domain-generated, server startup scripts, or configuration files in your pre-upgrade environment, it is important to note that these changes are overwritten during the installation, domain upgrade, and reconfiguration operations. Save your customized files to a shared library location so that you can continue to use them after the upgrade.

Every domain installation includes dynamically-generated domain and server startup scripts, such as `setDomainEnv`. These files are replaced by newer versions during the installation and upgrade process. To maintain your custom domain-level environment settings, Oracle recommends that you create a separate file to store the custom domain information before you upgrade, instead of modifying the scripts directly.

For example, if you want to customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverrides.cmd` (Windows) or `setUserOverrides.sh` (UNIX) and configure it to add custom libraries to the WebLogic Server classpath, specify additional command-line options for running the servers, or

specify additional environment variables. When using the `pack` and `unpack` commands, any custom settings that you add to this file are preserved during the domain upgrade operation and are carried over to the remote servers.

The following example illustrates startup customizations in a `setUserOverrides` file:

```
# add custom libraries to the WebLogic Server system claspath
  if [ "${POST_CLASSPATH}" != "" ] ; then
    POST_CLASSPATH="${POST_CLASSPATH}${CLASSPATHSEP}${HOME}/foo/fooBar.jar"
    export POST_CLASSPATH
  else
    POST_CLASSPATH="${HOME}/foo/fooBar.jar"
    export POST_CLASSPATH
  fi

# specify additional java command-line options for servers
JAVA_OPTIONS="${JAVA_OPTIONS}  -Dcustom.property.key=custom.value"
```

If the `setUserOverrides` file exists during a server startup, the file is included in the startup sequence and any overrides contained within this file take effect. You must store the `setUserOverrides` file in the *`DOMAIN_HOME`*`/bin` directory.

> **Note:**
>
> If you are unable to create the `setUserOverrides` script before an upgrade, you need to reapply your settings as described in Re-apply Customizations to Startup Scripts in *Upgrading Oracle WebLogic Server*.

# Verifying Certification and System Requirements

Review the certification matrix and system requirements documents to verify that your environment meets the necessary requirements for installation.

> **Note:**
>
> When checking the certification, system requirements, and interoperability information, be sure to check specifically for any 32-bit or 64-bit system requirements. It is important for you to download software specifically designed for the 32-bit or 64-bit environment, explicitly.

- Verify Your Environment Meets Certification Requirements
  Oracle has tested and verified the performance of your product on all certified systems and environments. Make sure that you are installing your product on a supported hardware and software configuration.
- Verify System Requirements and Specifications
  It is important to verify that the system requirements such as disk space, available memory, specific platform packages and patches, and other operating system-specific items are met.

- Verify That the Database Hosting Oracle Fusion Middleware is Supported
  You must have a supported Oracle database configured with the required schemas before you run Oracle Fusion Middleware 12*c*.
- Verify That the JDK Is Certified for This Release of Oracle Fusion Middleware
  At the time this document was published, the certified JDK for 12*c* (12.2.1.3.0) was 1.8.0_131.

## Verify Your Environment Meets Certification Requirements

Oracle has tested and verified the performance of your product on all certified systems and environments. Make sure that you are installing your product on a supported hardware and software configuration.

Whenever new certifications occur, they are added to the appropriate certification document right away. New certifications can occur at any time, and for this reason the certification documents are kept outside of the documentation libraries and are available on Oracle Technical Resources. See the Certification Matrix for 12*c* (12.2.1.3.0).

## Verify System Requirements and Specifications

It is important to verify that the system requirements such as disk space, available memory, specific platform packages and patches, and other operating system-specific items are met.

Use the *Oracle Fusion Middleware System Requirements and Specifications* document to verify that the requirements of the certification are met. For example, if the Certification Matrix for 12*c* (12.2.1.3.0) indicates that your product is certified for installation on 64-Bit Oracle Linux 7, verify that your Oracle Linux 7 system has met the required minimum specifications such as disk space, available memory, specific platform packages and patches, and other operating system-specific items. This document is updated as needed and resides outside of the documentation libraries on the Oracle Technical Resources.

> **Note:**
>
> When you install the Oracle Fusion Middleware Release 12*c* software in preparation for upgrade, you should use the same user account that you used to install and configure the existing, pre-upgrade Oracle Fusion Middleware software. On UNIX operating systems, this ensures that the proper owner and group is applied to new Oracle Fusion Middleware 12*c* files and directories.

If you are running a 32–bit environment, you will need to perform an additional set of steps:

- Migrating from a 32-Bit to a 64-Bit Operating System
  If you have a 32–bit operating system, then you must migrate your 32-bit environment to a 64-bit software environment before you upgrade.

# Migrating from a 32-Bit to a 64-Bit Operating System

If you have a 32–bit operating system, then you must migrate your 32-bit environment to a 64-bit software environment before you upgrade.

Make sure to validate the migration to ensure all your Oracle Fusion Middleware 11*g* software is working properly on the 64-bit machine, and only then perform the upgrade to Oracle Fusion Middleware 12*c*.

In these tasks, *host* refers to the 32-bit source machine and *target* refers to the new 64-bit target machine.

> **Note:**
>
> These steps assume that your database is located on a separate host and will not be moved.

Upgrading an operating system typically involves the following:

> **Caution:**
>
> These steps are provided as an example of the operating system upgrade process and may or may not include all of the procedures you must perform to update your specific operating system. Consult your operating system's upgrade documentation for more information.

- Procure the Hardware That Supports the Upgrade's 64-bit Software Requirement
  Make sure that you have supported target hardware in place before you begin the upgrade process.
- Stop All Processes
  Before upgrading, you must stop all processes, including Managed Servers, the Administration Server, and Node Manager, if they are started on the host.
- Back Up All Files from the 32-bit Host Machine
  Make sure that you have created a complete backup of your entire 11*g* deployment before you begin the upgrade process. These files can be used if there is an issue during the migration and you have to restart the process.
- Set Up the Target 64-bit Machine with the 11g Host Name and IP Address
  The host name and IP address of the target machine must be made identical to the host. This requires you to change the IP address and name of the source machine or decommission the source machine to avoid conflicts in the network.
- Restore the 11g Backup from 32-bit Host to 64-bit Host
  Restore the files you backed from the 32-bit host using the same directory structure that was used in 11*g*. The directory structure on the target machine must be identical to the structure of the host machine.
- Install the 12c Product Distributions on the Target Machine
  Oracle recommends an Out-of-Place approach for upgrade. Therefore, you must install the 12*c* product distributions in a new Oracle home on the target machine.

- Upgrade the Target 64-bit Environment Using the Standard Upgrade Procedure
  After installing the product on the target machine, you must upgrade each product component individually using an Upgrade Utility specified in the component-specific upgrade guide and complete any post-upgrade tasks.

## Procure the Hardware That Supports the Upgrade's 64-bit Software Requirement

Make sure that you have supported target hardware in place before you begin the upgrade process.

## Stop All Processes

Before upgrading, you must stop all processes, including Managed Servers, the Administration Server, and Node Manager, if they are started on the host.

> **Note:**
>
> Ensure that the Database is up and running, during the upgrade.

### Step 1: Stop the Managed Servers

Depending on the method you followed to start the managed servers, follow one of the following methods to stop the WebLogic Managed Server:

**Method 1**: To stop a WebLogic Server Managed Server by using the Weblogic Console:

- Log into Weblogic console as a `weblogic` Admin.
- Go to **Servers** > **Control** tab.
- Select the required managed server.
- Click **Shutdown**.

**Method 2**: To stop a WebLogic Server Managed Server using node manager, run the following commands:

```
wls:/offline>nmConnect('nodemanager_username','nodemanager_password',
            'AdminServerHostName','5556','domain_name',
            'DOMAIN_HOME')

wls:/offline>nmKill('ManagedServerName')
```

### Step 2: Stop the Administration Server

When you stop the Administration Server, you also stop the processes running in the Administration Server, including the WebLogic Server Administration Console and Fusion Middleware Control.

To stop the Administration Server, use the `stopWebLogic` script:

- (UNIX) *DOMAIN_HOME*`/bin/stopWebLogic.sh`
- (Windows) *DOMAIN_HOME*`\bin\stopWebLogic.cmd`

When prompted, enter your user name, password, and the URL of the Administration Server.

**Step 3: Stop Node Manager**

To stop Node Manager, close the command shell in which it is running.

Alternatively, after having set the `nodemanager.properties` attribute `QuitEnabled` to `true` (the default is `false`), you can use WLST to connect to Node Manager and shut it down. See stopNodeManager in *WLST Command Reference for WebLogic Server*.

## Back Up All Files from the 32-bit Host Machine

Make sure that you have created a complete backup of your entire 11*g* deployment before you begin the upgrade process. These files can be used if there is an issue during the migration and you have to restart the process.

> **Note:**
>
> If the upgrade from 32-bit to 64-bit takes place on the same machine, there is a risk of corrupting the source environment if the upgrade fails.

See Backing Up Your Environment in *Oracle Fusion Middleware Administrator's Guide*.

During the upgrade you must have access to the contents of the following:

- *11g_DOMAIN_HOME*
- *11g*`/nodemanager` directory located in *11g_ORACLE_HOME*`/wlserver/common/`

Some of the backup and recovery procedures described in Backing Up Your Environment in *Oracle Fusion Middleware Administrator's Guide* are product-specific. Do not proceed with the upgrade until you have a complete backup.

## Set Up the Target 64-bit Machine with the 11*g* Host Name and IP Address

The host name and IP address of the target machine must be made identical to the host. This requires you to change the IP address and name of the source machine or decommission the source machine to avoid conflicts in the network.

The process of changing an IP address and host name vary by operating system. Consult your operating system's administration documentation for more information.

## Restore the 11*g* Backup from 32-bit Host to 64-bit Host

Restore the files you backed from the 32-bit host using the same directory structure that was used in 11*g*. The directory structure on the target machine must be identical to the structure of the host machine.

See Recovering Your Environment in *Oracle Fusion Middleware Administrator's Guide*.

## Install the 12*c* Product Distributions on the Target Machine

Oracle recommends an Out-of-Place approach for upgrade. Therefore, you must install the 12*c* product distributions in a new Oracle home on the target machine.

Refer to the component-specific installation guides for the component(s) you are installing.

### Upgrade the Target 64-bit Environment Using the Standard Upgrade Procedure

After installing the product on the target machine, you must upgrade each product component individually using an Upgrade Utility specified in the component-specific upgrade guide and complete any post-upgrade tasks.

If you are upgrading additional components, see the component-specific upgrade guide.

> **Note:**
>
> The Node Manager upgrade procedure requires access to the original Node Manager files. Use the 11*g* Node Manger files that you backed up from the 32-bit source machine as part of Back Up All Files from the 32-bit Host Machine.

## Verify That the Database Hosting Oracle Fusion Middleware is Supported

You must have a supported Oracle database configured with the required schemas before you run Oracle Fusion Middleware 12*c*.

Review the Fusion Middleware database requirements before starting the upgrade to ensure that the database hosting Oracle Fusion Middleware is supported and has sufficient space to perform an upgrade. See the Certification Matrix for 12*c* (12.2.1.3.0).

> **Note:**
>
> If your database version is no longer supported, you must upgrade to a supported version before starting an upgrade. See Upgrading and Preparing Your Oracle Databases for 12c in *Planning an Upgrade of Oracle Fusion Middleware*.

## Verify That the JDK Is Certified for This Release of Oracle Fusion Middleware

At the time this document was published, the certified JDK for 12*c* (12.2.1.3.0) was 1.8.0_131.

Refer to the Oracle Fusion Middleware Supported System Configurations information on the Oracle Technical Resources to verify that the JDK you are using is supported.

If your JDK is not supported, or you do not have a JDK installed, you must download the required Java SE JDK, from the following website:

`http://www.oracle.com/technetwork/java/javase/downloads/index.html`

Make sure that the JDK is installed outside of the Oracle home. The Oracle Universal Installer validates that the designated Oracle home directory is empty, and the install does not progress until an empty directory is specified. If you install JDK under Oracle home, you may experience issues in future operations. Therefore, Oracle recommends that you use install the JDK in the following directory: `/home/oracle/products/jdk`.

# Updating Policy Files when Using Enhanced Encryption (AES 256)

If you plan to use enhanced encryption, such as Advanced Encryption Standard (AES 256), in your upgraded environment, Oracle recommends that you apply the latest required policy files to the JDK before you upgrade.

The Java platform defines a set of APIs spanning major security areas, including cryptography, public key infrastructure, authentication, secure communication, and access control. These APIs allow developers to easily integrate security mechanisms into their application code.

Some of the security algorithms used in Fusion Middleware 12*c* (12.2.1.3.0) require additional policy files for the JDK. See Java Cryptography Architecture Oracle Providers Documentation.

> **Note:**
>
> If you attempt to use enhanced encryption without applying these policy files to the JDK before you begin the upgrade, the upgrade can fail and you must restore the entire pre-upgrade environment and start the upgrade from the beginning.

# Creating a Non-SYSDBA User to Run the Upgrade Assistant

Oracle recommends that you create a non-SYSDBA user called `FMW` to run the Upgrade Assistant. This user has the privileges required to modify schemas, but does not have full administrator privileges.

SYSDBA is an administrative privilege that is required to perform high-level administrative operations such as creating, starting up, shutting down, backing up, or recovering the database. The SYSDBA system privilege is for a fully empowered database administrator. When you connect with the SYSDBA privilege, you connect with a default schema and not with the schema that is generally associated with your user name. For SYSDBA, this schema is SYS. Access to a default schema can be a very powerful privilege. For example, when you connect as user SYS, you have unlimited privileges on data dictionary tables. Therefore, Oracle recommends that you create a non-SYSDBA user to upgrade the schemas. The privileges listed below must be granted to user FMW before starting the Upgrade Assistant.

**Notes:**

The non-SYSDBA user FMW is created solely for the purpose of running the Upgrade Assistant. After this step is complete, drop the FMW user. Note that privileges required for running the Upgrade Assistant may change from release to release.
By default, the `v$xatrans$` table does not exist. You must run the `XAVIEW.SQL` script to create this table before creating the user. Moreover, the `grant select` privilege on the `v$xatrans$` table is required only by Oracle Identity Governance . If you do not require Oracle Identity Governance for configuration, or if you do not have the `v$xatrans$` table, then remove the following line from the script:

```
grant select on v$xatrans$ to FMW with grant option;
```

In the example below, *<password>* is the password that you set for the FMW user. When granting privileges, make sure that you specify your actual password.

```
create user FMW identified by <password>;
grant dba to FMW;
grant execute on DBMS_LOB to FMW with grant option;
grant execute on DBMS_OUTPUT to FMW with grant option;
grant execute on DBMS_STATS to FMW with grant option;
grant execute on sys.dbms_aqadm to FMW with grant option;
grant execute on sys.dbms_aqin to FMW with grant option;
grant execute on sys.dbms_aqjms to FMW with grant option;
grant execute on sys.dbms_aq to FMW with grant option;
grant execute on utl_file to FMW with grant option;
grant execute on dbms_lock to FMW with grant option;
grant select on sys.V_$INSTANCE to FMW with grant option;
grant select on sys.GV_$INSTANCE to FMW with grant option;
grant select on sys.V_$SESSION to FMW with grant option;
grant select on sys.GV_$SESSION to FMW with grant option;
grant select on dba_scheduler_jobs to FMW with grant option;
grant select on dba_scheduler_job_run_details to FMW with grant option;
grant select on dba_scheduler_running_jobs to FMW with grant option;
grant select on dba_aq_agents to FMW with grant option;
grant execute on sys.DBMS_SHARED_POOL to FMW with grant option;
grant select on dba_2pc_pending to FMW with grant option;
grant select on dba_pending_transactions to FMW with grant option;
grant execute on DBMS_FLASHBACK to FMW with grant option;
grant execute on dbms_crypto to FMW with grant option;
grant execute on DBMS_REPUTIL to FMW with grant option;
grant execute on dbms_job to FMW with grant option;
grant select on pending_trans$ to FMW with grant option;
grant select on dba_scheduler_job_classes to fmw with grant option;
grant select on SYS.DBA_DATA_FILES to FMW with grant option;
grant select on SYS.V_$ASM_DISKGROUP to FMW with grant option;
grant select on v$xatrans$ to FMW with grant option;
grant execute on sys.dbms_system to FMW with grant option;
grant execute on DBMS_SCHEDULER to FMW with grant option;
```

```
grant select on dba_data_files to FMW with grant option;
grant execute on UTL_RAW to FMW with grant option;
grant execute on DBMS_XMLDOM to FMW with grant option;
grant execute on DBMS_APPLICATION_INFO to FMW with grant option;
grant execute on DBMS_UTILITY to FMW with grant option;
grant execute on DBMS_SESSION to FMW with grant option;
grant execute on DBMS_METADATA to FMW with grant option;
grant execute on DBMS_XMLGEN to FMW with grant option;
grant execute on DBMS_DATAPUMP to FMW with grant option;
grant execute on DBMS_MVIEW to FMW with grant option;
grant select on ALL_ENCRYPTED_COLUMNS to FMW with grant option;
grant select on dba_queue_subscribers to FMW with grant option;
grant execute on SYS.DBMS_ASSERT to FMW with grant option;
grant select on dba_subscr_registrations to FMW with grant option;
grant manage scheduler to FMW;
```

# Identifying Existing Schemas Available for Upgrade

This optional task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.

You can let the Upgrade Assistant upgrade all of the schemas in the domain, or you can select individual schemas to upgrade. To help decide, follow these steps to view a list of all the schemas that are available for an upgrade:

1. If you are using an Oracle database, connect to the database by using an acount that has Oracle DBA privileges, and run the following from SQL*Plus:

   ```
   SET LINE 120
   COLUMN MRC_NAME FORMAT A14
   COLUMN COMP_ID FORMAT A20
   COLUMN VERSION FORMAT A12
   COLUMN STATUS FORMAT A9
   COLUMN UPGRADED FORMAT A8
   SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM
   SCHEMA_VERSION_REGISTRY ORDER BY MRC_NAME, COMP_ID;
   ```

2. Examine the report that is generated.

   If an upgrade is not needed for a schema, the `schema_version_registry` table retains the schema at its pre-upgrade version.

3. Note the schema prefix name that was used for your existing schemas. You will use the same prefix when you create new 12*c* schemas.

**Notes:**

- If you used an OID-based policy store in 11*g*, make sure to create a new OPSS schema before you perform the upgrade. After the upgrade, the OPSS schema remains an LDAP-based store.
- You can only upgrade schemas for products that are available for upgrade in Oracle Fusion Middleware release 12*c* (12.2.1.3.0). Do not attempt to upgrade a domain that includes components that are not yet available for upgrade to 12*c* (12.2.1.3.0).

# Validating the Keystore Files

Before performing the upgrade, ensure that the following keystore files are valid and are not in a corrupted state:

- `oamkeystore.jks`
  Use this command to validate the file:

  ```
  keytool -list -keystore $DOMAIN_HOME/config/fmwconfig/.oamkeystore -
  storepass
  xxx -storetype jceks
  ```

- `Default-keystore.jks`
  Use this command to validate the file:

  ```
  keytool -list -keystore $DOMAIN_HOME/config/fmwconfig/default-
  keystore.jks
  -storepass xxx -storetype jceks
  ```

**Note:**

Do not start the upgrade if there is an issue in any of these keystore files.

# Enabling SSL Server Authentication Only Mode (SSL Mode 2) for Oracle Internet Directory

If the Identity Store is Oracle Internet Directory (OID) SSL No Authentication Mode (SSL Mode 1), you should enable OID SSL Server Authentication Only Mode (SSL Mode 2) and import the OID certificate to OAM to prevent authentication failure after the upgrade.

If SSL Mode 2 is not enabled, authentication fails with the following error:

```
Simple Bind Failed
```

To enable OID SSL Mode 2, see Configuring Oracle Directory Integration Platform for Oracle Internet Directory SSL Authentication and Doc ID 1203927.1.

Alternatively, you can implement the workaround as described in Doc ID 2512386.1 after the upgrade.

# Shutting Down the Node Managers

Ensure that you have shut down all the local and remote Node Managers before starting the upgrade process.

The Node Managers should remain shut down until you start the WebLogic Administration Server after completing the upgrade. When the WebLogic Administration Server is up and running, start the Node Managers, followed by the Managed Servers.

# Part I

# In-Place Upgrade of Oracle Access Manager

You can perform an in-place upgrade of Oracle Access Manager single node deployments, highly available environments, and Oracle Access Manager in a Multi-Data Center setup by using the procedures described in this part.

This part contains the following topics:

- Upgrading Oracle Access Manager Single Node Environments
  You can upgrade Oracle Access Manager from Release `11g Release 2 (11.1.2.3.0)` to Oracle Access Manager 12*c* (12.2.1.3.0) .
- Upgrading Oracle Access Manager Highly Available Environments
  Describes the process of upgrading an Oracle Access Manager highly available environments from 11*g* Release 2 (11.1.2.3.0) to 12*c* (12.2.1.3.0).
- Upgrading Oracle Access Manager Multi-Data Center Environments
  You can upgrade Oracle Access Manager deployed across multi-data centers (MDC) from 11g Release 2 (11.1.2.3.0) to 12*c* (12.2.1.3.0).
- Upgrading OIM-OAM Integrated Environments set up Manually
  You can upgrade Oracle Identity Manager (OIM), Oracle Access Manager (OAM) integrated split domain highly available environments that are set up manually, from 11*g* Release 2 (11.1.2.3.0) to 12*c* (12.2.1.3.0) using the upgrade procedure described in this section.
- Upgrading OIM-OAM Integrated Environments set up Using Life Cycle Management Tool
  If you had set up an Oracle Identity Manager – Oracle Access Manager integrated environment in 11*g* Release 2 (11.1.2.3.0) using the Life Cycle Management (LCM) tool, follow the instructions in this chapter to upgrade the same to 12*c* (12.2.1.3.0).

3

# Upgrading Oracle Access Manager Single Node Environments

You can upgrade Oracle Access Manager from Release 11*g* Release 2 (11.1.2.3.0) to Oracle Access Manager 12*c* (12.2.1.3.0) .

Complete the steps in the following topics to perform the upgrade:

- About the Oracle Access Manager Single Node Upgrade Process
  Review the roadmap for an overview of the upgrade process for Oracle Access Manager single node deployments.
- Completing the Pre-Upgrade Tasks for Oracle Access Manager
  Complete the pre-upgrade tasks described in this section before you upgrade Oracle Access Manager.
- Installing Product Distributions
  Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure and Oracle Access Manager 12*c* (12.2.1.3.0) distributions on the target system and install them using Oracle Universal Installer.
- Installing the Latest Stack Patch Bundle
  After you install the product distributions, Oracle strongly recommends you to apply the latest IDM Stack Patch Bundle (SPB) 12.2.1.3.0 before proceeding with the upgrade process. You can apply the patch by using the Opatch tool. Applying the SPB helps eliminate most of the upgrade issues or workarounds.
- Creating the Required 12c Schemas Using RCU
  When upgrading from 11g, you must create extra schemas required for 12c. If your setup has non-SSL ports open, you can use the Upgrade Assistant to create schemas by using the default schema settings. In case of SSL enabled setup, you can use the Repository Creation Utility (RCU) to create customized schemas. This procedure describes how to create schemas using the RCU. Information about using the Upgrade Assistant to create schemas is covered in the upgrade procedures.
- Integrating Access Federation with BI Publisher
  Update to the required or latest patches to seamlessly integrate and view Oracle Access Manager audit information on BI Publisher.
- Running a Pre-Upgrade Readiness Check
  To identify potential issues with the upgrade, Oracle recommends that you run a readiness check before you start the upgrade process. Be aware that the readiness check may not be able to discover all potential issues with your upgrade. An upgrade may still fail, even if the readiness check reports success.
- Stopping Servers and Processes
  Before you run the Upgrade Assistant to upgrade your schemas and configurations, you must shut down all of the pre-upgrade processes and servers, including the Administration Server, Node manager, and any managed servers.

- Upgrading Product Schemas
  After stopping servers and processes, use the Upgrade Assistant to upgrade supported product schemas to the current release of Oracle Fusion Middleware.
- About Reconfiguring the Domain
  Run the Reconfiguration Wizard to reconfigure your domain component configurations to 12*c* (12.2.1.3.0).
- Upgrading Domain Component Configurations
  After reconfiguring the domain, use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.
- Performing Post-Upgrade Tasks
  After performing the upgrade of Oracle Access Manager to 12c (12.2.1.3), you should complete the tasks summarized in this section, if required.
- Performing the Post-Patch Install Steps
  After completing the upgrade, you have to perform the post-patch installation steps.

# About the Oracle Access Manager Single Node Upgrade Process

Review the roadmap for an overview of the upgrade process for Oracle Access Manager single node deployments.

The steps required to upgrade an existing domain will vary depending on how the domain is configured and which components are being upgraded.

**Table 3-1 Tasks for Upgrading Single Node Oracle Access Manager Deployments**

| Task | Description |
|---|---|
| **Optional**<br>If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks. | See:<br>• Introduction to Upgrading Oracle Access Manager to 12*c* (12.2.1.3.0)<br>• Pre-Upgrade Requirements |
| **Required**<br>Complete the necessary pre-upgrade tasks specific to Oracle Access Manager. | See Completing the Pre-Upgrade Tasks for Oracle Access Manager. |
| **Required**<br>Install Fusion Middleware Infrastructure and Oracle Access Manager 12*c* (12.2.1.3.0) in a new Oracle home. | Install Fusion Middleware Infrastructure and Oracle Access Manager in a *new* Oracle home on the same host as the 11*g* production deployment before you begin the upgrade. In 12*c*, Oracle home is used to describe the 11*g* Middleware home.<br>See Installing Product Distributions. |
| **Required**<br>Apply the latest bundle patches. | See Installing the Latest Stack Patch Bundle. |
| **Required**<br>Start the Repository Creation Utility (RCU) to create the required 12*c* database schemas. | The schemas you create will vary depending on your existing schema configuration.<br>See Creating the Required 12c Schemas with the RCU. |
| **Required**<br>Run a pre-upgrade readiness check. | See Running a Pre-Upgrade Readiness Check. |

**Table 3-1 (Cont.) Tasks for Upgrading Single Node Oracle Access Manager Deployments**

| Task | Description |
|---|---|
| **Required**<br>Shut down the 11*g* environment (stop all Administration and Managed Servers).<br>Ensure that the Database is up during the upgrade. | **WARNING**: Failure to shut down your servers during an upgrade may lead to data corruption.<br>See Stopping Servers and Processes. |
| **Required**<br>Start the Upgrade Assistant to upgrade the 11*g* database schemas and to migrate all active (in flight) instance data. | See Upgrading Product Schemas.<br>**NOTE:** The upgrade of active instance data is started automatically when running the Upgrade Assistant. Once the data is successfully upgraded to the new 12*c* (12.2.1.3.0) environment, you can close the Upgrade Assistant. The closed instances will continue to upgrade through a background process. |
| **Required**<br>Start the Reconfiguration Wizard to reconfigure the domain. | During an upgrade, the Configuration Wizard is run in reconfiguration mode to update the existing domain to use the newly installed software.<br>See Reconfiguring the Domain Using the Reconfiguration Wizard. |
| **Required**<br>Start the Upgrade Assistant (again) to upgrade Oracle Access Manager domain component configurations. | The Upgrade Assistant is used to update the reconfigured domain's component configurations.<br>See Upgrading Domain Component Configurations. |
| **Required**<br>Complete any necessary post-upgrade tasks. | These tasks are optional. See Performing Post-Upgrade Tasks. |
| **Required**<br>Perform the post-patch install steps. | See Performing the Post-Patch Install Steps. |

# Completing the Pre-Upgrade Tasks for Oracle Access Manager

Complete the pre-upgrade tasks described in this section before you upgrade Oracle Access Manager.

- Checking the Supported Starting Point for Oracle Access Manager Upgrade
  The Oracle Access Manager version that is supported for upgrade is `11g Release 2 (11.1.2.3.0)`.
- Checking if OAM is in a Different Domain to OAAM and OIM
  In the case of Oracle Access Manager (OAM), Oracle Adaptive Access Management (OAAM), and Oracle Identity Manager (OIM) integrated setup, where OAM and OAAM are in same domain, and OIM is in a separate domain, the OAM domain needs to be cloned that works with OAAM and OIM in the source domain.
- Removing the IAMSuiteAgent Deployment
  The `IAMSuiteAgent` deployment is not supported in 12*c*. Therefore, undeploy the `IAMSuiteAgent` before you proceed with the upgrade.
- Upgrading Java JSE Policy
  Upgrade Java JSE Policy, if required.

# Checking the Supported Starting Point for Oracle Access Manager Upgrade

The Oracle Access Manager version that is supported for upgrade is `11g Release 2 (11.1.2.3.0)`.

If you are using an earlier version of Oracle Access Manager, you must upgrade to Oracle Access Manager `11g Release 2 (11.1.2.3.0)` first, and then to 12*c*.

# Checking if OAM is in a Different Domain to OAAM and OIM

In the case of Oracle Access Manager (OAM), Oracle Adaptive Access Management (OAAM), and Oracle Identity Manager (OIM) integrated setup, where OAM and OAAM are in same domain, and OIM is in a separate domain, the OAM domain needs to be cloned that works with OAAM and OIM in the source domain.

> **Note:**
>
> Ensure that Oracle Access Manager and Oracle Identity Manager are in different domains. If they are in the same domain, then you need to separate them into multiple domains. For more information, see Separating Oracle Identity Management Applications Into Multiple Domains.

To separate the OAM and OAAM domain, do the following:

1. Perform the test-to-production of the source environment (machine-1) where OAM and OAAM is in the same domain, so as to form the 11.1.2.3.0 OAM-OAAM environment on machine-2. This machine-2 acts as the production machine.
2. On machine-1, open the *DOMAIN_HOME*`/config/fmwconfig/oam-config.xml` file in a text editor, and search for the parameter `HOST_ALIAS_1`.
3. Update the `serverhost` parameter to reflect the name of production machine, so that it knows the target (OAAM) machine to which it has to point to render the OAAM authentication page.
4. Search for the parameter `Version`, and increment its value by one.
5. Restart only the Administration Server and the OAM Server of source machine (machine-1) to reflect the changes.

   Ensure that the oaam_admin_server1 and oaam_server_server1 on the source machine are stopped.
6. Start the oaam_admin_server1 and oaam_server_server1 on production machine (machine-2). The Administration Server on the production machine will be in `Running` state after the T2P.
7. Access the tapscheme protected resource of machine-1. Make sure that the request gets redirected to OAAM server of machine—2 and subsequent taspscheme login is successful.

> **Note:**
>
> Ensure that the date and time on source and production machine are in sync. If they are not, the authentication fails.

If OIM is installed in a separate domain, and is integrated with OAM and OAAM, do the following:

1. Update the following Oracle Identity Manager properties to contain the details of the new OAAM host:

   ```
   OIM.ChangePasswordURL
   OIM.ChallengeQuestionModificationURL
   ```

   For information about setting the Oracle Identity Manager properties for OAAM, see Setting Oracle Identity Manager Properties for Oracle Adaptive Access Manager in the *Integration Guide for Oracle Identity Management Suite* for 11*g* Release 2 (11.1.2.3.0).
2. Restart the Oracle Identity Manager server.

> **Note:**
>
> You must upgrade the OAM domain whose Managed Server is in the running state after the domain separation.
>
> For example, if you have followed the steps in this section, you will have to upgrade OAM that resides on machine-1, to 12*c*.

# Removing the IAMSuiteAgent Deployment

The `IAMSuiteAgent` deployment is not supported in 12*c*. Therefore, undeploy the `IAMSuiteAgent` before you proceed with the upgrade.

## Removing `IAMSuiteAgent` from the WebLogic Administration Console

1. Log in to the WebLogic Administration Console using the following URL:

   `http://hostname:port/console`

   where *`hostname`* is the DNS name or IP address of the Administration Server and *`port`* is the listen port on which the Administration Server is listening for requests (port 7001 by default). If you have configured a domain-wide administration port, use that port number. If you configured the Administration Server to use Secure Socket Layer (SSL) you must add *`s`* after `http` as follows:

   `https://hostname:port/console`

   > **Note:**
   >
   > A domain-wide administration port always uses SSL.

2. Click **Security Realms**.

3. Click **myrealm**.
4. Click **Provider**, and then select **IAMSuiteAgent**.
5. Click **Delete**.
6. Restart the servers.

### Removing `IAMSuiteAgent` from the OAM Console

> **Note:**
>
> Before you delete `IAMSuiteAgent` from the OAM console, complete the following tasks:
>
> - Replace `IAMSuiteAgent` with an 11g WebGate. See Replacing the IAMSuiteAgent with an 11g WebGate. Removing `IAMSuiteAgent` without replacing it with an 11g WebGate may result in a loss of the OAM functionalities in the 11g server.
> - Back up the OAM configuration.

1. Log in to the OAM console.
2. Go to the **Application Security** tab, click **Agents**, and then **Managed single sign-on agents**.
3. From the list of SSO agents, select `IAMSuiteAgent`, and then click **Delete**.
4. Confirm the deletion.

## Upgrading Java JSE Policy

Upgrade Java JSE Policy, if required.

> **Note:**
>
> This is required if any of the Identity Management components like Oracle Access Management (OAM), Oracle Identity Manager (OIM), Oracle Adaptive Access Manager (OAAM), or Oracle Access Manager Webgates of a data center are yet to be upgraded to 12*c* (12.2.1.3.0). This is for the phased transition to 12*c* (12.2.1.3.0).
>
> For a Multi Data Center setup, this is required if any of the data centers has 12*c* (12.2.1.2.0) components (OAM, OIM, OAAM, OAM Webgates).

The jar files `local_policy.jar` and `US_export_policy.jar` are present in the directory `$JAVA_HOME/jre/lib/security`. You can upgrade Java JSE policy by overwriting these jar files with the specified versions. To do this, complete the following steps:

1. Download the `local_policy.jar` and `US_export_policy.jar` files from the following location:

http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html

2. Copy the jar files to the location `$JAVA_HOME/jre/lib/security`. This overwrites the existing files.

This completes the Java JSE policy upgrade.

# Installing Product Distributions

Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure and Oracle Access Manager 12*c* (12.2.1.3.0) distributions on the target system and install them using Oracle Universal Installer.

> **Note:**
>
> - The 12c binaries are installed in a different location from the previous 11g binaries. You can install 12c binaries before any planned downtime for upgrade.
> - If you are using Redundant binary locations, ensure that you install the software into each of those redundant locations.

To install the 12*c* (12.2.1.3.0) distributions:

1. Sign in to the target system.
2. Download the following from Oracle Technical Resources or Oracle Software Delivery Cloud to your target system:
   - Oracle Fusion Middleware Infrastructure (`fmw_12.2.1.3.0_infrastructure_generic.jar`)
   - Oracle Access Manager (`fmw_12.2.1.3.0_idm_generic.jar`)
   - Any additional distributions for your pre-upgrade environment

   > **Note:**
   >
   > If you are upgrading an integrated environment that was set up using Life Cycle Management (LCM) tool, that includes Oracle Access Manager, Oracle Identity Manager, and WebGates, then you must install the respective 12*c* Web Server (Oracle HTTP Server or Oracle Traffic Director) binaries in the same Oracle Home.
3. Change to the directory where you downloaded the 12*c* (12.2.1.3.0) product distribution.
4. Start the installation program for Oracle Fusion Middleware Infrastructure:
   - (UNIX) *JAVA_HOME*`/bin/java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`
   - (Windows) *JAVA_HOME*`\bin\java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`
5. On UNIX operating systems, the Installation Inventory Setup screen appears if this is the first time you are installing an Oracle product on this host.

Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location, and click **Next**.

> **Note:**
>
> The Installation Inventory Setup screen does not appear on Windows operating systems.

6. On the Welcome screen, review the information to make sure that you have met all the prerequisites. Click **Next**.
7. On the Auto Updates screen, select an option:
   - **Skip Auto Updates**: If you do not want your system to check for software updates at this time.
   - **Select patches from directory**: To navigate to a local directory if you downloaded patch files.
   - **Search My Oracle Support for Updates**: To automatically download software updates if you have a My Oracle Support account. You must enter Oracle Support credentials then click **Search**. To configure a proxy server for the installer to access My Oracle Support, click **Proxy Settings**. Click **Test Connection** to test the connection.

   Click **Next**.
8. On the Installation Location screen, specify the location for the Oracle home directory and click **Next**.

   For more information about Oracle Fusion Middleware directory structure, see About the Directories for Installation and Configuration in *Planning an Installation of Oracle Fusion Middleware*.
9. On the Installation Type screen, select the following:
   - For Infrastructure, select **Fusion Middleware Infrastructure**
   - For Oracle Access Manager, select **Collocated Oracle Identity and Access Manager**.

   Click **Next**.
10. The Prerequisite Checks screen analyzes the host computer to ensure that the specific operating system prerequisites have been met.

    To view the list of tasks that are verified, select **View Successful Tasks**. To view log details, select **View Log**. If any prerequisite check fails, then an error message appears at the bottom of the screen. Fix the error and click **Rerun** to try again. To ignore the error or the warning message and continue with the installation, click **Skip** (not recommended).
11. On the Installation Summary screen, verify the installation options that you selected.

    If you want to save these options to a response file, click **Save Response File** and enter the response file location and name. The response file collects and stores all the information that you have entered, and enables you to perform a silent installation (from the command line) at a later time.
    Click **Install** to begin the installation.

12. On the Installation Progress screen, when the progress bar displays 100%, click **Finish** to dismiss the installer, or click **Next** to see a summary.
13. The Installation Complete screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.
14. After you have installed Oracle Fusion Middleware Infrastructure, enter the following command to start the installer for your product distribution and repeat the steps above to navigate through the installer screens:

    (UNIX) `JAVA_HOME/bin/java -jar fmw_12.2.1.3.0_idm_generic.jar`

    (Windows) `JAVA_HOME\bin\java -jar fmw_12.2.1.3.0_idm_generic.jar`

> **Note:**
>
> - If your 11.1.2.3.0 setup was deployed using Life Cycle Management (LCM) tool, you must install Oracle HTTP Server 12*c* (12.2.1.3.0) in the 12*c* Middleware home. See Preparing to Install and Configure Oracle HTTP Server in *Installing and Configuring Oracle HTTP Server*.
> - By using the opatch tool, apply the latest recommended patchsets from Oracle Support. Complete only the binary installation of patchsets and follow any post-patch steps after the upgrade process is complete. This provides the latest known fixes for upgrade process, if any.

# Installing the Latest Stack Patch Bundle

After you install the product distributions, Oracle strongly recommends you to apply the latest IDM Stack Patch Bundle (SPB) 12.2.1.3.0 before proceeding with the upgrade process. You can apply the patch by using the Opatch tool. Applying the SPB helps eliminate most of the upgrade issues or workarounds.

Following are the high-level tasks you should complete to apply the Stack Patch Bundle:

- **Initial Preparation**: In this phase, you stage the software, read the `README.txt` file, and verify and/or update the Opatch tool to the appropriate versions.
- **Analysis Phase**: In this phase, you run the `prestop` command with the variables from the `README.txt` file to determine if the system is ready for patching.
- **Patching Phase**: In this phase, you backup *MW_HOME* and *DOMAIN_HOME*, run the downtime command for OIG with the variables from the `README.txt` file, and then clear any temporary files.

> **Note:**
>
> At this point, you will not restart the servers. There is currently no link between the schemas, the local configuration, and the new bits. The remainder of the patching process will happen after the bootstrap.

To avoid a false failure during the domain Reconfiguration Phase of the upgrade, after completing the Patching Phase, update the following entries in the `config.xml` for the `com.oracle.cie.comdev_7.8.2.0` and `com.oracle.cie.xmldh_3.4.2.0` libraries:

```
<name>com.oracle.cie.comdev#3.0.0.0@7.8.2.0</name>
com.oracle.cie.comdev_7.8.2.0.jar


<name>com.oracle.cie.xmldh#2.0.0.0@3.4.2.0</name>
com.oracle.cie.xmldh_3.4.2.0.jar
```

From:

```
<library>
<name>com.oracle.cie.comdev#3.0.0.0@7.8.2.0</name>
<target>oim_cluster</target>
<source-path><MW_HOME>/oracle_common/modules/
com.oracle.cie.comdev_7.8.2.0.jar
</source-path>
<deployment-order>511</deployment-order>
<security-dd-model>DDOnly</security-dd-model>
<staging-mode>nostage</staging-mode>
</library>

<library>
<name>com.oracle.cie.xmldh#2.0.0.0@3.4.2.0</name>
<target>oim_cluster</target>
<source-path><MW_HOME>/oracle_common/modules/
com.oracle.cie.xmldh_3.4.2.0.jar<
/source-path>
<deployment-order>511</deployment-order>
<security-dd-model>DDOnly</security-dd-model>
<staging-mode>nostage</staging-mode>
</library>
```

To this:

```
<library>
<name>com.oracle.cie.comdev#3.0.0.0@7.8.4.0</name>
<target>oim_cluster</target>
<source-path><MW_HOME>/oracle_common/modules/
com.oracle.cie.comdev_7.8.4.0.jar
</source-path>
<deployment-order>511</deployment-order>
<security-dd-model>DDOnly</security-dd-model>
<staging-mode>nostage</staging-mode>
</library>

<library>
<name>com.oracle.cie.xmldh#2.0.0.0@3.4.4.0</name>
<target>oim_cluster</target>
<source-path><MW_HOME>/oracle_common/modules/
com.oracle.cie.xmldh_3.4.4.0.jar<
```

```
/source-path>
<deployment-order>511</deployment-order>
<security-dd-model>DDOnly</security-dd-model>
<staging-mode>nostage</staging-mode>
</library>
```

This update to the `config.xml` file changes the name of the libraries and version of the jar file in each library to the one that will be used post the patching process. If it is a cluster, ensure that both nodes have these settings.

For more information on the patching process, see Doc ID 2657920.1.

> **Note:**
>
> If you are using Windows or Solaris OS, download the individual Bundle Patches (BPs) from Doc ID 2457034.1.

After completing the upgrade, you have to perform the post-patch install steps. See Performing the Post-Patch Install Steps.

# Creating the Required 12*c* Schemas Using RCU

When upgrading from 11g, you must create extra schemas required for 12c. If your setup has non-SSL ports open, you can use the Upgrade Assistant to create schemas by using the default schema settings. In case of SSL enabled setup, you can use the Repository Creation Utility (RCU) to create customized schemas. This procedure describes how to create schemas using the RCU. Information about using the Upgrade Assistant to create schemas is covered in the upgrade procedures.

> **Note:**
>
> You must use the 12*c* Repository Creation Utility (RCU) to create the 12*c* schemas. 12*c* RCU is located at *ORACLE_HOME*`/oracle_common/bin` directory, where *ORACLE_HOME* is the 12*c* Oracle Home.

You must create the following schemas using 12*c* RCU:

- Common Infrastructure Services Service Table (*prefix*_STB)
- WebLogic Services (*prefix*_WLS)
- User Messaging Service (*prefix*_UMS)

The existing schemas such as Oracle Access Manager (OAM), Oracle Platform Security Services (OPSS) will be upgraded, and therefore, you do not have to create new ones.

The following schemas must exist before you upgrade to 12*c*. If you are upgrading from 11*g*, and you are not sure which schemas you currently have, refer to the steps below to identify the existing schemas in your domain. You do not need to re-create these schemas if they already exist.

- **Service Table** schema (*prefix*_STB). This schema is new in 12*c* and is required for domain-based upgrades. It stores basic schema configuration information (for example, schema prefixes and passwords) that can be accessed and used by other Oracle Fusion Middleware components during the domain creation. This schema is automatically created when you run the Repository Creation Utility (RCU), where you specify the existing schema owner prefix that you used for your other 11*g* schemas.

  > **Note:**
  >
  > If the Service Table schema does not exist, you may encounter the error message `UPGAST-00328 : The schema version registry table does not exist on this database. If that happens it is necessary to create the service table schema in order to run Upgrade Assistant`

- **Oracle Platform Security Services (OPSS)** schema (*prefix*_OPSS). This schema is required if you are using an OID-based security store in 11*g*. This schema is automatically created when you run the Repository Creation Utility (RCU). The only supported LDAP-based OPSS security store is Oracle Internet Directory (OID). An LDAP-based policy store is typically used in production environments. You do not need to reassociate an OID-based security store before upgrade. While the Upgrade Assistant is running, you can select the OPSS schema. The Upgrade Assistant upgrades the OID-based security store automatically.

  > **Note:**
  >
  > The 12*c* OPSS database schema is required so that you can reference the 12*c* schema during the reconfiguration of the domain. Your domain continues to use the OID-based security store after the upgrade is complete.

To create the 12*c* schemas with the RCU:

1. (Optional) If you are upgrading from 11*g*, and you wish to confirm the schemas which are present in your existing domain, then connect to the database as a user with DBA privileges, and run the following code from SQL*Plus:

   ```
   SET LINE 120
   COLUMN MRC_NAME FORMAT A14
   COLUMN COMP_ID FORMAT A20
   COLUMN VERSION FORMAT A12
   COLUMN STATUS FORMAT A9
   COLUMN UPGRADED FORMAT A8
   SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM
   SCHEMA_VERSION_REGISTRY ORDER BY MRC_NAME, COMP_ID ;
   ```

2. Verify that a certified JDK already exists on your system by running `java -version` from the command line. For 12*c* (12.2.1.3.0), the certified JDK is 1.8.0_131 and later.

   Ensure that the `JAVA_HOME` environment variable is set to the location of the certified JDK. For example:

- (UNIX) `setenv JAVA_HOME=/home/Oracle/Java/jdk1.8.0_131`
- (Windows) `set JAVA_HOME=C:\home\Oracle\Java\jdk1.8.0_131`

Add `$JAVA_HOME/bin` to `$PATH`.

3. Go to the `oracle_common/bin` directory:
   - (UNIX) *`ORACLE_HOME`*`/oracle_common/bin`
   - (Windows) *`ORACLE_HOME`*`\oracle_common\bin`
4. Start the RCU:
   - (UNIX) `./rcu`
   - (Windows) `rcu.bat`
5. On the Welcome screen, click **Next**.
6. On the Create Repository screen, select **Create Repository** and then select **System Load and Product Load**.

   If you do not have DBA privileges, select **Prepare Scripts for System Load**. This will generate a SQL script containing all the same SQL statements and blocks that would have been called if the RCU were to execute the actions for the selected components. After the script is generated, the user you created earlier, in Creating a Non-SYSDBA User to Run the Upgrade Assistant, with the necessary SYS or SYSDBA privileges can execute the script to complete the system load phase.

   Click **Next**.
7. On the Database Connection Details screen, select the **Database Type** and enter the connection information for the database that hosts the 11*g* schemas. See the pertinent table below.

**Table 3-2 Connection Credentials for Oracle Databases and Oracle Databases with Edition-Based Redefinition**

| Option | Description and Example |
|---|---|
| Host Name | Specify the name of the server where your database is running in the following format:<br>`examplehost.exampledomain.com`<br>For Oracle RAC databases, specify the SCAN name or one of the node names in this field. |
| Port | Specify the port number for your database. The default port number for Oracle databases is `1521`. |
| Service Name | Specify the service name for the database. Typically, the service name is the same as the global database name.<br>For Oracle RAC databases, specify the service name of one of the nodes in this field. For example:<br>`examplehost.exampledomain.com` |
| Username | Enter the user name for your database. The default user name is `SYS`. |
| Password | Enter the password for your database user. |
| Role | Select the database user's role from the drop-down list:<br>**Normal** or **SYSDBA** |

**Table 3-3 Connection Credentials for MySQL Databases**

| Option | Description and Example |
|---|---|
| Host Name | Specify the host name, IP address, or complete server name in *host\server* format of the server where your database is running. |
| Port | Specify the port number for your database. |
| Database Name | Specify the name of your database. |
| Username | Specify the name of a user with administrator privileges. |
| Password | Enter the password for your database user. |

**Table 3-4 Connection Credentials for Microsoft SQL Server Databases**

| Option | Description and Example |
|---|---|
| Unicode Support | Select **Yes** or **No** from the drop-down list. |
| Server Name | Specify the host name, IP address, or complete server name in *host\server* format of the server where your database is running.<br>MSSQL named instances: A named instance is identified by the network name of the computer and the instance name that you specify during installation. The client must specify both the server name and the instance name when connecting. |
| Port | Specify the port number for your database. |
| Database Name | Specify the name of your database. |
| Username | Specify the name of a user with administrator privileges. |
| Password | Enter the password for your database user. |

**Table 3-5 Connection Credentials for IBM DB2 Databases**

| Option | Description and Example |
|---|---|
| Server Name | Specify the host name, IP address, or complete server name in *host\server* format of the server where your database is running. |
| Port | Specify the port number for your database. |
| Database Name | Specify the name of your database. |
| Username | Specify the name of a user with DB Owner privileges. The default user name for IBM DB2 databases is `db2admin`. |
| Password | Enter the password for your database user. |

If the prerequisite check is successful, click **OK** to continue to the next screen. If the check fails, review the details you entered and try again.

8. On the Select Components screen, select **Select existing prefix** and select the prefix that was used to create the existing 11*g* schemas from the drop-down menu (for example, `DEV11G`). This prefix is used to logically group schemas together for use in this domain. Select the following schemas:

   a. Common Infrastructure Services Service Table (*prefix*_STB)

b. WebLogic Services (*prefix*_WLS)

c. User Messaging Service (*prefix*_UMS)

> **Note:**
>
> The Common Infrastructure Services (*prefix*_STB) and Oracle Platform Security Services (*prefix*_OPSS) schemas are selected by default if they have not yet been created.

Make a note of the prefix and schema names for the components you are installing as you will need this information when you configure the installation. Click **Next**.

9. In the Checking Prerequisites dialog, verify that the prerequisites check is successful, then click **OK**.
10. On the Schema Passwords screen, specify the passwords for your schema owners.

    Make a note of the passwords you enter on this screen as you will need this information while configuring your product installation.
11. On the Map Tablespaces screen, configure the required tablespace mapping for the schemas you want to create.

    Click **Next**, then click **OK** in the confirmation dialog. When the progress dialog shows the tablespace creation is complete, click **OK**.

    You see the **Encrypt Tablespace** check box only if you have enabled Transparent Data Encryption (TDE) in the database (Oracle or Oracle EBR) when you start the RCU. Select the **Encrypt Tablespace** check box on the Map Tablespaces screen to encrypt all new tablespaces that the RCU creates.
12. Verify the information on the Summary screen and click **Create** to begin schema creation.

    This screen contains information about the log files that were created from this RCU operation. Click on the name of a particular log file to view the contents of that file.
13. Review the information on the Completion Summary screen to verify that the operation is completed successfully. Click **Close** to complete the schema creation.

# Integrating Access Federation with BI Publisher

Update to the required or latest patches to seamlessly integrate and view Oracle Access Manager audit information on BI Publisher.

Complete the following tasks:
**Task 1**: Integrate Access Audit with OPSS Store [IAU Schema]

Apply the latest patch for Oracle Access Manager 12*c* (12.2.1.3.0) or upgrade to 12c (12.2.1.4).

**Task 2**: Integrate Access Audit with BI Publisher

For Oracle Platform Security Services (OPSS), apply patch 12.2.1.3.181016 or later.

**Task 3**: Intergrate Access Federation Audit

For Oracle Platform Security Services (OPSS), apply patch 12.2.1.3.201013 or later.

# Running a Pre-Upgrade Readiness Check

To identify potential issues with the upgrade, Oracle recommends that you run a readiness check before you start the upgrade process. Be aware that the readiness check may not be able to discover all potential issues with your upgrade. An upgrade may still fail, even if the readiness check reports success.

- About Running a Pre-Upgrade Readiness Check
  You can run the Upgrade Assistant in `-readiness` mode to detect issues before you perform the actual upgrade. You can run the readiness check in GUI mode using the Upgrade Assistant or in silent mode using a response file.
- Starting the Upgrade Assistant in Readiness Mode
  Use the `-readiness` parameter to start the Upgrade Assistant in readiness mode.
- Performing a Readiness Check with the Upgrade Assistant
  Navigate through the screens in the Upgrade Assistant to complete the pre-upgrade readiness check.
- Understanding the Readiness Report
  After performing a readiness check for your domain, review the report to determine whether you need to take any action for a successful upgrade.
- OAM Configuration Upgrade Readiness Checks
  The Upgrade Assistant (UA), when run in the readiness mode, performs several configuration upgrade validation checks. Ensure that each of these validation checks are successful before you proceed with the upgrade process.

## About Running a Pre-Upgrade Readiness Check

You can run the Upgrade Assistant in `-readiness` mode to detect issues before you perform the actual upgrade. You can run the readiness check in GUI mode using the Upgrade Assistant or in silent mode using a response file.

The Upgrade Assistant readiness check performs a read-only, pre-upgrade review of your Fusion Middleware schemas and WebLogic domain configurations that are at a supported starting point. The review is a read-only operation.

The readiness check generates a formatted, time-stamped readiness report so you can address potential issues before you attempt the actual upgrade. If no issues are detected, you can begin the upgrade process. Oracle recommends that you read this report thoroughly before performing an upgrade.

You can run the readiness check while your existing Oracle Fusion Middleware domain is online (while other users are actively using it) or offline.

You can run the readiness check any number of times before performing any actual upgrade. However, do not run the readiness check after an upgrade has been performed, as the report results may differ from the result of pre-upgrade readiness checks.

> **Note:**
>
> To prevent performance from being affected, Oracle recommends that you run the readiness check during off-peak hours.

## Starting the Upgrade Assistant in Readiness Mode

Use the `-readiness` parameter to start the Upgrade Assistant in readiness mode.

To perform a readiness check on your pre-upgrade environment with the Upgrade Assistant:

1. Go to the `oracle_common/upgrade/bin` directory:
   - (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/bin`
   - (Windows) *ORACLE_HOME*`\oracle_common\upgrade\bin`

   Where, *ORACLE_HOME* is the 12c Oracle Home.
2. Start the Upgrade Assistant.
   - (UNIX) `./ua -readiness`
   - (Windows) `ua.bat -readiness`

   > **Note:**
   >
   > If the `DISPLAY` environment variable is not set up properly to allow for GUI mode, you may encounter the following error:
   >
   > ```
   > Xlib: connection to ":1.0" refused by server
   > Xlib: No protocol specified
   > ```
   >
   > To resolve this issue you need to set the `DISPLAY` variable to the host and desktop where a valid `X` environment is working.
   >
   > For example, if you are running an `X` environment inside a VNC on the local host in desktop 6, then you would set `DISPLAY=:6`. If you are running `X` on a remote host on desktop 1 then you would set this to `DISPLAY=remoteHost:1`.

   For information about other parameters that you can specify on the command line, see:

- Upgrade Assistant Parameters

## Upgrade Assistant Parameters

When you start the Upgrade Assistant from the command line, you can specify additional parameters.

**Table 3-6 Upgrade Assistant Command-Line Parameters**

| Parameter | Required or Optional | Description |
|---|---|---|
| `-readiness` | Required for readiness checks<br>**Note**: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server). | Performs the upgrade readiness check without performing an actual upgrade.<br>Schemas and configurations are checked.<br>Do not use this parameter if you have specified the `-examine` parameter. |
| `-threads` | Optional | Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas.<br>The value must be a positive integer in the range 1 to 8. The default is 4. |
| `-response` | Required for silent upgrades or silent readiness checks | Runs the Upgrade Assistant using inputs saved to a response file generated from the data that is entered when the Upgrade Assistant is run in GUI mode. Using this parameter runs the Upgrade Assistant in *silent mode* (without displaying Upgrade Assistant screens). |
| `-examine` | Optional | Performs the examine phase but does not perform an actual upgrade.<br>Do not specify this parameter if you have specified the `-readiness` parameter. |
| `-logLevel` *attribute* | Optional | Sets the logging level, specifying one of the following attributes:<br>• `TRACE`<br>• `NOTIFICATION`<br>• `WARNING`<br>• `ERROR`<br>• `INCIDENT_ERROR`<br>The default logging level is `NOTIFICATION`.<br>Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used. |

**Table 3-6 (Cont.) Upgrade Assistant Command-Line Parameters**

| Parameter | Required or Optional | Description |
|---|---|---|
| `-logDir` *location* | Optional | Sets the default location of upgrade log files and temporary files. You must specify an existing, writable directory where the Upgrade Assistant creates log files and temporary files.<br>The default locations are:<br>(UNIX)<br>*ORACLE_HOME*`/oracle_common/upgrade/logs`<br>*ORACLE_HOME*`/oracle_common/upgrade/temp`<br>(Windows)<br>*ORACLE_HOME*`\oracle_common\upgrade\logs`<br>*ORACLE_HOME*`\oracle_common\upgrade\temp` |
| `-help` | Optional | Displays all of the command-line options. |

# Performing a Readiness Check with the Upgrade Assistant

Navigate through the screens in the Upgrade Assistant to complete the pre-upgrade readiness check.

Readiness checks are performed only on schemas or component configurations that are at a supported upgrade starting point.

To complete the readiness check:

1. On the Welcome screen, review information about the readiness check. Click **Next**.
2. On the Readiness Check Type screen, select the readiness check that you want to perform:
   - **Individually Selected Schemas** allows you to select individual schemas for review before upgrade. The readiness check reports whether a schema is supported for an upgrade or where an upgrade is needed.
     When you select this option, the screen name changes to Selected Schemas.
   - **Domain Based** allows the Upgrade Assistant to discover and select all upgrade-eligible schemas or component configurations in the domain specified in the **Domain Directory** field.
     When you select this option, the screen name changes to Schemas and Configuration.

     Leave the default selection if you want the Upgrade Assistant to check all schemas and component configurations at the same time, or select a specific option:

- **Include checks for all schemas** to discover and review all components that have a schema available to upgrade.
- **Include checks for all configurations** to review component configurations for a managed WebLogic Server domain.

Click **Next**.

3. If you selected **Individually Selected Schemas**: On the Available Components screen, select the components that have a schema available to upgrade for which you want to perform a readiness check.

   If you selected **Domain Based**: On the Component List screen, review the list of components that are present in your domain for which you want to perform a readiness check.

   If you select a component that has dependent components, those components are automatically selected. For example, if you select Oracle Platform Security Services, Oracle Audit Services is automatically selected.

   Depending on the components you select, additional screens may display. For example, you may need to:

   - Specify the Administrator server domain directory.

     Ensure that you specify the 11.1.2.3.0 Administrator server domain directory.

   - Specify schema credentials to connect to the selected schema: **Database Type**, **DBA User Name**, and **DBA Password**. As part of the pre-upgrade requirements, you had created the required user, see Creating a Non-SYSDBA User to Run the Upgrade Assistant.

     Then click **Connect**.

     > **Note:**
     >
     > Oracle database is the default database type. Make sure that you select the correct database type before you continue. If you discover that you selected the wrong database type, do not go back to this screen to change it to the correct type. Instead, close the Upgrade Assistant and restart the readiness check with the correct database type selected to ensure that the correct database type is applied to all schemas.

   - Select the **Schema User Name** option and specify the **Schema Password**.

   Click **Next** to start the readiness check.

4. On the Readiness Summary screen, review the summary of the readiness checks that will be performed based on your selections.

   If you want to save your selections to a response file to run the Upgrade Assistant again later in response (or silent) mode, click **Save Response File** and provide the location and name of the response file. A silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again.

   For a detailed report, click **View Log**.

   Click **Next**.

5. On the Readiness Check screen, review the status of the readiness check. The process can take several minutes.

   If you are checking multiple components, the progress of each component displays in its own progress bar in parallel.

   When the readiness check is complete, click **Continue**.

6. On the End of Readiness screen, review the results of the readiness check (**Readiness Success** or **Readiness Failure**):

   - If the readiness check is successful, click **View Readiness Report** to review the complete report. Oracle recommends that you review the Readiness Report before you perform the actual upgrade even when the readiness check is successful. Use the **Find** option to search for a particular word or phrase within the report. The report also indicates where the completed Readiness Check Report file is located.
   - If the readiness check encounters an issue or error, click **View Log** to review the log file, identify and correct the issues, and then restart the readiness check. The log file is managed by the command-line options you set.

## Understanding the Readiness Report

After performing a readiness check for your domain, review the report to determine whether you need to take any action for a successful upgrade.

The format of the readiness report file is:

`readiness_`*`timestamp`*`.txt`

where *`timestamp`* indicates the date and time of when the readiness check was run.

A readiness report contains the following information:

**Table 3-7 Readiness Report Elements**

| Report Information | Description | Required Action |
|---|---|---|
| Overall Readiness Status: SUCCESS or FAILURE | The top of the report indicates whether the readiness check passed or completed with one or more errors. | If the report completed with one or more errors, search for FAIL and correct the failing issues before attempting to upgrade. You can re-run the readiness check as many times as necessary before an upgrade. |
| Timestamp | The date and time that the report was generated. | No action required. |
| Log file location *`ORACLE_HOME`*`/oracle_common/upgrade/logs` | The directory location of the generated log file. | No action required. |
| Readiness report location *`ORACLE_HOME`*`/oracle_common/upgrade/logs` | The directory location of the generated readiness report. | No action required. |

**Table 3-7 (Cont.) Readiness Report Elements**

| Report Information | Description | Required Action |
|---|---|---|
| Names of components that were checked | The names and versions of the components included in the check and status. | If your domain includes components that cannot be upgraded to this release, such as SOA Core Extension, do not attempt an upgrade. |
| Names of schemas that were checked | The names and current versions of the schemas included in the check and status. | Review the version numbers of your schemas. If your domain includes schemas that cannot be upgraded to this release, do not attempt an upgrade. |
| Individual Object Test Status: FAIL | The readiness check test detected an issue with a specific object. | Do not upgrade until all failed issues have been resolved. |
| Individual Object Test Status: PASS | The readiness check test detected no issues for the specific object. | If your readiness check report shows only the PASS status, you can upgrade your environment. Note, however, that the Readiness Check cannot detect issues with externals such as hardware or connectivity during an upgrade. You should always monitor the progress of your upgrade. |
| Completed Readiness Check of <Object> Status: FAILURE | The readiness check detected one or more errors that must be resolved for a particular object such as a schema, an index, or datatype. | Do not upgrade until all failed issues have been resolved. |
| Completed Readiness Check of <Object> Status: SUCCESS | The readiness check test detected no issues. | No action required. |

Here is a sample Readiness Report file. Your report may not include all of these checks.

```
Upgrade readiness check completed with one or more errors.

This readiness check report was created on Tue May 30 11:15:52 EDT 2016
Log file is located at: ORACLE_HOME/oracle_common/upgrade/logs/
ua2016-05-30-11-14-06AM.log
Readiness Check Report File: ORACLE_HOME/oracle_common/upgrade/logs/
readiness2016-05-30-11-15-52AM.txt

Starting readiness check of components.

Oracle Metadata Services
   Starting readiness check of Oracle Metadata Services.
     Schema User Name: DEV11_MDS
     Database Type: Oracle Database
     Database Connect String: machinename@yourcompany.com
     VERSION Schema DEV11_MDS is currently at version 12.1.1.1.0.
Readiness checks will now be performed.
   Starting schema test:  TEST_REQUIRED_TABLES  Test that the schema
contains all the required tables
```

```
   Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema
contains all the required tables +++ PASS
   Starting schema test:  TEST_REQUIRED_PROCEDURES  Test that the schema
contains all the required stored procedures
     EXCEPTION     Schema is missing a required procedure:
GETREPOSITORYFEATURES
   Completed schema test: TEST_REQUIRED_PROCEDURES --> Test that the schema
contains all the required stored procedures +++ FAIL
   Starting schema test:  TEST_REQUIRED_VIEWS  Test that the schema contains
all the required database views
   Completed schema test: TEST_REQUIRED_VIEWS --> Test that the schema
contains all the required database views +++ PASS
   Starting index test for table MDS_ATTRIBUTES:  TEST_REQUIRED_INDEXES -->
Test that the table contains all the required indexes
   Completed index test for table MDS_ATTRIBUTES: TEST_REQUIRED_INDEXES -->
Test that the table contains all the required indexes +++ PASS
   Starting index test for table MDS_COMPONENTS:  TEST_REQUIRED_INDEXES -->
Test that the table contains all the required indexes
   Completed index test for table MDS_TXN_LOCKS: TEST_REQUIRED_INDEXES -->
Test that the table contains all the required indexes +++ PASS
   Starting schema test:  TEST_REQUIRED_TRIGGERS  Test that the schema has
all the required triggers
   Completed schema test: TEST_REQUIRED_TRIGGERS --> Test that the schema
has all the required triggers +++ PASS
   Starting schema test:  TEST_MISSING_COLUMNS  Test that tables and views
are not missing any required columns
   Completed schema test: TEST_MISSING_COLUMNS --> Test that tables and
views are not missing any required columns +++ PASS
   Starting schema test:  TEST_UNEXPECTED_TABLES  Test that the schema does
not contain any unexpected tables
   Completed schema test: TEST_UNEXPECTED_TABLES --> Test that the schema
does not contain any unexpected tables +++ PASS
   Starting schema test:  TEST_UNEXPECTED_PROCEDURES  Test that the schema
does not contain any unexpected stored procedures
   Completed schema test: TEST_UNEXPECTED_PROCEDURES --> Test that the
schema does not contain any unexpected stored procedures +++ PASS
   Starting schema test:  TEST_UNEXPECTED_VIEWS  Test that the schema does
not contain any unexpected views
   Completed schema test: TEST_UNEXPECTED_VIEWS --> Test that the schema
does not contain any unexpected views +++ PASS
   Starting index test for table MDS_ATTRIBUTES:  TEST_UNEXPECTED_INDEXES --
> Test that the table does not contain any unexpected indexes
   Completed index test for table MDS_ATTRIBUTES: TEST_UNEXPECTED_INDEXES --
> Test that the table does not contain any unexpected indexes +++ PASS
   Completed index test for table MDS_LABELS: TEST_UNEXPECTED_INDEXES -->
Test that the table does not contain any unexpected indexes +++ PASS
   Starting index test for table MDS_LARGE_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes
   Starting schema test:  TEST_UNEXPECTED_TRIGGERS  Test that the schema
does not contain any unexpected triggers
   Completed schema test: TEST_UNEXPECTED_TRIGGERS --> Test that the schema
does not contain any unexpected triggers +++ PASS
   Starting schema test:  TEST_UNEXPECTED_COLUMNS  Test that tables and
views do not contain any unexpected columns
```

```
   Completed schema test: TEST_UNEXPECTED_COLUMNS --> Test that tables
and views do not contain any unexpected columns +++ PASS
   Starting datatype test for table MDS_ATTRIBUTES:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes
   Completed datatype test for table MDS_ATTRIBUTES:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes +++ PASS
   Starting datatype test for table MDS_COMPONENTS:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes
   Starting permissions test:  TEST_DBA_TABLE_GRANTS  Test that DBA
user has privilege to view all user tables
   Completed permissions test: TEST_DBA_TABLE_GRANTS --> Test that DBA
user has privilege to view all user tables +++ PASS
   Starting schema test:  TEST_ENOUGH_TABLESPACE  Test that the schema
tablespaces automatically extend if full
   Completed schema test: TEST_ENOUGH_TABLESPACE --> Test that the
schema tablespaces automatically extend if full +++ PASS
   Starting schema test:  TEST_USER_TABLESPACE_QUOTA  Test that
tablespace quota for this user is sufficient to perform the upgrade
   Completed schema test: TEST_USER_TABLESPACE_QUOTA --> Test that
tablespace quota for this user is sufficient to perform the upgrade ++
+ PASS
   Starting schema test:  TEST_ONLINE_TABLESPACE  Test that schema
tablespaces are online
   Completed schema test: TEST_ONLINE_TABLESPACE --> Test that schema
tablespaces are online +++ PASS
   Starting schema test:  TEST_DATABASE_VERSION  Test that the
database server version number is supported for upgrade
     INFO   Database product version: Oracle Database 11g Enterprise
Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing
options
   Completed schema test: TEST_DATABASE_VERSION --> Test that the
database server version number is supported for upgrade +++ PASS
   Finished readiness check of Oracle Metadata Services with status:
FAILURE.
```

If you are running the 12.1.3.0 version of Oracle Fusion Middleware IAU Schemas, and those schemas were upgraded from 11*g* (11.1.1.7 and later) or 12*c* (12.1.2.0), your readiness check may fail with the following error:

```
Starting index test for table IAU_COMMON:  TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes
     INFO Audit schema index DYN_EVENT_CATEGORY_INDEX in table IAU_COMMON is
missing the required columns or index itself is missing. This maybe caused by
a known issue, anyway, this missing index will be added in 12.2.2 upgrade.
     INFO Audit schema index DYN_EVENT_TYPE_INDEX in table IAU_COMMON is
missing the required columns or index itself is missing. This maybe caused by
a known issue, anyway, this missing index will be added in 12.2.2 upgrade.
     INFO Audit schema index DYN_TENANT_INDEX in table IAU_COMMON is missing
the required columns or index itself is missing. This maybe caused by a known
issue, anyway, this missing index will be added in 12.2.2 upgrade.
     INFO Audit schema index DYN_USER_INDEX in table IAU_COMMON is missing
```

```
the required columns or index itself is missing. This maybe caused by a known
issue, anyway, this missing index will be added in 12.2.2 upgrade.
     INFO Audit schema index DYN_COMPONENT_TYPE_INDEX in table IAU_COMMON is
missing the required columns or index itself is missing. This maybe caused by
a known issue, anyway, this missing index will be added in 12.2.2 upgrade.
     INFO Audit schema index DYN_USER_TENANT_INDEX in table IAU_COMMON is
missing the required columns or index itself is missing. This maybe caused by
a known issue, anyway, this missing index will be added in 12.2.2 upgrade.
   Completed index test for table IAU_COMMON: TEST_REQUIRED_INDEXES --> Test
that the table contains all the required indexes +++ FAIL
```

> **Note:**
>
> You can ignore the missing index error in the readiness report. This is a known issue. The corresponding missing index is added during the schema upgrade operation. This error does not occur if the schema to be upgraded was created in 12*c* using the RCU.

## OAM Configuration Upgrade Readiness Checks

The Upgrade Assistant (UA), when run in the readiness mode, performs several configuration upgrade validation checks. Ensure that each of these validation checks are successful before you proceed with the upgrade process.

> **Note:**
>
> For UA to perform the config upgrade readiness checks described below, ensure that you apply the latest bundle patch that contains the fix for Bug 32081498.

The UA performs the following validation checks:

- **Validation of the OPSS and OAM Keystores (Check Name: OAM_OPSS_KEYSTORE_CHECK)**
  The UA extracts the Credential Store Framework (CSF) key from both the OPSS keystore and the OAM keystore and compares them. If either of these keystores is corrupted, the read operation fails and consequently, the readiness check also fails. If the CSF keys are read successfully, they must be identical. Otherwise, the readiness check fails.

  Reasons for the readiness check failure and suggestions to resolve them:

  – If the readiness check fails because the `jks key` or the `keystore-csf-key` is not present in the CSF (the OAM keystore validation fails), modify or add the keystore password. See Doc ID 2710662.1 or Doc ID 2642638.1.

  – If the readiness check fails because the values of the `jks key` and the `keystore-csf-key` are not equal, correct the `keystore-csf-key` value to be same as the `jks key` value. See Doc ID 2710664.1 or Doc ID 2642638.1.

  – If the readiness check fails because the OAM keystore file (`.oamkeystore`) is not present, restore the file from a backup, and then restart the OAM administration server and the managed server to re-create the `.oamkeystore` file. See Doc ID 2710664.1.

  - If the readiness check fails to decrypt `sslGlobalPassphrase`, which is present in the `oam-config.xml` file, see Doc ID 2710716.1.

- **Validation of OAM Configuration Version Consistency (Check Name: OAM_CONFIG_VERSION_CHECK)**
  The UA verifies that the OAM configuration version that is set in the `oam-config.xml` file system is consistent with the version set in the database. The UA extracts the schema credentials and database connection details to fetch the `oam-config` version from the database. It then reads the version of `oam-config.xml` in the file system and compares the version with the `oam-config` version it fetched from the database. If the two versions match, the readiness check succeeds; otherwise, it fails.

  The version mismatch occurs when OAM uses the incorrect datasource name. Configure the correct OAM datasource to resolve the issue. See Doc ID 2492188.1.

- **Validation of the Default Keystore File (Check Name: OAM_DEFAULT_KEYSTORE_CHECK)**
  The UA checks the existence and validity of the default keystore file, `default-keystore.jks`. If the file is not present before the upgrade, readiness check fails. Restart the OAM server to generate the `default-keystore.jks` file.

  If the file exists, but fails to open using the CSF key, it is considered as invalid. The invalid file causes the readiness check to fail. This failure is shown as a failure to decrypt `sslGlobalPassphrase`, which is present in `oam-config.xml`.

  The invalid `default-keystore.jks` file may be due to the corrupt OAM key (`oamKey`). To resolve this issue, take a backup of the `.oamkeystore` file, remove it from `<domain_home>/config/fmwconfig`, restore the file from a backup, and then restart the OAM administration server and the managed server to re-create the `.oamkeystore` file. See Doc ID 2710664.1.

- **Check for the Existence of Unsupported Agents (Check Name: OBSOLETE_AGENT_CHECK)**
  The UA checks the existence of the following agents in the environment:

  - 10g OSSO agent
  - OpenSSO agent
  - OAM 10g WebGate agent
  - Coexistence agent

  If any of these agents exist, the readiness check requirement is not met. For a description of the unsupported agents, see Features Not Supported in Access Manager 12.2.1.3.0 in *Release Notes for Oracle Identity Management*.

  Remove the unsupported agents before you start the upgrade.

- **Validation of the Coherence Keystore (Check Name: COHERENCE_KEYSTORE_CHECK)**
  The UA extracts the CSF key from the keystore and loads the Coherence keystore. It then checks the presence of the key alias `admin` and the certificate alias `assertion-cert` in the Coherence keystore. Both the key values must be present in the keystore. This readiness check succeeds if the Coherence keystore is loaded properly and both the keys are present in it. Otherwise, the check fails.

  If the check fails, create the missing keys and values in the Coherence keystore, and then validate the keystore. See Doc ID 1986560.1.

Here is a sample Readiness Report file. This report shows the portion relevant to OAM schema and configuration upgrade readiness checks:

```
Upgrade readiness check completed with one or more errors.

This readiness check report was created on Wed Sep 16 15:50:33 PDT 2020
Log file is located at: /scratch/idmqa/tmp/ua2020-09-16-15-40-18PM.log
Readiness Check Report File: /scratch/idmqa/tmp/
readiness2020-09-16-15-50-33PM.txt
Domain Directory: /scratch/idmqa/work/mw35/user_projects/domains/WLS_IDM

Starting readiness check of components.

...

Oracle Access Management Suite (Schema Upgrade)
   Starting readiness check of Oracle Access Management Suite.
     Schema User Name: UPG_OAM
     Database Type: Oracle Database
     Database Connect String: slc11ykm.us.oracle.com:1521:db6844
   Starting schema test:  TEST_DATABASE_VERSION  Test that the database
server version number is supported for upgrade
     INFO   Database product version: Oracle Database 11g Enterprise Edition
Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
   Completed schema test: TEST_DATABASE_VERSION --> Test that the database
server version number is supported for upgrade +++ PASS
   Starting schema test:  OAM_CONFIG_VERSION_CHECK  Test to check OAM Config
version in Database and XML file are equal or not
     INFO OAM Config version from Database: 105
     INFO OAM Config version from XML: 105
     INFO OAM Config version in Database and oam-config.xml are equal
   Completed schema test: OAM_CONFIG_VERSION_CHECK --> Test to check OAM
Config version in Database and XML file are equal or not +++ PASS
   Finished readiness check of Oracle Access Management Suite with status:
SUCCESS.

...

Oracle Access Management Suite (Config Upgrade)
   Starting readiness check of Oracle Access Management Suite.
   Starting config test:  CUSTOM_AUTH_PROVIDER_CHECK  Check that custom auth
provider exist.
   Completed config test: CUSTOM_AUTH_PROVIDER_CHECK --> Check that custom
auth provider exist. +++ PASS
   Starting config test:  SOURCE_CONFIG_CHECK  Test that OAM System
configuration is valid.
   Completed config test: SOURCE_CONFIG_CHECK --> Test that OAM System
configuration is valid. +++ PASS
   Starting config test:  OAM_OPSS_KEYSTORE_CHECK.  Test that OAM and OPSS
keys are valid.
   Completed config test: OAM_OPSS_KEYSTORE_CHECK. --> Test that OAM and
OPSS keys are valid. +++ PASS
   Starting config test:  OAM_DEFAULT_KEYSTORE_CHECK  Check that the default
keystore is present and valid
```

```
     INFO Checking default keystore file: /scratch/idmqa/work/mw35/
user_projects/domains/WLS_IDM/config/fmwconfig//default-keystore.jks
     INFO Default keystore file exists and is valid
   Completed config test: OAM_DEFAULT_KEYSTORE_CHECK --> Check that
the default keystore is present and valid +++ PASS
   Starting config test:  POLICY_PROVIDER_CHECK  Check that policy
provider is valid
   Completed config test: POLICY_PROVIDER_CHECK --> Check that policy
provider is valid +++ PASS
   Starting config test:  OBSOLETE_AGENT_CHECK  Check that no obsolete
agent exist in oam-config.xml.
     INFO OAM agent co-existence setting is disabled.
     INFO No OpenSSO agent instance exist.
     INFO No OSSO agent instance exist.
     WARNING Please remove following 10G webgate agent before upgrade:
IAMSuiteAgent.
   Completed config test: OBSOLETE_AGENT_CHECK --> Check that no
obsolete agent exist in oam-config.xml. +++ FAIL
   Finished readiness check of Oracle Access Management Suite with
status: FAILURE.

...

Finished readiness check of components.
```

# Stopping Servers and Processes

Before you run the Upgrade Assistant to upgrade your schemas and configurations, you must shut down all of the pre-upgrade processes and servers, including the Administration Server, Node manager, and any managed servers.

An Oracle Fusion Middleware environment can consist of an Oracle WebLogic Server domain, an Administration Server, multiple managed servers, Java components, system components such as Identity Management components, and a database used as a repository for metadata. The components may be dependent on each other, so they must be stopped in the correct order.

> **Note:**
>
> The procedures in this section describe how to stop the existing, pre-upgrade servers and processes using the WLST command-line utility or a script. You can also use the Oracle Fusion Middleware Control and the Oracle WebLogic Server Administration Console. See Starting and Stopping Administration and Managed Servers and Node Manager.

> **Note:**
>
> Stop all of the servers in your deployment, except for the Database. The Database must be up during the upgrade process.

To stop your pre-upgrade Fusion Middleware environment, navigate to the pre-upgrade domain and follow the steps below.

### Step 1: Stop System Components

To stop 11g system components, such as Oracle HTTP Server, use the `opmnctl` script:

> **Note:**
>
> If the Oracle HTTP server is shared with other services, then you can choose *not* to stop the Oracle HTTP server.

- (UNIX) *OHS_INSTANCE_HOME*`/bin/opmnctl stopall`
- (Windows) *OHS_INSTANCE_HOME*`\bin\opmnctl stopall`

You can stop system components in any order.

### Step 2: Stop the Managed Servers

Depending on the method you followed to start the managed servers, follow one of the following methods to stop the WebLogic Managed Server:

**Method 1**: To stop a WebLogic Server Managed Server not managed by Node Manager:

- (UNIX) *DOMAIN_HOME*`/bin/stopManagedWebLogic.sh` *managed_server_name admin_url*
- (Windows) *DOMAIN_HOME*`\bin\stopManagedWebLogic.cmd` *managed_server_name admin_url*

When prompted, enter your user name and password.

**Method 2**: To stop a WebLogic Server Managed Server by using the Weblogic Console:

- Log into Weblogic console as a `weblogic` Admin.
- Go to **Servers** > **Control** tab.
- Select the required managed server.
- Click **Shutdown**.

**Method 3**: To stop a WebLogic Server Managed Server using node manager, run the following commands:

```
wls:/offline>nmConnect('nodemanager_username','nodemanager_password',
            'AdminServerHostName','5556','domain_name',
            'DOMAIN_HOME')

wls:/offline>nmKill('ManagedServerName')
```

### Step 3: Stop the Administration Server

When you stop the Administration Server, you also stop the processes running in the Administration Server, including the WebLogic Server Administration Console and Fusion Middleware Control.

Follow one of the following methods to stop the Administration Server:

**Method 1**: To stop the Administration Server not managed by Node Manager:

- (UNIX) *DOMAIN_HOME*/bin/stopWebLogic.sh
- (Windows) *DOMAIN_HOME*\bin\stopWebLogic.cmd

When prompted, enter your user name, password, and the URL of the Administration Server.

**Method 2**: To stop a Administration Server by using the Weblogic Console:

- Log into Weblogic console as a `weblogic` Admin.
- Go to **Servers** > **Control** tab.
- Select the required admin server.
- Click **Shutdown**.

**Method 3**: To stop a WebLogic Server Managed Server using node manager, run the following commands:

```
wls:/offline>nmConnect('nodemanager_username','nodemanager_password',
            'AdminServerHostName','5556','domain_name',
            'DOMAIN_HOME')

wls:/offline>nmKill('AdminServer')
```

#### Step 4: Stop Node Manager

To stop Node Manager, run the following command:

```
kill $(ps -ef | grep nodemanager | awk '{print $2}')
```

# Upgrading Product Schemas

After stopping servers and processes, use the Upgrade Assistant to upgrade supported product schemas to the current release of Oracle Fusion Middleware.

The Upgrade Assistant allows you to upgrade individually selected schemas or all schemas associated with a domain. The option you select determines which Upgrade Assistant screens you will use.

- Identifying Existing Schemas Available for Upgrade
  This optional task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.
- Starting the Upgrade Assistant
  Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.
- Upgrading Oracle Access Manager Schemas Using the Upgrade Assistant
  Navigate through the screens in the Upgrade Assistant to upgrade the product schemas.

- Verifying the Schema Upgrade
  After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

# Identifying Existing Schemas Available for Upgrade

This optional task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.

You can let the Upgrade Assistant upgrade all of the schemas in the domain, or you can select individual schemas to upgrade. To help decide, follow these steps to view a list of all the schemas that are available for an upgrade:

1. If you are using an Oracle database, connect to the database by using an acount that has Oracle DBA privileges, and run the following from SQL*Plus:

   ```
   SET LINE 120
   COLUMN MRC_NAME FORMAT A14
   COLUMN COMP_ID FORMAT A20
   COLUMN VERSION FORMAT A12
   COLUMN STATUS FORMAT A9
   COLUMN UPGRADED FORMAT A8
   SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM
   SCHEMA_VERSION_REGISTRY ORDER BY MRC_NAME, COMP_ID;
   ```

2. Examine the report that is generated.

   If an upgrade is not needed for a schema, the `schema_version_registry` table retains the schema at its pre-upgrade version.

3. Note the schema prefix name that was used for your existing schemas. You will use the same prefix when you create new 12*c* schemas.

**Notes:**

- If you used an OID-based policy store in 11*g*, make sure to create a new OPSS schema before you perform the upgrade. After the upgrade, the OPSS schema remains an LDAP-based store.
- You can only upgrade schemas for products that are available for upgrade in Oracle Fusion Middleware release 12*c* (12.2.1.3.0). Do not attempt to upgrade a domain that includes components that are not yet available for upgrade to 12*c* (12.2.1.3.0).

# Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.

To start the Upgrade Assistant:

> **Note:**
>
> Before you start the Upgrade Assistant, make sure that the JVM character encoding is set to UTF-8 for the platform on which the Upgrade Assistant is running. If the character encoding is not set to UTF-8, then you will not be able to download files containing Unicode characters in their names. This can cause the upgrade to fail.
>
> To ensure that UTF-8 is used by the JVM, use the JVM option `-Dfile.encoding=UTF-8`.

1. Go to the `oracle_common/upgrade/bin` directory:
   - (UNIX) *`ORACLE_HOME`*`/oracle_common/upgrade/bin`
   - (Windows) *`ORACLE_HOME`*`\oracle_common\upgrade\bin`
2. Start the Upgrade Assistant:
   - (UNIX) `./ua`
   - (Windows) `ua.bat`

> **Note:**
>
> In the above command, *`ORACLE_HOME`* refers to the 12*c* (12.2.1.3.0) Oracle Home.

For information about other parameters that you can specify on the command line, such as logging parameters, see:

- Upgrade Assistant Parameters

# Upgrade Assistant Parameters

When you start the Upgrade Assistant from the command line, you can specify additional parameters.

**Table 3-8 Upgrade Assistant Command-Line Parameters**

| Parameter | Required or Optional | Description |
|---|---|---|
| `-readiness` | Required for readiness checks<br>**Note**: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server). | Performs the upgrade readiness check without performing an actual upgrade.<br>Schemas and configurations are checked.<br>Do not use this parameter if you have specified the `-examine` parameter. |
| `-threads` | Optional | Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas.<br>The value must be a positive integer in the range 1 to 8. The default is 4. |
| `-response` | Required for silent upgrades or silent readiness checks | Runs the Upgrade Assistant using inputs saved to a response file generated from the data that is entered when the Upgrade Assistant is run in GUI mode. Using this parameter runs the Upgrade Assistant in *silent mode* (without displaying Upgrade Assistant screens). |
| `-examine` | Optional | Performs the examine phase but does not perform an actual upgrade.<br>Do not specify this parameter if you have specified the `-readiness` parameter. |
| `-logLevel` *attribute* | Optional | Sets the logging level, specifying one of the following attributes:<br>• `TRACE`<br>• `NOTIFICATION`<br>• `WARNING`<br>• `ERROR`<br>• `INCIDENT_ERROR`<br>The default logging level is `NOTIFICATION`.<br>Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used. |

**Table 3-8 (Cont.) Upgrade Assistant Command-Line Parameters**

| Parameter | Required or Optional | Description |
|---|---|---|
| `-logDir` *location* | Optional | Sets the default location of upgrade log files and temporary files. You must specify an existing, writable directory where the Upgrade Assistant creates log files and temporary files.<br>The default locations are:<br>(UNIX)<br>*ORACLE_HOME*`/oracle_common/upgrade/logs`<br>*ORACLE_HOME*`/oracle_common/upgrade/temp`<br>(Windows)<br>*ORACLE_HOME*`\oracle_common\upgrade\logs`<br>*ORACLE_HOME*`\oracle_common\upgrade\temp` |
| `-help` | Optional | Displays all of the command-line options. |

# Upgrading Oracle Access Manager Schemas Using the Upgrade Assistant

Navigate through the screens in the Upgrade Assistant to upgrade the product schemas.

> **Caution:**
>
> You can skip this step if you have already upgraded your schemas using RCU.

> **Note:**
>
> - If the pre-upgrade environment has Audit schema (IAU), you must first upgrade Audit schema only, using the **Individually Selected Schema** option on the Selected Schemas screen, and selecting **Oracle Audit Services** schema. Ensure that you select the appropriate IAU schema from the list of available IAU schemas. The upgrade assistant will not detect the corresponding IAU schema from the provided domain directory automatically. Hence, you must select it manually. Once the IAU schema is upgraded, run the Upgrade Assistant again to upgrade the remaining schemas using the **All Schema Used by a domain** option on the Selected Schemas screen.
> - If there is no Audit schema (IAU) in your pre-upgrade environment, use the **All Schema Used by a Domain** option on the Selected Schemas screen and proceed.
> - To check whether the pre-upgrade environment has the IAU schema, run the following SQL command using the user with sysdba privileges:
>
>   ```
>   select username from dba_users where username like '%IAU%';
>   ```
>
>   This command lists the IAU schemas available in your configured database.

To upgrade product schemas with the Upgrade Assistant:

1. On the Welcome screen, review an introduction to the Upgrade Assistant and information about important pre-upgrade tasks. Click **Next**.

   > **Note:**
   >
   > For more information about any Upgrade Assistant screen, click **Help** on the screen.

2. On the Selected Schemas screen, select the schema upgrade operation that you want to perform:
   - **Individually Selected Schemas** if you want to select individual schemas for upgrade and you do not want to upgrade all of the schemas used by the domain.

     > **Caution:**
     >
     > Upgrade only those schemas that are used to support your 12*c* (12.2.1.3.0) components. Do not upgrade schemas that are currently being used to support components that are not included in Oracle Fusion Middleware 12*c* (12.2.1.3.0).

   - **All Schemas Used by a Domain** to allow the Upgrade Assistant to discover and select all components that have a schema available to upgrade in the domain specified in the **Domain Directory** field. This is also known as a *domain assisted schema upgrade*. Additionally, the Upgrade Assistant pre-populates connection information on the schema input screens.

> **Note:**
>
> Oracle recommends that you select **All Schemas Used by a Domain** for most upgrades to ensure all of the required schemas are included in the upgrade.

> **Note:**
>
> If your 11g domain contains Oracle Identity Navigator, choose **Individually Selected Schemas** and select only the Oracle Access Manager (OAM) and the OAM-related schemas.
>
> Do *not* select Oracle Identity Navigator (OIN) and OIN-related schemas, as Oracle Identity Navigator is not supported in 12c.

Click **Next**.

3. If you selected **Individually Selected Schemas**: On the Available Components screen, select the components for which you want to upgrade schemas. When you select a component, the schemas and any dependencies are automatically selected.

   If you selected **All schemas used by a domain**: On the Create Schema screen, enter the necessary Database details. This retrieves all of the schemas in the domain.

   Click **Next**.

4. On the Prerequisites screen, acknowledge that the prerequisites have been met by selecting all the check boxes. Click **Next**.

> **Note:**
>
> The Upgrade Assistant does not verify whether the prerequisites have been met.

5. On the Schema Credentials screen(s), specify the database connection details for each schema you are upgrading (the screen name changes based on the schema selected):
   - Select the database type from the **Database Type** drop-down menu.
   - Enter the database connection details, and click **Connect**.
   - Select the schema you want to upgrade from the **Schema User Name** drop-down menu, and then enter the password for the schema. Be sure to use the correct schema prefix for the schemas you are upgrading.

> **Note:**
>
> The component ID or schema name is changed for `UCSUMS` schema as of release 12.1.2, which means the Upgrade Assistant does not automatically recognize the possible schemas and display them in a drop-down list. You must manually enter the name in a text field. The name can be either *`prefix`*`_ORASDPM` or *`prefix`*`_UMS`, depending on the starting point for the upgrade.
> The `UCSUMS` schema is not auto-populated. Enter *`prefix`*`_ORASDPM` as the user. The upgrade environment uses `_ORASDPM` as the schema name, whereas in the 12*c* environment it is referred to as `_UMS`.

6. On the Examine screen, review the status of the Upgrade Assistant as it examines each schema, verifying that the schema is ready for upgrade. If the status is **Examine finished**, click **Next**.

   If the examine phase fails, Oracle recommends that you cancel the upgrade by clicking **No** in the Examination Failure dialog. Click **View Log** to see what caused the error and refer to Troubleshooting Your Upgrade in *Upgrading with the Upgrade Assistant* for information on resolving common upgrade errors.

   > **Note:**
   >
   > - If you resolve any issues detected during the examine phase without proceeding with the upgrade, you can start the Upgrade Assistant again without restoring from backup. However, if you proceed by clicking **Yes** in the Examination Failure dialog box, you need to restore your pre-upgrade environment from backup before starting the Upgrade Assistant again.
   > - Canceling the examination process has no effect on the schemas or configuration data; the only consequence is that the information the Upgrade Assistant has collected must be collected again in a future upgrade session.

7. On the Upgrade Summary screen, review the summary of the options you have selected for schema upgrade.

   Verify that the correct Source and Target Versions are listed for each schema you intend to upgrade.

   If you want to save these options to a response file to run the Upgrade Assistant again later in response (or silent) mode, click **Save Response File** and provide the location and name of the response file. A silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again.

   Click **Upgrade** to start the upgrade process.

8. On the Upgrade Progress screen, monitor the status of the upgrade.

> **Caution:**
>
> Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

If any schemas are not upgraded successfully, refer to the Upgrade Assistant log files for more information.

> **Note:**
>
> The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

Click **Next**.

9. If the upgrade is successful: On the Upgrade Success screen, click **Close** to complete the upgrade and close the wizard.

   If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at *ORACLE_HOME*/oracle_common/upgrade/logs.

> **Note:**
>
> If the upgrade fails, you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

## Verifying the Schema Upgrade

After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

If you are using an Oracle database, connect to the database as a user having Oracle DBA privileges, and run the following from SQL*Plus to get the current version numbers:

```
SET LINE 120
COLUMN MRC_NAME FORMAT A14
COLUMN COMP_ID FORMAT A20
COLUMN VERSION FORMAT A12
COLUMN STATUS FORMAT A9
COLUMN UPGRADED FORMAT A8
SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM
SCHEMA_VERSION_REGISTRY ORDER BY MRC_NAME, COMP_ID ;
```

In the query result:

- Check that the number in the `VERSION` column matches the latest version number for that schema. For example, verify that the schema version number is 12.2.1.3.0.

> **Note:**
>
> However, that not all schema versions will be updated. Some schemas do not require an upgrade to this release and will retain their pre-upgrade version number.

- The `STATUS` field will be either `UPGRADING` or `UPGRADED` during the schema patching operation, and will become `VALID` when the operation is completed.
- If the status appears as `INVALID`, the schema update failed. You should examine the logs files to determine the reason for the failure.
- Synonym objects owned by `IAU_APPEND` and `IAU_VIEWER` will appear as `INVALID`, but that does not indicate a failure.

  They become invalid because the target object changes after the creation of the synonym. The synonyms objects will become valid when they are accessed. You can safely ignore these `INVALID` objects.

> **Note:**
>
> Undo any non-SSL port changes and any non-SYSDBA user that you made when preparing for the upgrade.

# About Reconfiguring the Domain

Run the Reconfiguration Wizard to reconfigure your domain component configurations to 12*c* (12.2.1.3.0).

When you reconfigure a WebLogic Server domain, the following items are automatically updated, depending on the applications in the domain:

- WebLogic Server core infrastructure
- Domain version

> **Note:**
>
> Before you begin the domain reconfiguration, note the following limitations:
>
> - The Reconfiguration Wizard does not update any of your own applications that are included in the domain.
> - Transforming a non-dynamic cluster domain to a dynamic cluster domain during the upgrade process is not supported.
>
>  The dynamic cluster feature is available when running the Reconfiguration Wizard, but Oracle only supports upgrading a non-dynamic cluster upgrade and then adding dynamic clusters. You cannot add dynamic cluster during the upgrade process.

Specifically, when you reconfigure a domain, the following occurs:

- The domain version number in the `config.xml` file for the domain is updated to the Administration Server's installed WebLogic Server version.
- Reconfiguration templates for all installed Oracle products are automatically selected and applied to the domain. These templates define any reconfiguration tasks that are required to make the WebLogic domain compatible with the current WebLogic Server version.
- Start scripts are updated.

  If you want to preserve your modified start scripts, be sure to back them up before starting the Reconfiguration Wizard.

> **Note:**
>
> When the domain reconfiguration process starts, you can't undo the changes that it makes. Before running the Reconfiguration Wizard, ensure that you have backed up the domain as covered in the pre-upgrade checklist. If an error or other interruption occurs while running the Reconfiguration Wizard, you must restore the domain by copying the files and directories from the backup location to the original domain directory. This is the only way to ensure that the domain has been returned to its original state before reconfiguration.

Follow these instructions to reconfigure the existing domain using the Reconfiguration Wizard. See Reconfiguring WebLogic Domains in *Upgrading Oracle WebLogic Server*.

- Backing Up the Domain
- Starting the Reconfiguration Wizard
- Reconfiguring the Oracle Access Manager Domain
  Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing 11g domain.

## Backing Up the Domain

Before running the Reconfiguration Wizard, create a backup copy of the domain directory.

To create a backup of the Administration server domain directory:

1. Copy the source domain to a separate location to preserve the contents.

   (Windows) `copy /home/Oracle/config/domains to /home/Oracle/config/domains_backup`.

   (UNIX) `cp -rf domains domains_backup`

2. For HA environments, before updating the domain on each remote Managed Server, create a backup copy of the domain directory on each remote machine.
3. Verify that the backed up versions of the domain are complete.

If domain reconfiguration fails for any reason, you must restore all files and directories from the backup directory into the original domain directory to ensure that the domain is returned entirely to its original state before reconfiguration.

# Starting the Reconfiguration Wizard

> **Note:**
>
> Shut down the administration server and all collocated managed servers before starting the reconfiguration process. See Stopping Servers and Processes.

To start the Reconfiguration Wizard in graphical mode:

1. Open the command shell (on UNIX operating systems) or open a command prompt window (on Windows operating systems).
2. **Edition Based Database Users Only**: If your schemas are configured with EBR database, a default edition name must be manually supplied before you run the Reconfiguration Wizard.

   Run the following SQL command to set the default edition:
   `ALTER DATABASE DEFAULT EDITION = edition_name;`

   where *edition_name* is the child edition name.
3. Go to the `oracle_common/common/bin` directory:
   - (UNIX) *ORACLE_HOME*`/oracle_common/common/bin`
   - (Windows) *ORACLE_HOME*`\oracle_common\commom\bin`

   Where, *ORACLE_HOME* is the 12c Oracle Home.
4. Start the Reconfiguration Wizard:

   The `./reconfig.sh` command, might display the following error to indicate that the default cache directory is not valid:

   ```
   *sys-package-mgr*: can't create package cache dir
   ```

   So, first, change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`.

   For example: `CONFIG_JVM_ARGS=-Dpython.cachedir=`*valid_directory*

   Start the Reconfiguration Wizard with the following logging options:
   - (UNIX) `./reconfig.sh -log=`*log_file* `-log_priority=ALL`
   - (Windows) `reconfig.cmd -log=`*log_file* `-log_priority=ALL`

   where *log_file* is the absolute path of the log file you'd like to create for the domain reconfiguration session. This can be helpful if you need to troubleshoot the reconfiguration process.

   The parameter `-log_priority=ALL` ensures that logs are logged in fine mode.

# Reconfiguring the Oracle Access Manager Domain

Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing 11g domain.

> **Note:**
>
> If the source is a clustered environment, run the Reconfiguration Wizard on the primary node only. Where, primary node is the Administration Server. Use the pack/unpack utility to apply the changes to other cluster members in the domain.

To reconfigure the domain with the Reconfiguration Wizard:

1. On the Select Domain screen, specify the location of the domain you want to upgrade or click **Browse** to navigate and select the domain directory. Click **Next**.
2. On the Reconfiguration Setup Progress screen, view the progress of the setup process. When complete, click **Next**.

   During this process:

   - The reconfiguration templates for your installed products, including Fusion Middleware products, are automatically applied. This updates various domain configuration files such as `config.xml`, `config-groups.xml`, and `security.xml` (among others).
   - Schemas, scripts, and other such files that support your Fusion Middleware products are updated.
   - The domain upgrade is validated.
3. On the Domain Mode and JDK screen, select the JDK to use in the domain or click **Browse** to navigate to the JDK you want to use. The supported JDK version for 12*c* (12.2.1.3.0) is 1.8.0_131 and later. Click **Next**.

   > **Note:**
   >
   > You cannot change the **Domain Mode** at this stage.

   For a list of JDKs that are supported for a specific platform, see Oracle Fusion Middleware Supported System Configurations.
4. On the Database Configuration Type screen, select **RCU Data** to connect to the Server Table (_STB) schema.

   Enter the database connection details using the RCU service table (_STB) schema credentials and click **Get RCU Configuration**.

   The Reconfiguration Wizard uses this connection to automatically configure the data sources required for components in your domain.

> **Note:**
>
> By default **Oracle's Driver (Thin) for Service connections; Versions: Any** is the selected driver. If you specified an instance name in your connection details — instead of the service name — you must select **Oracle's Driver (Thin) for pooled instance connections; Versions: Any** If you do not change the driver type, then the connection will fail.
> For information about selecting grid link for RAC databases in HA environments, see Access Manager High Availability Architecture.

> **Note:**
>
> For any existing 11*g* datasource, the reconfiguration will preserve the existing values. For new datasources where the schema was created for 12*c* by the RCU, the default connection data will be retrieved from the _STB schema. If no connection data for a given schema is found in the _STB schema, then the default connection data is used.

If the check is successful, click **Next**. If the check fails, reenter the connection details correctly and try again.

> **Note:**
>
> If your database has _OPSS or _IAU 11*g* database schemas, you must manually enter database connection details for those schemas. These schemas were not required in 11*g* and had to be created manually. Users could assign any name to these schemas, therefore the Reconfiguration Wizard does not recognize them. When providing connection information for _IAU, use the IAU_APPEND user information.

5. On the JDBC Component Schema screen, verify that the DBMS/Service and the Host name is correct for the following component schemas:
   - OPSS Audit schema
   - OPSS Audit viewer schema
   - OPSS schema

   If you are connecting to a RAC database, select each of the schemas you want to update and click **Convert to Grid Link**. Click **Next** to update the Service Name, Schema Password, SCAN, Hostname/Port, ONS Host/Port.

   Click **Next**.
6. On the JDBC Component Schema Test screen, select all the component schemas and click **Test Selected Connections** to test the connection for each schema. The result of the test is indicated in the Status column.

   When the check is complete, click **Next**.
7. On the Node Manager screen, select the appropriate Node Manager Type based on your requirements, specify the details, and click **Next**.

> **Note:**
>
> There are two types of node managers. It is recommend to use the domain-based node manager, so that, you can have different versions of the node manager for each domain.

8. On the Advanced Configuration screen, select **Administration Server**, **Topology**, and **Deployments and Services**. Select **Domain Frontend Host Capture** if required.

   For each of the categories you select, the appropriate configuration screen is displayed to allow you to perform advanced configuration.

   > **Note:**
   >
   > Ensure that you assign oam_server1 or the OAM managed server name used to the server group **OAM-MDG-SVRS**, and `oam_policy_mgr1` to the server group **OAM-POLICY-MANAGED-SERVER**.

9. On the Configuration Summary screen, review the detailed configuration settings of the domain before continuing.

   You can limit the items that are displayed in the right-most panel by selecting a filter option from the **View** drop-down list.

   To change the configuration, click **Back** to return to the appropriate screen. To reconfigure the domain, click **Reconfig**.

   > **Note:**
   >
   > The location of the domain does not change when you reconfigure it.

10. The Reconfiguration Progress screen displays the progress of the reconfiguration process.

    During this process:

    - Domain information is extracted, saved, and updated.
    - Schemas, scripts, and other such files that support your Fusion Middleware products are updated.

    When the progress bar shows 100%, click **Next**.

11. The End of Configuration screen indicates whether the reconfiguration process completed successfully or failed. It also displays the location of the domain that was reconfigured as well as the Administration Server URL (including the listen port). If the reconfiguration is successful, it displays **Oracle WebLogic Server Reconfiguration Succeeded**.

    If the reconfiguration process did not complete successfully, an error message is displayed indicates the reason. Take appropriate action to resolve the issue. If you cannot resolve the issue, contact My Oracle Support.

    Note the Domain Location and the Admin Server URL for further operations.

# Upgrading Domain Component Configurations

After reconfiguring the domain, use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.

- Starting the Upgrade Assistant
  Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.
- Upgrading Oracle Access Manager Domain Component Configurations
  Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.
- Removing the Oracle Mobile Security Manager Servers Footprint
  This activity applies only to Oracle Access Manager 11g (OAM 11.1.23.x) environments that used the Oracle Mobile Security Manager (OMSM) application and have been upgraded to Oracle Access Manager 12c (12.2.1.3.0).
- Starting Servers and Processes
  After a successful upgrade, start all processes and servers, including the Administration Server and any Managed Servers.
- Verifying the Domain-Specific-Component Configurations Upgrade
  To verify that the domain-specific-component configurations upgrade was successful, sign in to the Administration console and the Oracle Enterprise Manager Fusion Middleware Control and verify that the version numbers for each component is 12.2.1.3.0.

## Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.

To start the Upgrade Assistant:

> **Note:**
>
> Before you start the Upgrade Assistant, make sure that the JVM character encoding is set to UTF-8 for the platform on which the Upgrade Assistant is running. If the character encoding is not set to UTF-8, then you will not be able to download files containing Unicode characters in their names. This can cause the upgrade to fail.
>
> To ensure that UTF-8 is used by the JVM, use the JVM option `-Dfile.encoding=UTF-8`.

1. Go to the `oracle_common/upgrade/bin` directory:
   - (UNIX) *`ORACLE_HOME`*`/oracle_common/upgrade/bin`
   - (Windows) *`ORACLE_HOME`*`\oracle_common\upgrade\bin`

2. Start the Upgrade Assistant:
   - (UNIX) `./ua`
   - (Windows) `ua.bat`

> **Note:**
>
> In the above command, *ORACLE_HOME* refers to the 12*c* (12.2.1.3.0) Oracle Home.

For information about other parameters that you can specify on the command line, such as logging parameters, see:

- Upgrade Assistant Parameters

## Upgrade Assistant Parameters

When you start the Upgrade Assistant from the command line, you can specify additional parameters.

**Table 3-9 Upgrade Assistant Command-Line Parameters**

| Parameter | Required or Optional | Description |
|---|---|---|
| `-readiness` | Required for readiness checks<br>**Note**: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server). | Performs the upgrade readiness check without performing an actual upgrade.<br>Schemas and configurations are checked.<br>Do not use this parameter if you have specified the `-examine` parameter. |
| `-threads` | Optional | Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas.<br>The value must be a positive integer in the range 1 to 8. The default is 4. |
| `-response` | Required for silent upgrades or silent readiness checks | Runs the Upgrade Assistant using inputs saved to a response file generated from the data that is entered when the Upgrade Assistant is run in GUI mode. Using this parameter runs the Upgrade Assistant in *silent mode* (without displaying Upgrade Assistant screens). |
| `-examine` | Optional | Performs the examine phase but does not perform an actual upgrade.<br>Do not specify this parameter if you have specified the `-readiness` parameter. |

**Table 3-9 (Cont.) Upgrade Assistant Command-Line Parameters**

| Parameter | Required or Optional | Description |
|---|---|---|
| `-logLevel` *attribute* | Optional | Sets the logging level, specifying one of the following attributes:<br>• `TRACE`<br>• `NOTIFICATION`<br>• `WARNING`<br>• `ERROR`<br>• `INCIDENT_ERROR`<br>The default logging level is `NOTIFICATION`.<br>Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used. |
| `-logDir` *location* | Optional | Sets the default location of upgrade log files and temporary files. You must specify an existing, writable directory where the Upgrade Assistant creates log files and temporary files.<br>The default locations are:<br>(UNIX)<br>*ORACLE_HOME*`/oracle_common/upgrade/logs`<br>*ORACLE_HOME*`/oracle_common/upgrade/temp`<br>(Windows)<br>*ORACLE_HOME*`\oracle_common\upgrade\logs`<br>*ORACLE_HOME*`\oracle_common\upgrade\temp` |
| `-help` | Optional | Displays all of the command-line options. |

# Upgrading Oracle Access Manager Domain Component Configurations

Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.

After running the Reconfiguration Wizard to reconfigure the WebLogic domain to 12*c* (12.2.1.3.0), you must run the Upgrade Assistant to upgrade the domain *component* configurations to match the updated domain configuration.

To upgrade domain component configurations with the Upgrade Assistant:

1. On the Welcome screen, review an introduction to the Upgrade Assistant and information about important pre-upgrade tasks. Click **Next**.

   > **Note:**
   >
   > For more information about any Upgrade Assistant screen, click **Help** on the screen.

2. On the next screen:
   - Select **All Configurations Used By a Domain**. The screen name changes to WebLogic Components.
   - In the **Domain Directory** field, enter the 11.1.2.3.0 domain directory path.

   Click **Next**.

3. On the Component List screen, verify that the list includes all the components for which you want to upgrade configurations and click **Next**.

   If you do not see the components you want to upgrade, click **Back** to go to the previous screen and specify a different domain.

4. On the Prerequisites screen, acknowledge that the prerequisites have been met by selecting all the check boxes. Click **Next**.

   > **Note:**
   >
   > The Upgrade Assistant does not verify whether the prerequisites have been met.

5. On the Examine screen, review the status of the Upgrade Assistant as it examines each component, verifying that the component configuration is ready for upgrade. If the status is **Examine finished**, click **Next**.

   If the examine phase fails, Oracle recommends that you cancel the upgrade by clicking **No** in the Examination Failure dialog. Click **View Log** to see what caused the error and refer to Troubleshooting Your Upgrade in *Upgrading with the Upgrade Assistant* for information on resolving common upgrade errors.

> **Note:**
>
> - If you resolve any issues detected during the examine phase without proceeding with the upgrade, you can start the Upgrade Assistant again without restoring from backup. However, if you proceed by clicking **Yes** in the Examination Failure dialog box, you need to restore your pre-upgrade environment from backup before starting the Upgrade Assistant again.
> - Canceling the examination process has no effect on the configuration data; the only consequence is that the information the Upgrade Assistant has collected must be collected again in a future upgrade session.

6. On the Upgrade Summary screen, review the summary of the options you have selected for component configuration upgrade.

   The response file collects and stores all the information that you have entered, and enables you to perform a silent upgrade at a later time. The silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again. If you want to save these options to a response file, click **Save Response File** and provide the location and name of the response file.

   Click **Upgrade** to start the upgrade process.

7. On the Upgrade Progress screen, monitor the status of the upgrade.

   > **Caution:**
   >
   > Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

   If any components are not upgraded successfully, refer to the Upgrade Assistant log files for more information.
   Upgrade Assistant log files location:

   - (UNIX) *ORACLE_HOME*/oracle_common/upgrade/logs/ua<*timestamp*>.log
   - (Windows) *ORACLE_HOME*\oracle_common\upgrade\logs\ua<*timestamp*>.log

   > **Note:**
   >
   > The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

   Click **Next**.

8. If the upgrade is successful: On the Upgrade Success screen, click **Close** to complete the upgrade and close the wizard. The Post-Upgrade Actions window describes the manual tasks you must perform to make components functional in the new installation. This window appears only if a component has post-upgrade steps.

If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at *ORACLE_HOME*/oracle_common/upgrade/logs.

> **Note:**
>
> If the upgrade fails you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

# Removing the Oracle Mobile Security Manager Servers Footprint

This activity applies only to Oracle Access Manager 11g (OAM 11.1.23.x) environments that used the Oracle Mobile Security Manager (OMSM) application and have been upgraded to Oracle Access Manager 12c (12.2.1.3.0).

Oracle Mobile Security Manager (OMSM) application is not supported in OAM 12c (12.2.1.3.0). Therefore, Oracle recommends you to remove all components of OMSM. Removing all components will avoid any potential issues if the WebLogic Server Managed Server that runs the OMSM application gets started accidentally.

You have to remove the OMSM components from the following areas:

- The WebLogic Server Managed Server(s)
- The product's directory structure
- The database schema

- Removing the WebLogic Server OMSM Managed Server(s) From the Domain
- Removing the WebLogic Server OMSM Managed Server(s) From the Directory Structure
- Removing the OMSM Server Schema Objects From the Database

## Removing the WebLogic Server OMSM Managed Server(s) From the Domain

To remove the WebLogic Server OMSM Managed Server(s) from the domain:

1. Ensure that only the WebLogic Server Administration Server is running.
2. Access and log in to the WebLogic Server Console.
3. Click **Environment**, select **Clusters**, and then **Coherence Cluster**.
4. Select the cluster name that contains the names of members that include the OMSM server(s).
5. Click the **Members** tab, uncheck the OMSM server(s), and click **Save**.
6. Click **Environment** and select **Servers**.
7. Select the OMSM server(s) (check it), and click **Delete**.
8. Depending on the WebLogic Server type, Production or Development, perform the additional steps to activate the changes.

   The WebLogic Server Managed OMSM Server(s) is now no longer present.
9. Log out of the WebLogic Server Administration application.

## Removing the WebLogic Server OMSM Managed Server(s) From the Directory Structure

To remove the WebLogic Server OMSM Managed Server(s) from the directory structure:

1. Verify that the WebLogic Server Administration Server and Managed Servers are stopped.
2. From a terminal prompt, navigate to the *DOMAIN_HOME*/servers location.
3. Run the `ls` command.

   The list of server names is displayed. Make a note of the name of the OMSM Server(s).

   For example:

   ```
   wls_msm1, wls_msm2, and so on.
   ```

4. Run the following command to remove the OMSM Server(s):

   ```
   rm -rf MSM_Server
   ```

   In the above command, `MSM_Server` is the name of the Oracle Mobile Security Manager (OMSM) server. For example:

   ```
   rm -rf wls_msm1
   ```

   Repeat this step as needed for any additional OMSM Server(s).

   Now the OMSM Server(s) directory structure is no longer present:

   ```
   m wls_msm1
   ```

## Removing the OMSM Server Schema Objects From the Database

To remove the Oracle Mobile Security Manager (MSM) schema objects:

1. Verify that the WebLogic Server Administration Server and Managed Servers are stopped.
2. Using your preferred tool, connect to the database system schema and run the following query:

   ```
   SET LINE 120
   COLUMN MRC_NAME FORMAT A14
   COLUMN COMP_ID FORMAT A20
   COLUMN VERSION FORMAT A12
   COLUMN STATUS FORMAT A9
   COLUMN UPGRADED FORMAT A8
   SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM
   SCHEMA_VERSION_REGISTRY ORDER BY MRC_NAME, COMP_ID;
   ```

   The query result shows `COMP_ID` as `OMSM`. Note the `OWNER`.

3. Use the 11.1.1.9.0 Repository Creation Utility (RCU) to drop the OMSM schema.
   a. Run the following command to start the RCU application:

      ```
      /rcu
      ```

   b. On the Welcome screen, click **Next**.
   c. On the Create Repository screen, select **Drop Repository**, and click **Next**.
   d. Specify the database connection credentials, as described in the following table:

**Table 3-10 Database Connection Details**

| Option | Description and Example |
|---|---|
| **Host Name** | Specify the name of the server where your database is running, in the following format:<br>`<FQDN>`<br>For Oracle RAC databases, specify the VIP name or the name of one of the nodes. |
| **Port** | Specify the port number for your database. The default port number for Oracle databases is `1521`. |
| **Service Name** | Specify the service name for the database. Typically, the service name is same as the global database name.<br>For Oracle RAC databases, specify the service name of one of the nodes. For example: `<INSTANCE_NAME_FQDN>` |
| **Username** | Specify the user name for your database. The default user name is `SYS`. |
| **Password** | Specify the password for the database user. |
| **Role** | Select the database user's role from the drop-down list: **Normal** or **SYSDBA**. |

   Click **Next**.

   A separate dialog window appears while RCU checks connectivity and the database prerequisites. When the database checking passes without errors, click **OK** to dismiss the dialog window and go to the next screen.

   e. On the Summary screen, review the information and click **Drop** to drop the schemas.
   f. On the Completion Summary screen, note the location of the log files and click **Close** to dismiss the screen.

4. Repeat Step 1 and run the following query:

```
SET LINE 120
COLUMN MRC_NAME FORMAT A14
COLUMN COMP_ID FORMAT A20
COLUMN VERSION FORMAT A12
COLUMN STATUS FORMAT A9
COLUMN UPGRADED FORMAT A8
SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM
SCHEMA_VERSION_REGISTRY ORDER BY MRC_NAME, COMP_ID;
```

The `COMP_ID OMSM` is now not available in the query result.

5. Start the WebLogic Server Administration server and the Managed servers.

## Starting Servers and Processes

After a successful upgrade, start all processes and servers, including the Administration Server and any Managed Servers.

The components may be dependent on each other so they must be started in the correct order.

> **Note:**
>
> The procedures in this section describe how to start servers and process using the WLST command line or a script. You can also use the Oracle Fusion Middleware Control and the Oracle WebLogic Server Administration Console. See Starting and Stopping Administration and Managed Servers and Node Manager in *Administering Oracle Fusion Middleware.*

To start your Fusion Middleware environment, follow the steps below.

**Step 1: Start Node Manager**

Starting the Node Manager in the Administration Server domain home:

- (UNIX) `nohup ./startNodeManager.sh > DOMAIN_HOME/nodemanager/nodemanager.out 2>&1 &`
- (Windows) `nohup .\startNodeManager.sh > DOMAIN_HOME\nodemanager\nodemanager.out 2>&1 &`

**Step 2: Start the Administration Server**

When you start the Administration Server, you also start the processes running in the Administration Server, including the WebLogic Server Administration Console and Fusion Middleware Control.

To start the Administration Server, use the `startWebLogic` script:

- (UNIX) `DOMAIN_HOME/bin/startWebLogic.sh`
- (Windows) `DOMAIN_HOME\bin\startWebLogic.cmd`

When prompted, enter your user name, password, and the URL of the Administration Server.

**Step 3: Start the Managed Servers**

**Method 1:** Start a WebLogic Server Managed Server by using the Weblogic Console:

- Log into Weblogic console as a `weblogic` Admin.
- Go to **Servers** > **Control** tab.
- Select the required managed server.
- Click **Start**.

**Method 2:** Start a WebLogic Server Managed Server by using the `startManagedWebLogic` script:

- (UNIX) *DOMAIN_HOME*`/bin/startManagedWebLogic.sh` *managed_server_name admin_url*
- (Windows) *DOMAIN_HOME*`\bin\startManagedWebLogic.cmd` *managed_server_name admin_url*

When prompted, enter your user name and password.

> **Note:**
>
> - The startup of a Managed Server will typically start the applications that are deployed to it. Therefore, it should not be necessary to manually start applications after the Managed Server startup.
> - The Mobile Security Manager (MSM) servers are not supported in 12c. After restarting the servers, the 11*g* configurations of MSM servers, like `omsm_server1` or `WLS_MSM1`, might remain. Ignore these configurations and do not restart the MSM servers.

## Verifying the Domain-Specific-Component Configurations Upgrade

To verify that the domain-specific-component configurations upgrade was successful, sign in to the Administration console and the Oracle Enterprise Manager Fusion Middleware Control and verify that the version numbers for each component is 12.2.1.3.0.

To sign in to the Administration Console, go to: `http://`*administration_server_host*`:`*administration_server_port*`/console`

To sign in to the Administration Console in an EDG deployment, see Validating the Virtual Server Configuration and Access to the Consoles.

To sign in to Oracle Enterprise Manager Fusion Middleware Control Console, go to: `http://`*administration_server_host*`:`*administration_server_port*`/em`

> **Note:**
>
> - After upgrade, ensure you run the administration tools from the new 12*c* Oracle home directory and not from the previous Oracle home directory.
> - During the upgrade process, some OWSM documents, including policy sets and predefined documents such as policies and assertion templates, may need to be upgraded. If a policy set or a predefined document is upgraded, its version number is incremented by 1.
> - In the site-specific configuration, the WebLogic and EM consoles must be accessible with the URLs either directly or through proxy URLs.

# Performing Post-Upgrade Tasks

After performing the upgrade of Oracle Access Manager to 12c (12.2.1.3), you should complete the tasks summarized in this section, if required.

This section includes the following tasks:

- WebGates Configuration Fails during Authentication
  WebGates configured with the `hmacEnabled=true` in environments where `globalHMACEnabled` is not set to `true` fails during authentication.
- Updating the java.security File
  If you have multiple components of Oracle Identity and Access Management (Oracle Access Manager, Oracle Identity Manager, WebGates and so on) deployed, until you upgrade all of the components to 12*c* (12.2.1.3.0), you must update the `java.security` file with the changes described in this section.

## WebGates Configuration Fails during Authentication

WebGates configured with the `hmacEnabled=true` in environments where `globalHMACEnabled` is not set to `true` fails during authentication.

To solve this issue, apply patch 12.2.1.3.181016 or later.
For more information, see Upgrading to OHS/OTD 12c WebGate.

## Updating the java.security File

If you have multiple components of Oracle Identity and Access Management (Oracle Access Manager, Oracle Identity Manager, WebGates and so on) deployed, until you upgrade all of the components to 12*c* (12.2.1.3.0), you must update the `java.security` file with the changes described in this section.

To do this:

1. Open the `java.security` file located at *`JAVA_HOME`*`/jre/lib/security/` in an editor.
2. Remove `TLSv1`, `TLSv1.1`, `MD5withRSA` from the following key:

   ```
   key - jdk.tls.disabledAlgorithms
   ```
3. Remove `MD5` from the following key:

   ```
   key - jdk.certpath.disabledAlgorithms
   ```

For more information on possible upgrade scenarios, see Troubleshooting Security Policy Issues When Upgrading.

# Performing the Post-Patch Install Steps

After completing the upgrade, you have to perform the post-patch installation steps.

The post-patch installation steps comprises the following:

- Running the Poststart Command to Confirm Successful Binary Patching
- Performing a Clean Restart of the Servers

## Running the Poststart Command to Confirm Successful Binary Patching

Use the variables and the instructions in the Stack Patch Bundle README.txt file to run the `poststart` command for your product, as shown below:

```
$ ./spbat.sh -type oig -phase poststart -mw_home /
<INSTALLATION_DIRECTORY>/IAM12c -spb_download_dir /<DOWNLOAD_LOCATION>/
IDM_SPB_12.2.1.4.200714 -log_dir /<DOWNLOAD_LOCATION>/OIGlogs
```

For details, see Doc ID 2657920.1.

## Performing a Clean Restart of the Servers

Restart all the servers including the Administration Server and any Managed Servers. See Starting Servers and Processes .

4

# Upgrading Oracle Access Manager Highly Available Environments

Describes the process of upgrading an Oracle Access Manager highly available environments from 11*g* Release 2 (11.1.2.3.0) to 12*c* (12.2.1.3.0).

**Topics**

- Performing the Post-Patch Install Steps
  After completing the upgrade, you have to perform the post-patch installation steps.

# About the Oracle Access Manager Multinode Upgrade Process

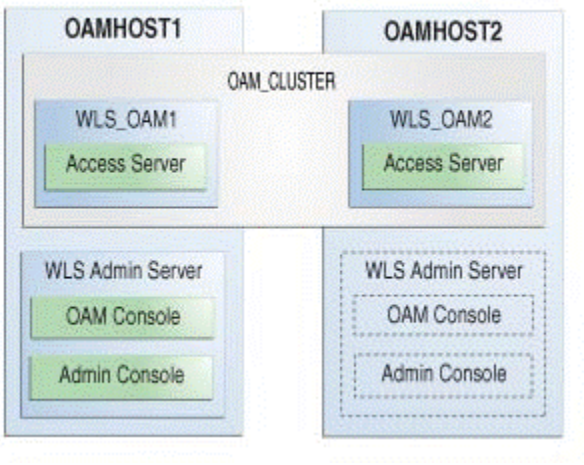
Review the topology and the roadmap for an overview of the upgrade process for Oracle Access Manager highly available environments.

The steps you take to upgrade your existing domain will vary depending on how your domain is configured and which components are being upgraded. Follow only those steps that are applicable to your deployment.

### Upgrade Topology

The following topology shows the Oracle Access Manager cluster set up that can be upgraded to 12*c* (12.2.1.3.0) by following the procedure described in this chapter.

**Figure 4-1 Oracle Access Manager High Availability Upgrade Topology**




On OAMHOST1, the following installations have been performed:

- An Oracle Access Management Access Manager instance has been installed in the WLS_OAM1 Managed Server.
- A WebLogic Server Administration Server has been installed. Under normal operations, this is the active Administration Server.

On OAMHOST2, the following installations have been performed:

- An Oracle Access Management Access Manager instance has been installed in the WLS_OAM2 Managed Server.
- A WebLogic Server Administration Server has been installed. Under normal operations, this is the passive Administration Server. You make this Administration Server active if the Administration Server on OAMHOST1 becomes unavailable.

The instances in the WLS_OAM1 and WLS_OAM2 Managed Servers on OAMHOST1 and OAMHOST2 are configured in a cluster named OAM_CLUSTER.

**Table 4-1 Tasks for Upgrading Oracle Access Manager Highly Available Environments**

| Task | Description |
|---|---|
| **Required**<br>If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks. | See:<br>• Introduction to Upgrading Oracle Access Manager to 12*c* (12.2.1.3.0)<br>• Pre-Upgrade Requirements |
| **Required**<br>Complete the necessary pre-upgrade tasks specific to Oracle Access Manager. | See Completing the Pre-Upgrade Tasks for Oracle Access Manager. |
| **Required**<br>Create the 12*c* Oracle Home Folder on both OAMHOST1 and OAMHOST2, so that you can use the location for installing the product distributions. | See Creating 12c Oracle Home Folder on OAMHOST1 and OAMHOST2. |
| **Required**<br>Install Oracle Access Manager 12*c* (12.2.1.3.0) in the new Oracle home. | See Installing Product Distributions on OAMHOST1 and OAMHOST2. |
| **Required**<br>Apply the latest bundle patches | See Installing the Latest Stack Patch Bundle. |
| **Required**<br>Upgrade the necessary schemas on OAMHOST1. | See Upgrading Schemas on OAMHOST1. |
| **Required**<br>Reconfigure the Oracle Access Manager domain on OAMHOST1. | See Reconfiguring the Domain on OAMHOST1. |
| **Required**<br>Replicate the Oracle Access Manager domain configurations on OAMHOST2. | This includes packing the domain on OAMHOST1 and unpacking it on OAMHOST2.<br>See Replicating the Domain Configurations on each OAMHOST. |
| **Required**<br>Upgrade the domain component configurations on both OAMHOST1 and OAMHOST2. | The Upgrade Assistant is used to update the reconfigured domain's component configurations.<br>See Upgrading Domain Component Configurations on OAMHOST1 and OAMHOST2. |
| **Required**<br>Start the servers on OAMHOST1 and OAMHSOT2. | See Starting the Servers. |
| **Required**<br>Complete any necessary post-upgrade tasks. | These tasks are optional. See Performing Post-Upgrade Tasks. |
| **Required**<br>Complete the post-patch install steps. | See Performing the Post-Patch Install Steps. |

# Completing the Pre-Upgrade Tasks for Oracle Access Manager

Complete the pre-upgrade tasks described in this section before you upgrade Oracle Access Manager.

- Checking the Supported Starting Point for Oracle Access Manager Upgrade
  The Oracle Access Manager version that is supported for upgrade is `11g Release 2 (11.1.2.3.0)`.
- Checking if OAM is in a Different Domain to OAAM and OIM
  In the case of Oracle Access Manager (OAM), Oracle Adaptive Access Management (OAAM), and Oracle Identity Manager (OIM) integrated setup, where OAM and OAAM are in same domain, and OIM is in a separate domain, the OAM domain needs to be cloned that works with OAAM and OIM in the source domain.
- Removing the IAMSuiteAgent Deployment
  The `IAMSuiteAgent` deployment is not supported in 12*c*. Therefore, undeploy the `IAMSuiteAgent` before you proceed with the upgrade.
- Upgrading Java JSE Policy
  Upgrade Java JSE Policy, if required.
- Disabling Deprecated Services in OAM
  Applies only to Mobile and Social, Security Token Service, Mobile Security Service, and MSAS proxy users.

# Checking the Supported Starting Point for Oracle Access Manager Upgrade

The Oracle Access Manager version that is supported for upgrade is `11g Release 2 (11.1.2.3.0)`.

If you are using an earlier version of Oracle Access Manager, you must upgrade to Oracle Access Manager `11g Release 2 (11.1.2.3.0)` first, and then to 12*c*.

# Checking if OAM is in a Different Domain to OAAM and OIM

In the case of Oracle Access Manager (OAM), Oracle Adaptive Access Management (OAAM), and Oracle Identity Manager (OIM) integrated setup, where OAM and OAAM are in same domain, and OIM is in a separate domain, the OAM domain needs to be cloned that works with OAAM and OIM in the source domain.

> **Note:**
>
> Ensure that Oracle Access Manager and Oracle Identity Manager are in different domains. If they are in the same domain, then you need to separate them into multiple domains. For more information, see Separating Oracle Identity Management Applications Into Multiple Domains.

To separate the OAM and OAAM domain, do the following:

1. Perform the test-to-production of the source environment (machine-1) where OAM and OAAM is in the same domain, so as to form the 11.1.2.3.0 OAM-OAAM environment on machine-2. This machine-2 acts as the production machine.
2. On machine-1, open the *`DOMAIN_HOME`*`/config/fmwconfig/oam-config.xml` file in a text editor, and search for the parameter `HOST_ALIAS_1`.

3. Update the `serverhost` parameter to reflect the name of production machine, so that it knows the target (OAAM) machine to which it has to point to render the OAAM authentication page.
4. Search for the parameter `Version`, and increment its value by one.
5. Restart only the Administration Server and the OAM Server of source machine (machine-1) to reflect the changes.

   Ensure that the oaam_admin_server1 and oaam_server_server1 on the source machine are stopped.
6. Start the oaam_admin_server1 and oaam_server_server1 on production machine (machine-2). The Administration Server on the production machine will be in `Running` state after the T2P.
7. Access the tapscheme protected resource of machine-1. Make sure that the request gets redirected to OAAM server of machine—2 and subsequent taspscheme login is successful.

> **Note:**
>
> Ensure that the date and time on source and production machine are in sync. If they are not, the authentication fails.

If OIM is installed in a separate domain, and is integrated with OAM and OAAM, do the following:

1. Update the following Oracle Identity Manager properties to contain the details of the new OAAM host:

   ```
   OIM.ChangePasswordURL
   OIM.ChallengeQuestionModificationURL
   ```

   For information about setting the Oracle Identity Manager properties for OAAM, see Setting Oracle Identity Manager Properties for Oracle Adaptive Access Manager in the *Integration Guide for Oracle Identity Management Suite* for 11*g* Release 2 (11.1.2.3.0).
2. Restart the Oracle Identity Manager server.

> **Note:**
>
> You must upgrade the OAM domain whose Managed Server is in the running state after the domain separation.
>
> For example, if you have followed the steps in this section, you will have to upgrade OAM that resides on machine-1, to 12*c*.

## Removing the IAMSuiteAgent Deployment

The `IAMSuiteAgent` deployment is not supported in 12*c*. Therefore, undeploy the `IAMSuiteAgent` before you proceed with the upgrade.

## Removing `IAMSuiteAgent` from the WebLogic Administration Console

1. Log in to the WebLogic Administration Console using the following URL:

   `http://hostname:port/console`

   where *`hostname`* is the DNS name or IP address of the Administration Server and *`port`* is the listen port on which the Administration Server is listening for requests (port 7001 by default). If you have configured a domain-wide administration port, use that port number. If you configured the Administration Server to use Secure Socket Layer (SSL) you must add *`s`* after `http` as follows:

   `https://hostname:port/console`

   > **Note:**
   >
   > A domain-wide administration port always uses SSL.

2. Click **Security Realms**.
3. Click **myrealm**.
4. Click **Provider**, and then select **IAMSuiteAgent**.
5. Click **Delete**.
6. Restart the servers.

## Removing `IAMSuiteAgent` from the OAM Console

> **Note:**
>
> Before you delete `IAMSuiteAgent` from the OAM console, complete the following tasks:
>
> - Replace `IAMSuiteAgent` with an 11g WebGate. See Replacing the IAMSuiteAgent with an 11g WebGate. Removing `IAMSuiteAgent` without replacing it with an 11g WebGate may result in a loss of the OAM functionalities in the 11g server.
> - Back up the OAM configuration.

1. Log in to the OAM console.
2. Go to the **Application Security** tab, click **Agents**, and then **Managed single sign-on agents**.
3. From the list of SSO agents, select `IAMSuiteAgent`, and then click **Delete**.
4. Confirm the deletion.

## Upgrading Java JSE Policy

Upgrade Java JSE Policy, if required.

> **Note:**
>
> This is required if any of the Identity Management components like Oracle Access Management (OAM), Oracle Identity Manager (OIM), Oracle Adaptive Access Manager (OAAM), or Oracle Access Manager Webgates of a data center are yet to be upgraded to 12*c* (12.2.1.3.0). This is for the phased transition to 12*c* (12.2.1.3.0).
>
> For a Multi Data Center setup, this is required if any of the data centers has 12*c* (12.2.1.2.0) components (OAM, OIM, OAAM, OAM Webgates).

The jar files `local_policy.jar` and `US_export_policy.jar` are present in the directory `$JAVA_HOME/jre/lib/security`. You can upgrade Java JSE policy by overwriting these jar files with the specified versions. To do this, complete the following steps:

1. Download the `local_policy.jar` and `US_export_policy.jar` files from the following location:

   http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html

2. Copy the jar files to the location `$JAVA_HOME/jre/lib/security`. This overwrites the existing files.

This completes the Java JSE policy upgrade.

## Disabling Deprecated Services in OAM

Applies only to Mobile and Social, Security Token Service, Mobile Security Service, and MSAS proxy users.

Mobile and Social, Security Token Service, Mobile Security, and MSAS proxy Service cannot be used in OAM 12*c* (12.2.1.3.0). If your current installation makes use of any of these services, you must disable them before attempting to perform this upgrade. If any of these services are active during the upgrade, the upgrade will fail with an **upgrade not feasible** error message. You can find additional information about these features in the Oracle Mobile Security Suite Statement Of Direction support document.

# Creating 12c Oracle Home Folder on OAMHOST1 and OAMHOST2

Create a folder for 12*c* Oracle Home on both OAMHOST1 and OAMHOST2.

It is recommended that you have the identical directory structure on OAMHOST1 and OAMHOST2.
For example:

`/home/Oracle/product/`*`ORACLE_HOME`*

# Installing Product Distributions on OAMHOST1 and OAMHOST2

You must install the 12c binaries onto OAMHOST1 and OAMHOST2 or onto shared storage accessible by both. If you are using redundant binaries ensure you install into each of the redundant locations

The following products must be installed on both OAMHOST1 and OAMHOST2:

- Oracle Fusion Middleware Infrastructure 12*c* (12.2.1.3.0)
- Oracle Access Manager 12*c* (12.2.1.3.0)
- Any additional distributions for your pre-upgrade environment

For instructions to install the 12c binaries, see Installing Product Distributions.

> **Note:**
>
> If you have redundant *Oracle_Home* installations, binaries must be installed into each of the redundant locations.

- Installing Product Distributions
  Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure and Oracle Access Manager 12*c* (12.2.1.3.0) distributions on the target system and install them using Oracle Universal Installer.

## Installing Product Distributions

Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure and Oracle Access Manager 12*c* (12.2.1.3.0) distributions on the target system and install them using Oracle Universal Installer.

> **Note:**
>
> - The 12c binaries are installed in a different location from the previous 11g binaries. You can install 12c binaries before any planned downtime for upgrade.
> - If you are using Redundant binary locations, ensure that you install the software into each of those redundant locations.

To install the 12*c* (12.2.1.3.0) distributions:

1. Sign in to the target system.
2. Download the following from Oracle Technical Resources or Oracle Software Delivery Cloud to your target system:
   - Oracle Fusion Middleware Infrastructure (`fmw_12.2.1.3.0_infrastructure_generic.jar`)

- Oracle Access Manager (`fmw_12.2.1.3.0_idm_generic.jar`)
- Any additional distributions for your pre-upgrade environment

> **Note:**
>
> If you are upgrading an integrated environment that was set up using Life Cycle Management (LCM) tool, that includes Oracle Access Manager, Oracle Identity Manager, and WebGates, then you must install the respective 12*c* Web Server (Oracle HTTP Server or Oracle Traffic Director) binaries in the same Oracle Home.

3. Change to the directory where you downloaded the 12*c* (12.2.1.3.0) product distribution.
4. Start the installation program for Oracle Fusion Middleware Infrastructure:
   - (UNIX) *JAVA_HOME*`/bin/java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`
   - (Windows) *JAVA_HOME*`\bin\java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`
5. On UNIX operating systems, the Installation Inventory Setup screen appears if this is the first time you are installing an Oracle product on this host.

   Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location, and click **Next**.

   > **Note:**
   >
   > The Installation Inventory Setup screen does not appear on Windows operating systems.

6. On the Welcome screen, review the information to make sure that you have met all the prerequisites. Click **Next**.
7. On the Auto Updates screen, select an option:
   - **Skip Auto Updates**: If you do not want your system to check for software updates at this time.
   - **Select patches from directory**: To navigate to a local directory if you downloaded patch files.
   - **Search My Oracle Support for Updates**: To automatically download software updates if you have a My Oracle Support account. You must enter Oracle Support credentials then click **Search**. To configure a proxy server for the installer to access My Oracle Support, click **Proxy Settings**. Click **Test Connection** to test the connection.

   Click **Next**.
8. On the Installation Location screen, specify the location for the Oracle home directory and click **Next**.

   For more information about Oracle Fusion Middleware directory structure, see About the Directories for Installation and Configuration in *Planning an Installation of Oracle Fusion Middleware*.

9. On the Installation Type screen, select the following:
   - For Infrastructure, select **Fusion Middleware Infrastructure**
   - For Oracle Access Manager, select **Collocated Oracle Identity and Access Manager**.

   Click **Next**.
10. The Prerequisite Checks screen analyzes the host computer to ensure that the specific operating system prerequisites have been met.

    To view the list of tasks that are verified, select **View Successful Tasks**. To view log details, select **View Log**. If any prerequisite check fails, then an error message appears at the bottom of the screen. Fix the error and click **Rerun** to try again. To ignore the error or the warning message and continue with the installation, click **Skip** (not recommended).
11. On the Installation Summary screen, verify the installation options that you selected.

    If you want to save these options to a response file, click **Save Response File** and enter the response file location and name. The response file collects and stores all the information that you have entered, and enables you to perform a silent installation (from the command line) at a later time.
    Click **Install** to begin the installation.
12. On the Installation Progress screen, when the progress bar displays 100%, click **Finish** to dismiss the installer, or click **Next** to see a summary.
13. The Installation Complete screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.
14. After you have installed Oracle Fusion Middleware Infrastructure, enter the following command to start the installer for your product distribution and repeat the steps above to navigate through the installer screens:

    (UNIX) `JAVA_HOME/bin/java -jar fmw_12.2.1.3.0_idm_generic.jar`

    (Windows) `JAVA_HOME\bin\java -jar fmw_12.2.1.3.0_idm_generic.jar`

> **Note:**
>
> - If your 11.1.2.3.0 setup was deployed using Life Cycle Management (LCM) tool, you must install Oracle HTTP Server 12*c* (12.2.1.3.0) in the 12*c* Middleware home. See Preparing to Install and Configure Oracle HTTP Server in *Installing and Configuring Oracle HTTP Server*.
> - By using the opatch tool, apply the latest recommended patchsets from Oracle Support. Complete only the binary installation of patchsets and follow any post-patch steps after the upgrade process is complete. This provides the latest known fixes for upgrade process, if any.

# Installing the Latest Stack Patch Bundle

After you install the product distributions, Oracle strongly recommends you to apply the latest IDM Stack Patch Bundle (SPB) 12.2.1.3.0 before proceeding with the

upgrade process. You can apply the patch by using the Opatch tool. Applying the SPB helps eliminate most of the upgrade issues or workarounds.

Following are the high-level tasks you should complete to apply the Stack Patch Bundle:

- **Initial Preparation**: In this phase, you stage the software, read the `README.txt` file, and verify and/or update the Opatch tool to the appropriate versions.
- **Analysis Phase**: In this phase, you run the `prestop` command with the variables from the `README.txt` file to determine if the system is ready for patching.
- **Patching Phase**: In this phase, you backup *MW_HOME* and *DOMAIN_HOME*, run the downtime command for OIG with the variables from the `README.txt` file, and then clear any temporary files.

> **Note:**
>
> At this point, you will not restart the servers. There is currently no link between the schemas, the local configuration, and the new bits. The remainder of the patching process will happen after the bootstrap.

To avoid a false failure during the domain Reconfiguration Phase of the upgrade, after completing the Patching Phase, update the following entries in the `config.xml` for the `com.oracle.cie.comdev_7.8.2.0` and `com.oracle.cie.xmldh_3.4.2.0` libraries:

```
<name>com.oracle.cie.comdev#3.0.0.0@7.8.2.0</name>
com.oracle.cie.comdev_7.8.2.0.jar


<name>com.oracle.cie.xmldh#2.0.0.0@3.4.2.0</name>
com.oracle.cie.xmldh_3.4.2.0.jar
```

From:

```
<library>
<name>com.oracle.cie.comdev#3.0.0.0@7.8.2.0</name>
<target>oim_cluster</target>
<source-path><MW_HOME>/oracle_common/modules/
com.oracle.cie.comdev_7.8.2.0.jar
</source-path>
<deployment-order>511</deployment-order>
<security-dd-model>DDOnly</security-dd-model>
<staging-mode>nostage</staging-mode>
</library>

<library>
<name>com.oracle.cie.xmldh#2.0.0.0@3.4.2.0</name>
<target>oim_cluster</target>
<source-path><MW_HOME>/oracle_common/modules/
com.oracle.cie.xmldh_3.4.2.0.jar<
/source-path>
<deployment-order>511</deployment-order>
<security-dd-model>DDOnly</security-dd-model>
```

```
<staging-mode>nostage</staging-mode>
</library>
```

To this:

```
<library>
<name>com.oracle.cie.comdev#3.0.0.0@7.8.4.0</name>
<target>oim_cluster</target>
<source-path><MW_HOME>/oracle_common/modules/
com.oracle.cie.comdev_7.8.4.0.jar
</source-path>
<deployment-order>511</deployment-order>
<security-dd-model>DDOnly</security-dd-model>
<staging-mode>nostage</staging-mode>
</library>

<library>
<name>com.oracle.cie.xmldh#2.0.0.0@3.4.4.0</name>
<target>oim_cluster</target>
<source-path><MW_HOME>/oracle_common/modules/
com.oracle.cie.xmldh_3.4.4.0.jar<
/source-path>
<deployment-order>511</deployment-order>
<security-dd-model>DDOnly</security-dd-model>
<staging-mode>nostage</staging-mode>
</library>
```

This update to the `config.xml` file changes the name of the libraries and version of the jar file in each library to the one that will be used post the patching process. Ensure that both nodes have the same settings.

For more information on the patching process, see Doc ID 2657920.1.

> **Note:**
>
> If you are using Windows or Solaris OS, download the individual Bundle Patches (BPs) from Doc ID 2457034.1.

After completing the upgrade, you have to perform the post-patch install steps. See Performing the Post-Patch Install Steps.

# Upgrading Schemas on OAMHOST1

Upgrade all of the necessary schemas for Oracle Access Manager, on OAMHOST1 by using the Upgrade Assistant.

- Upgrading Product Schemas
  After stopping servers and processes, use the Upgrade Assistant to upgrade supported product schemas to the current release of Oracle Fusion Middleware.

# Upgrading Product Schemas

After stopping servers and processes, use the Upgrade Assistant to upgrade supported product schemas to the current release of Oracle Fusion Middleware.

The Upgrade Assistant allows you to upgrade individually selected schemas or all schemas associated with a domain. The option you select determines which Upgrade Assistant screens you will use.

- Identifying Existing Schemas Available for Upgrade
  This optional task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.
- Starting the Upgrade Assistant
  Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.
- Upgrading Oracle Access Manager Schemas Using the Upgrade Assistant
  Navigate through the screens in the Upgrade Assistant to upgrade the product schemas.
- Verifying the Schema Upgrade
  After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

## Identifying Existing Schemas Available for Upgrade

This optional task enables you to review the list of available schemas before you begin the upgrade by querying the schema version registry. The registry contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.

You can let the Upgrade Assistant upgrade all of the schemas in the domain, or you can select individual schemas to upgrade. To help decide, follow these steps to view a list of all the schemas that are available for an upgrade:

1. If you are using an Oracle database, connect to the database by using an acount that has Oracle DBA privileges, and run the following from SQL*Plus:

   ```
   SET LINE 120
   COLUMN MRC_NAME FORMAT A14
   COLUMN COMP_ID FORMAT A20
   COLUMN VERSION FORMAT A12
   COLUMN STATUS FORMAT A9
   COLUMN UPGRADED FORMAT A8
   SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM
   SCHEMA_VERSION_REGISTRY ORDER BY MRC_NAME, COMP_ID;
   ```

2. Examine the report that is generated.

   If an upgrade is not needed for a schema, the `schema_version_registry` table retains the schema at its pre-upgrade version.

3. Note the schema prefix name that was used for your existing schemas. You will use the same prefix when you create new 12*c* schemas.

> **Notes:**
>
> - If you used an OID-based policy store in 11*g*, make sure to create a new OPSS schema before you perform the upgrade. After the upgrade, the OPSS schema remains an LDAP-based store.
> - You can only upgrade schemas for products that are available for upgrade in Oracle Fusion Middleware release 12*c* (12.2.1.3.0). Do not attempt to upgrade a domain that includes components that are not yet available for upgrade to 12*c* (12.2.1.3.0).

## Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.

To start the Upgrade Assistant:

> **Note:**
>
> Before you start the Upgrade Assistant, make sure that the JVM character encoding is set to UTF-8 for the platform on which the Upgrade Assistant is running. If the character encoding is not set to UTF-8, then you will not be able to download files containing Unicode characters in their names. This can cause the upgrade to fail.
>
> To ensure that UTF-8 is used by the JVM, use the JVM option `-Dfile.encoding=UTF-8`.

1. Go to the `oracle_common/upgrade/bin` directory:
   - (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/bin`
   - (Windows) *ORACLE_HOME*`\oracle_common\upgrade\bin`
2. Start the Upgrade Assistant:
   - (UNIX) `./ua`
   - (Windows) `ua.bat`

> **Note:**
>
> In the above command, *ORACLE_HOME* refers to the 12*c* (12.2.1.3.0) Oracle Home.

For information about other parameters that you can specify on the command line, such as logging parameters, see:

- Upgrade Assistant Parameters

## Upgrade Assistant Parameters

When you start the Upgrade Assistant from the command line, you can specify additional parameters.

**Table 4-2 Upgrade Assistant Command-Line Parameters**

| Parameter | Required or Optional | Description |
|---|---|---|
| `-readiness` | Required for readiness checks<br>**Note**: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server). | Performs the upgrade readiness check without performing an actual upgrade.<br>Schemas and configurations are checked.<br>Do not use this parameter if you have specified the `-examine` parameter. |
| `-threads` | Optional | Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas.<br>The value must be a positive integer in the range 1 to 8. The default is 4. |
| `-response` | Required for silent upgrades or silent readiness checks | Runs the Upgrade Assistant using inputs saved to a response file generated from the data that is entered when the Upgrade Assistant is run in GUI mode. Using this parameter runs the Upgrade Assistant in *silent mode* (without displaying Upgrade Assistant screens). |
| `-examine` | Optional | Performs the examine phase but does not perform an actual upgrade.<br>Do not specify this parameter if you have specified the `-readiness` parameter. |

**Table 4-2 (Cont.) Upgrade Assistant Command-Line Parameters**

| Parameter | Required or Optional | Description |
|---|---|---|
| `-logLevel` *attribute* | Optional | Sets the logging level, specifying one of the following attributes:<br>• `TRACE`<br>• `NOTIFICATION`<br>• `WARNING`<br>• `ERROR`<br>• `INCIDENT_ERROR`<br>The default logging level is `NOTIFICATION`.<br>Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used. |
| `-logDir` *location* | Optional | Sets the default location of upgrade log files and temporary files. You must specify an existing, writable directory where the Upgrade Assistant creates log files and temporary files.<br>The default locations are:<br>(UNIX)<br>*ORACLE_HOME*`/oracle_common/upgrade/logs`<br>*ORACLE_HOME*`/oracle_common/upgrade/temp`<br>(Windows)<br>*ORACLE_HOME*`\oracle_common\upgrade\logs`<br>*ORACLE_HOME*`\oracle_common\upgrade\temp` |
| `-help` | Optional | Displays all of the command-line options. |

# Upgrading Oracle Access Manager Schemas Using the Upgrade Assistant

Navigate through the screens in the Upgrade Assistant to upgrade the product schemas.

> **Caution:**
>
> You can skip this step if you have already upgraded your schemas using RCU.

> **Note:**
>
> - If the pre-upgrade environment has Audit schema (IAU), you must first upgrade Audit schema only, using the **Individually Selected Schema** option on the Selected Schemas screen, and selecting **Oracle Audit Services** schema. Ensure that you select the appropriate IAU schema from the list of available IAU schemas. The upgrade assistant will not detect the corresponding IAU schema from the provided domain directory automatically. Hence, you must select it manually. Once the IAU schema is upgraded, run the Upgrade Assistant again to upgrade the remaining schemas using the **All Schema Used by a domain** option on the Selected Schemas screen.
> - If there is no Audit schema (IAU) in your pre-upgrade environment, use the **All Schema Used by a Domain** option on the Selected Schemas screen and proceed.
> - To check whether the pre-upgrade environment has the IAU schema, run the following SQL command using the user with sysdba privileges:
>
>   ```
>   select username from dba_users where username like '%IAU%';
>   ```
>
>   This command lists the IAU schemas available in your configured database.

To upgrade product schemas with the Upgrade Assistant:

1. On the Welcome screen, review an introduction to the Upgrade Assistant and information about important pre-upgrade tasks. Click **Next**.

   > **Note:**
   >
   > For more information about any Upgrade Assistant screen, click **Help** on the screen.

2. On the Selected Schemas screen, select the schema upgrade operation that you want to perform:
   - **Individually Selected Schemas** if you want to select individual schemas for upgrade and you do not want to upgrade all of the schemas used by the domain.

> **Caution:**
>
> Upgrade only those schemas that are used to support your 12*c* (12.2.1.3.0) components. Do not upgrade schemas that are currently being used to support components that are not included in Oracle Fusion Middleware 12*c* (12.2.1.3.0).

- **All Schemas Used by a Domain** to allow the Upgrade Assistant to discover and select all components that have a schema available to upgrade in the domain specified in the **Domain Directory** field. This is also known as a *domain assisted schema upgrade*. Additionally, the Upgrade Assistant pre-populates connection information on the schema input screens.

> **Note:**
>
> Oracle recommends that you select **All Schemas Used by a Domain** for most upgrades to ensure all of the required schemas are included in the upgrade.

> **Note:**
>
> If your 11g domain contains Oracle Identity Navigator, choose **Individually Selected Schemas** and select only the Oracle Access Manager (OAM) and the OAM-related schemas.
>
> Do *not* select Oracle Identity Navigator (OIN) and OIN-related schemas, as Oracle Identity Navigator is not supported in 12c.

Click **Next**.

3. If you selected **Individually Selected Schemas**: On the Available Components screen, select the components for which you want to upgrade schemas. When you select a component, the schemas and any dependencies are automatically selected.

   If you selected **All schemas used by a domain**: On the Create Schema screen, enter the necessary Database details. This retrieves all of the schemas in the domain.

   Click **Next**.

4. On the Prerequisites screen, acknowledge that the prerequisites have been met by selecting all the check boxes. Click **Next**.

> **Note:**
>
> The Upgrade Assistant does not verify whether the prerequisites have been met.

5. On the Schema Credentials screen(s), specify the database connection details for each schema you are upgrading (the screen name changes based on the schema selected):
   - Select the database type from the **Database Type** drop-down menu.
   - Enter the database connection details, and click **Connect**.
   - Select the schema you want to upgrade from the **Schema User Name** drop-down menu, and then enter the password for the schema. Be sure to use the correct schema prefix for the schemas you are upgrading.

   > **Note:**
   >
   > The component ID or schema name is changed for `UCSUMS` schema as of release 12.1.2, which means the Upgrade Assistant does not automatically recognize the possible schemas and display them in a drop-down list. You must manually enter the name in a text field. The name can be either *`prefix`*`_ORASDPM` or *`prefix`*`_UMS`, depending on the starting point for the upgrade.
   > The `UCSUMS` schema is not auto-populated. Enter *`prefix`*`_ORASDPM` as the user. The upgrade environment uses `_ORASDPM` as the schema name, whereas in the 12*c* environment it is referred to as `_UMS`.

6. On the Examine screen, review the status of the Upgrade Assistant as it examines each schema, verifying that the schema is ready for upgrade. If the status is **Examine finished**, click **Next**.

   If the examine phase fails, Oracle recommends that you cancel the upgrade by clicking **No** in the Examination Failure dialog. Click **View Log** to see what caused the error and refer to Troubleshooting Your Upgrade in *Upgrading with the Upgrade Assistant* for information on resolving common upgrade errors.

   > **Note:**
   >
   > - If you resolve any issues detected during the examine phase without proceeding with the upgrade, you can start the Upgrade Assistant again without restoring from backup. However, if you proceed by clicking **Yes** in the Examination Failure dialog box, you need to restore your pre-upgrade environment from backup before starting the Upgrade Assistant again.
   > - Canceling the examination process has no effect on the schemas or configuration data; the only consequence is that the information the Upgrade Assistant has collected must be collected again in a future upgrade session.

7. On the Upgrade Summary screen, review the summary of the options you have selected for schema upgrade.

   Verify that the correct Source and Target Versions are listed for each schema you intend to upgrade.

   If you want to save these options to a response file to run the Upgrade Assistant again later in response (or silent) mode, click **Save Response File** and provide the location

and name of the response file. A silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again.

Click **Upgrade** to start the upgrade process.

8. On the Upgrade Progress screen, monitor the status of the upgrade.

> **Caution:**
>
> Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

If any schemas are not upgraded successfully, refer to the Upgrade Assistant log files for more information.

> **Note:**
>
> The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

Click **Next**.

9. If the upgrade is successful: On the Upgrade Success screen, click **Close** to complete the upgrade and close the wizard.

   If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at *ORACLE_HOME*`/oracle_common/upgrade/logs`.

> **Note:**
>
> If the upgrade fails, you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

## Verifying the Schema Upgrade

After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

If you are using an Oracle database, connect to the database as a user having Oracle DBA privileges, and run the following from SQL*Plus to get the current version numbers:

```
SET LINE 120
COLUMN MRC_NAME FORMAT A14
COLUMN COMP_ID FORMAT A20
COLUMN VERSION FORMAT A12
COLUMN STATUS FORMAT A9
```

```
COLUMN UPGRADED FORMAT A8
SELECT MRC_NAME, COMP_ID, OWNER, VERSION, STATUS, UPGRADED FROM
SCHEMA_VERSION_REGISTRY ORDER BY MRC_NAME, COMP_ID ;
```

In the query result:

- Check that the number in the `VERSION` column matches the latest version number for that schema. For example, verify that the schema version number is 12.2.1.3.0.

  > **Note:**
  >
  > However, that not all schema versions will be updated. Some schemas do not require an upgrade to this release and will retain their pre-upgrade version number.

- The `STATUS` field will be either `UPGRADING` or `UPGRADED` during the schema patching operation, and will become `VALID` when the operation is completed.
- If the status appears as `INVALID`, the schema update failed. You should examine the logs files to determine the reason for the failure.
- Synonym objects owned by `IAU_APPEND` and `IAU_VIEWER` will appear as `INVALID`, but that does not indicate a failure.

  They become invalid because the target object changes after the creation of the synonym. The synonyms objects will become valid when they are accessed. You can safely ignore these `INVALID` objects.

> **Note:**
>
> Undo any non-SSL port changes and any non-SYSDBA user that you made when preparing for the upgrade.

# Reconfiguring the Domain on OAMHOST1

Run the Reconfiguration Wizard on OAMHOST1 to reconfigure your domain component configurations to 12*c* (12.2.1.3.0).

- About Reconfiguring the Domain
  Run the Reconfiguration Wizard to reconfigure your domain component configurations to 12*c* (12.2.1.3.0).

## About Reconfiguring the Domain

Run the Reconfiguration Wizard to reconfigure your domain component configurations to 12*c* (12.2.1.3.0).

When you reconfigure a WebLogic Server domain, the following items are automatically updated, depending on the applications in the domain:

- WebLogic Server core infrastructure
- Domain version

> **Note:**
>
> Before you begin the domain reconfiguration, note the following limitations:
>
> - The Reconfiguration Wizard does not update any of your own applications that are included in the domain.
> - Transforming a non-dynamic cluster domain to a dynamic cluster domain during the upgrade process is not supported.
>
>  The dynamic cluster feature is available when running the Reconfiguration Wizard, but Oracle only supports upgrading a non-dynamic cluster upgrade and then adding dynamic clusters. You cannot add dynamic cluster during the upgrade process.

Specifically, when you reconfigure a domain, the following occurs:

- The domain version number in the `config.xml` file for the domain is updated to the Administration Server's installed WebLogic Server version.
- Reconfiguration templates for all installed Oracle products are automatically selected and applied to the domain. These templates define any reconfiguration tasks that are required to make the WebLogic domain compatible with the current WebLogic Server version.
- Start scripts are updated.

  If you want to preserve your modified start scripts, be sure to back them up before starting the Reconfiguration Wizard.

> **Note:**
>
> When the domain reconfiguration process starts, you can't undo the changes that it makes. Before running the Reconfiguration Wizard, ensure that you have backed up the domain as covered in the pre-upgrade checklist. If an error or other interruption occurs while running the Reconfiguration Wizard, you must restore the domain by copying the files and directories from the backup location to the original domain directory. This is the only way to ensure that the domain has been returned to its original state before reconfiguration.

Follow these instructions to reconfigure the existing domain using the Reconfiguration Wizard. See Reconfiguring WebLogic Domains in *Upgrading Oracle WebLogic Server*.

- Backing Up the Domain
- Starting the Reconfiguration Wizard
- Reconfiguring the Oracle Access Manager Domain
  Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing 11g domain.

## Backing Up the Domain

Before running the Reconfiguration Wizard, create a backup copy of the domain directory.

To create a backup of the Administration server domain directory:

1. Copy the source domain to a separate location to preserve the contents.

   (Windows) `copy /home/Oracle/config/domains to /home/Oracle/config/domains_backup`.

   (UNIX) `cp -rf domains domains_backup`

2. For HA environments, before updating the domain on each remote Managed Server, create a backup copy of the domain directory on each remote machine.
3. Verify that the backed up versions of the domain are complete.

If domain reconfiguration fails for any reason, you must restore all files and directories from the backup directory into the original domain directory to ensure that the domain is returned entirely to its original state before reconfiguration.

## Starting the Reconfiguration Wizard

> **Note:**
>
> Shut down the administration server and all collocated managed servers before starting the reconfiguration process. See Stopping Servers and Processes.

To start the Reconfiguration Wizard in graphical mode:

1. Open the command shell (on UNIX operating systems) or open a command prompt window (on Windows operating systems).
2. **Edition Based Database Users Only**: If your schemas are configured with EBR database, a default edition name must be manually supplied before you run the Reconfiguration Wizard.

   Run the following SQL command to set the default edition:
   `ALTER DATABASE DEFAULT EDITION = edition_name;`

   where *edition_name* is the child edition name.
3. Go to the `oracle_common/common/bin` directory:
   - (UNIX) *ORACLE_HOME*`/oracle_common/common/bin`
   - (Windows) *ORACLE_HOME*`\oracle_common\commom\bin`

   Where, *ORACLE_HOME* is the 12c Oracle Home.
4. Start the Reconfiguration Wizard:

   The `./reconfig.sh` command, might display the following error to indicate that the default cache directory is not valid:

   ```
   *sys-package-mgr*: can't create package cache dir
   ```

So, first, change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`.

For example: `CONFIG_JVM_ARGS=-Dpython.cachedir=`*`valid_directory`*

Start the Reconfiguration Wizard with the following logging options:

- (UNIX) `./reconfig.sh -log=`*`log_file`*` -log_priority=ALL`
- (Windows) `reconfig.cmd -log=`*`log_file`*` -log_priority=ALL`

where *`log_file`* is the absolute path of the log file you'd like to create for the domain reconfiguration session. This can be helpful if you need to troubleshoot the reconfiguration process.

The parameter `-log_priority=ALL` ensures that logs are logged in fine mode.

## Reconfiguring the Oracle Access Manager Domain

Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing 11g domain.

> **Note:**
>
> If the source is a clustered environment, run the Reconfiguration Wizard on the primary node only. Where, primary node is the Administration Server. Use the pack/unpack utility to apply the changes to other cluster members in the domain.

To reconfigure the domain with the Reconfiguration Wizard:

1. On the Select Domain screen, specify the location of the domain you want to upgrade or click **Browse** to navigate and select the domain directory. Click **Next**.
2. On the Reconfiguration Setup Progress screen, view the progress of the setup process. When complete, click **Next**.

   During this process:

   - The reconfiguration templates for your installed products, including Fusion Middleware products, are automatically applied. This updates various domain configuration files such as `config.xml`, `config-groups.xml`, and `security.xml` (among others).
   - Schemas, scripts, and other such files that support your Fusion Middleware products are updated.
   - The domain upgrade is validated.
3. On the Domain Mode and JDK screen, select the JDK to use in the domain or click **Browse** to navigate to the JDK you want to use. The supported JDK version for 12*c* (12.2.1.3.0) is 1.8.0_131 and later. Click **Next**.

   > **Note:**
   >
   > You cannot change the **Domain Mode** at this stage.

For a list of JDKs that are supported for a specific platform, see Oracle Fusion Middleware Supported System Configurations.

4. On the Database Configuration Type screen, select **RCU Data** to connect to the Server Table (_STB) schema.

   Enter the database connection details using the RCU service table (_STB) schema credentials and click **Get RCU Configuration**.

   The Reconfiguration Wizard uses this connection to automatically configure the data sources required for components in your domain.

   > **Note:**
   >
   > By default **Oracle's Driver (Thin) for Service connections; Versions: Any** is the selected driver. If you specified an instance name in your connection details — instead of the service name — you must select **Oracle's Driver (Thin) for pooled instance connections; Versions: Any** If you do not change the driver type, then the connection will fail.
   > For information about selecting grid link for RAC databases in HA environments, see Access Manager High Availability Architecture.

   > **Note:**
   >
   > For any existing 11*g* datasource, the reconfiguration will preserve the existing values. For new datasources where the schema was created for 12*c* by the RCU, the default connection data will be retrieved from the _STB schema. If no connection data for a given schema is found in the _STB schema, then the default connection data is used.

   If the check is successful, click **Next**. If the check fails, reenter the connection details correctly and try again.

   > **Note:**
   >
   > If your database has _OPSS or _IAU 11*g* database schemas, you must manually enter database connection details for those schemas. These schemas were not required in 11*g* and had to be created manually. Users could assign any name to these schemas, therefore the Reconfiguration Wizard does not recognize them. When providing connection information for _IAU, use the IAU_APPEND user information.

5. On the JDBC Component Schema screen, verify that the DBMS/Service and the Host name is correct for the following component schemas:

   - OPSS Audit schema
   - OPSS Audit viewer schema
   - OPSS schema

If you are connecting to a RAC database, select each of the schemas you want to update and click **Convert to Grid Link**. Click **Next** to update the Service Name, Schema Password, SCAN, Hostname/Port, ONS Host/Port.

Click **Next**.

6. On the JDBC Component Schema Test screen, select all the component schemas and click **Test Selected Connections** to test the connection for each schema. The result of the test is indicated in the Status column.

   When the check is complete, click **Next**.

7. On the Node Manager screen, select the appropriate Node Manager Type based on your requirements, specify the details, and click **Next**.

> **Note:**
>
> There are two types of node managers. It is recommend to use the domain-based node manager, so that, you can have different versions of the node manager for each domain.

8. On the Advanced Configuration screen, select **Administration Server**, **Topology**, and **Deployments and Services**. Select **Domain Frontend Host Capture** if required.

   For each of the categories you select, the appropriate configuration screen is displayed to allow you to perform advanced configuration.

> **Note:**
>
> Ensure that you assign oam_server1 or the OAM managed server name used to the server group **OAM-MDG-SVRS**, and `oam_policy_mgr1` to the server group **OAM-POLICY-MANAGED-SERVER**.

9. On the Configuration Summary screen, review the detailed configuration settings of the domain before continuing.

   You can limit the items that are displayed in the right-most panel by selecting a filter option from the **View** drop-down list.

   To change the configuration, click **Back** to return to the appropriate screen. To reconfigure the domain, click **Reconfig**.

> **Note:**
>
> The location of the domain does not change when you reconfigure it.

10. The Reconfiguration Progress screen displays the progress of the reconfiguration process.

    During this process:

    - Domain information is extracted, saved, and updated.
    - Schemas, scripts, and other such files that support your Fusion Middleware products are updated.

When the progress bar shows 100%, click **Next**.

11. The End of Configuration screen indicates whether the reconfiguration process completed successfully or failed. It also displays the location of the domain that was reconfigured as well as the Administration Server URL (including the listen port). If the reconfiguration is successful, it displays **Oracle WebLogic Server Reconfiguration Succeeded**.

    If the reconfiguration process did not complete successfully, an error message is displayed indicates the reason. Take appropriate action to resolve the issue. If you cannot resolve the issue, contact My Oracle Support.

    Note the Domain Location and the Admin Server URL for further operations.

# Replicating the Domain Configurations on each OAMHOST

Replicate the domain configurations on OAMHOST2. This involves packing the upgraded domain on OAMHOST1 and unpacking it on OAMHOST2.

To do this, complete the following steps:

1. On OAMHOST1, run the following command from the location `$ORACLE_HOME/oracle_common/common/bin` to pack the upgraded domain:

   - On UNIX:
     ```
     sh pack.sh -domain=<Location_of_OAM_domain> -
     template=<Location_where_domain_configuration_jar_to_be_created> -
     template_name="OAM Domain" -managed=true
     ```
   - On Windows:
     ```
     pack.cmd -domain=<Location_of_OAM_domain> -
     template=<Location_where_domain_configuration_jar_to_be_created> -
     template_name="OAM Domain" -managed=true
     ```

2. Copy the domain configuration jar file created by the pack command on OAMHOST1 to any accessible location on OAMHOST2.

3. On OAMHOST2, run the following command from the location `$ORACLE_HOME/oracle_common/common/bin` to unpack the domain:

   - On UNIX:
     ```
     sh unpack.sh -domain=<Location_of_OAM_domain> -template=<absolute_path_to
     the_location_of_domain_configuration_jar_file> -overwrite_domain=true
     ```
   - On Windows:
     ```
     unpack.cmd -domain=<Location_of_OAM_domain> -template=<absolute_path_to
     the_location_of_domain_configuration_jar_file> -overwrite_domain=true
     ```

4. If you have other OAMHOSTs, repeat step 2 through step 3 on those hosts.

> **Note:**
>
> If you are following the EDG methodology you also need to pack and unpack the domain in the OAM managed server location on OAMHOST1.

# Upgrading Domain Component Configurations on OAMHOST1 and OAMHOST2

After reconfiguring the domain, use the Upgrade Assistant to upgrade the domain component configurations inside the domain to match the updated domain configuration.

Upgrade the domain configurations on both OAMHOST1 and OAMHOST2.

- Upgrading Domain Component Configurations
  After reconfiguring the domain, use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.

## Upgrading Domain Component Configurations

After reconfiguring the domain, use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.

- Starting the Upgrade Assistant
  Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.
- Upgrading Oracle Access Manager Domain Component Configurations
  Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.
- Removing Oracle Mobile Security Manager Servers From the Domain
  Remove the Oracle Mobile Security Manager (MSM) servers from the upgraded domain, as they are not supported in 12*c* (12.2.1.3.0).

## Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (12.2.1.3.0). Oracle recommends that you run the Upgrade Assistant as a non-SYSDBA user, completing the upgrade for one domain at a time.

To start the Upgrade Assistant:

> **Note:**
>
> Before you start the Upgrade Assistant, make sure that the JVM character encoding is set to UTF-8 for the platform on which the Upgrade Assistant is running. If the character encoding is not set to UTF-8, then you will not be able to download files containing Unicode characters in their names. This can cause the upgrade to fail.
>
> To ensure that UTF-8 is used by the JVM, use the JVM option `-Dfile.encoding=UTF-8`.

1. Go to the `oracle_common/upgrade/bin` directory:
   - (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/bin`
   - (Windows) *ORACLE_HOME*`\oracle_common\upgrade\bin`
2. Start the Upgrade Assistant:
   - (UNIX) `./ua`
   - (Windows) `ua.bat`

> **Note:**
>
> In the above command, *ORACLE_HOME* refers to the 12*c* (12.2.1.3.0) Oracle Home.

For information about other parameters that you can specify on the command line, such as logging parameters, see:

- Upgrade Assistant Parameters

## Upgrade Assistant Parameters

When you start the Upgrade Assistant from the command line, you can specify additional parameters.

**Table 4-3 Upgrade Assistant Command-Line Parameters**

| Parameter | Required or Optional | Description |
|---|---|---|
| `-readiness` | Required for readiness checks<br>**Note**: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server). | Performs the upgrade readiness check without performing an actual upgrade.<br>Schemas and configurations are checked.<br>Do not use this parameter if you have specified the `-examine` parameter. |

**Table 4-3 (Cont.) Upgrade Assistant Command-Line Parameters**

| Parameter | Required or Optional | Description |
|---|---|---|
| `-threads` | Optional | Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas.<br>The value must be a positive integer in the range 1 to 8. The default is 4. |
| `-response` | Required for silent upgrades or silent readiness checks | Runs the Upgrade Assistant using inputs saved to a response file generated from the data that is entered when the Upgrade Assistant is run in GUI mode. Using this parameter runs the Upgrade Assistant in *silent mode* (without displaying Upgrade Assistant screens). |
| `-examine` | Optional | Performs the examine phase but does not perform an actual upgrade.<br>Do not specify this parameter if you have specified the `-readiness` parameter. |
| `-logLevel` *attribute* | Optional | Sets the logging level, specifying one of the following attributes:<br>• `TRACE`<br>• `NOTIFICATION`<br>• `WARNING`<br>• `ERROR`<br>• `INCIDENT_ERROR`<br>The default logging level is `NOTIFICATION`.<br>Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used. |

**Table 4-3 (Cont.) Upgrade Assistant Command-Line Parameters**

| Parameter | Required or Optional | Description |
|---|---|---|
| `-logDir` *location* | Optional | Sets the default location of upgrade log files and temporary files. You must specify an existing, writable directory where the Upgrade Assistant creates log files and temporary files.<br>The default locations are:<br>(UNIX)<br>*ORACLE_HOME*`/oracle_common/upgrade/logs`<br>*ORACLE_HOME*`/oracle_common/upgrade/temp`<br>(Windows)<br>*ORACLE_HOME*`\oracle_common\upgrade\logs`<br>*ORACLE_HOME*`\oracle_common\upgrade\temp` |
| `-help` | Optional | Displays all of the command-line options. |

## Upgrading Oracle Access Manager Domain Component Configurations

Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.

After running the Reconfiguration Wizard to reconfigure the WebLogic domain to 12*c* (12.2.1.3.0), you must run the Upgrade Assistant to upgrade the domain *component* configurations to match the updated domain configuration.

To upgrade domain component configurations with the Upgrade Assistant:

1. On the Welcome screen, review an introduction to the Upgrade Assistant and information about important pre-upgrade tasks. Click **Next**.

> **Note:**
>
> For more information about any Upgrade Assistant screen, click **Help** on the screen.

2. On the next screen:
   - Select **All Configurations Used By a Domain**. The screen name changes to WebLogic Components.
   - In the **Domain Directory** field, enter the 11.1.2.3.0 domain directory path.

Click **Next**.

3. On the Component List screen, verify that the list includes all the components for which you want to upgrade configurations and click **Next**.

   If you do not see the components you want to upgrade, click **Back** to go to the previous screen and specify a different domain.

4. On the Prerequisites screen, acknowledge that the prerequisites have been met by selecting all the check boxes. Click **Next**.

   > **Note:**
   >
   > The Upgrade Assistant does not verify whether the prerequisites have been met.

5. On the Examine screen, review the status of the Upgrade Assistant as it examines each component, verifying that the component configuration is ready for upgrade. If the status is **Examine finished**, click **Next**.

   If the examine phase fails, Oracle recommends that you cancel the upgrade by clicking **No** in the Examination Failure dialog. Click **View Log** to see what caused the error and refer to Troubleshooting Your Upgrade in *Upgrading with the Upgrade Assistant* for information on resolving common upgrade errors.

   > **Note:**
   >
   > - If you resolve any issues detected during the examine phase without proceeding with the upgrade, you can start the Upgrade Assistant again without restoring from backup. However, if you proceed by clicking **Yes** in the Examination Failure dialog box, you need to restore your pre-upgrade environment from backup before starting the Upgrade Assistant again.
   > - Canceling the examination process has no effect on the configuration data; the only consequence is that the information the Upgrade Assistant has collected must be collected again in a future upgrade session.

6. On the Upgrade Summary screen, review the summary of the options you have selected for component configuration upgrade.

   The response file collects and stores all the information that you have entered, and enables you to perform a silent upgrade at a later time. The silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again. If you want to save these options to a response file, click **Save Response File** and provide the location and name of the response file.

   Click **Upgrade** to start the upgrade process.

7. On the Upgrade Progress screen, monitor the status of the upgrade.

> **Caution:**
>
> Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

If any components are not upgraded successfully, refer to the Upgrade Assistant log files for more information.
Upgrade Assistant log files location:

- (UNIX) *ORACLE_HOME*/oracle_common/upgrade/logs/ua<*timestamp*>.log
- (Windows) *ORACLE_HOME*\oracle_common\upgrade\logs\ua<*timestamp*>.log

> **Note:**
>
> The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

Click **Next**.

8. If the upgrade is successful: On the Upgrade Success screen, click **Close** to complete the upgrade and close the wizard. The Post-Upgrade Actions window describes the manual tasks you must perform to make components functional in the new installation. This window appears only if a component has post-upgrade steps.

   If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at *ORACLE_HOME*/oracle_common/upgrade/logs.

> **Note:**
>
> If the upgrade fails you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

## Removing Oracle Mobile Security Manager Servers From the Domain

Remove the Oracle Mobile Security Manager (MSM) servers from the upgraded domain, as they are not supported in 12*c* (12.2.1.3.0).

To do this, complete the following steps

1. Go to the location *DOMAIN_HOME*/servers.
2. Run the following command to remove the Oracle Mobile Security Manager server(s):

   ```
   rm MSM_Server
   ```

   In the above command, *MSM_Server* is the name of the Oracle Mobile Security Manager (MSM) server.

   For example:

   ```
   rm wls_msm1
   ```

3. Repeat the step for all of the Oracle Mobile Security Manager servers in the domain.
4. Post upgrade and after the OAM Admin Server is running, complete the following:
   a. Log in to the WLS Console.
   b. Under **Server**, check for MSM and MSAS Servers.
   c. If present, delete the server entries.

# Starting the Servers on OAMHOST1 and OAMHOST2

After you upgrade Oracle Access Manager on both OAMHOST1 and OAMHOST2, start the servers.

You must start the servers in the following order:

1. Start the Node Manager on both OAMHOST1 and OAMHOST2.
2. Start the Administration Server on OAMHOST1.
3. Start the Oracle Access Manager Managed Servers on OAMHOST1.
4. Start the Oracle Access Manager Managed Servers on OAMHOST2.

- Starting Servers and Processes
  After a successful upgrade, start all processes and servers, including the Administration Server and any Managed Servers.
- Configuring Oracle HTTP Servers to Front End OIM, and SOA Managed Servers
- Verifying the Domain-Specific-Component Configurations Upgrade
  To verify that the domain-specific-component configurations upgrade was successful, sign in to the Administration console and the Oracle Enterprise Manager Fusion Middleware Control and verify that the version numbers for each component is 12.2.1.3.0.

## Starting Servers and Processes

After a successful upgrade, start all processes and servers, including the Administration Server and any Managed Servers.

The components may be dependent on each other so they must be started in the correct order.

> **Note:**
>
> The procedures in this section describe how to start servers and process using the WLST command line or a script. You can also use the Oracle Fusion Middleware Control and the Oracle WebLogic Server Administration Console. See Starting and Stopping Administration and Managed Servers and Node Manager in *Administering Oracle Fusion Middleware*.

To start your Fusion Middleware environment, follow the steps below.

### Step 1: Start Node Manager

Start the Node Manager in the Administration Server domain home.

Go to the *WLS_HOME*/server/bin directory and run the following command:

Where, *WLS_HOME* is the top-level directory for the WebLogic Server installation.

- (UNIX) `nohup ./startNodeManager.sh > DOMAIN_HOME/nodemanager/nodemanager.out 2>&1 &`
- (Windows) `nohup .\startNodeManager.sh > DOMAIN_HOME\nodemanager\nodemanager.out 2>&1 &`

Where, *DOMAIN_HOME* is the Administration server domain home.

### Step 2: Start the Administration Server

When you start the Administration Server, you also start the processes running in the Administration Server, including the WebLogic Server Administration Console and Fusion Middleware Control.

**Method 1:** To start a Administration Server, run the following command:

```
nohup DOMAIN_HOME/bin/startWeblogic.sh &
```

**Method 2:** To start a Administration Server by using node manager, run the following commands:

```
cd ORACLE_COMMON_HOME/common/bin
./wlst.sh
wlst offline> nmConnect('nodemanager_username','nodemanager_password',
                    'ADMINVHN','5556','domain_name',
                   'DOMAIN_HOME')
nmStart('AdminServer')
```

### Step 3: Start the Managed Servers

> **Note:**
>
> In an HA environment, it is preferred to use the console or node manager to start servers.

Start a WebLogic Server Managed Server by using the Weblogic Console:

- Log into Weblogic console as a `weblogic` Admin.
- Go to **Servers** > **Control** tab.
- Select the required managed server.
- Click **Start**.

# Configuring Oracle HTTP Servers to Front End OIM, and SOA Managed Servers

Complete the following steps:

1. On each of the web servers on `WEBHOST1` and `WEBHOST2`, create a file named `mod_wls_ohs.conf` in the directory *`OHS_DOMAIN_HOME`*`/config/fmwconfig/components/OHS/instances/`*`OHS_INSTANCE_NAME`*.

   This file must contain the following information:

```
# oam admin console(idmshell based)
   <Location /admin>
    SetHandler weblogic-handler
    WLCookieName    oamjsessionid
    WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
   </Location>

# oam self and advanced admin webapp consoles(canonic webapp)

  <Location /oam>
    SetHandler weblogic-handler
    WLCookieName    oamjsessionid
    WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
   </Location>

  <Location /identity>
    SetHandler weblogic-handler
    WLCookieName    oamjsessionid
    WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
    </Location>

  <Location /sysadmin>
    SetHandler weblogic-handler
    WLCookieName    oamjsessionid
    WebLogicCluster
```

```
oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
    </Location>

# SOA Callback webservice for SOD - Provide the SOA Managed Server Ports
  <Location /sodcheck>
    SetHandler weblogic-handler
    WLCookieName    oamjsessionid
    WebLogicCluster soavhn1.example.com:7003,soavhn2.example.com:7003
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
   </Location>

# Callback webservice for SOA. SOA calls this when a request is approved/
rejected
# Provide the oam Managed Server Port
  <Location /workflowservice>
    SetHandler weblogic-handler
    WLCookieName    oamjsessionid
    WebLogicCluster oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

# xlWebApp - Legacy 9.x webapp (struts based)
   <Location /xlWebApp>
    SetHandler weblogic-handler
    WLCookieName    oamjsessionid
    WebLogicCluster oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

# Nexaweb WebApp - used for workflow designer and DM
  <Location /Nexaweb>
    SetHandler weblogic-handler
    WLCookieName    oamjsessionid
    WebLogicCluster oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

# used for FA Callback service.
  <Location /callbackResponseService>
```

```
    SetHandler weblogic-handler
    WLCookieName    oamjsessionid
    WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

# spml xsd profile
  <Location /spml-xsd>
    SetHandler weblogic-handler
    WLCookieName    oamjsessionid
    WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

  <Location /HTTPClnt>
    SetHandler weblogic-handler
    WLCookieName    oamjsessionid
    WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>


  <Location /reqsvc>
    SetHandler weblogic-handler
    WLCookieName oamjsessionid
    WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>


  <Location /integration>
    SetHandler weblogic-handler
    WLCookieName oamjsessionid
    WebLogicCluster
soavhn1.example.com:7003,soavhn2.example.com:7003
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>
```

```
  <Location /provisioning-callback>
    SetHandler weblogic-handler
    WLCookieName oamjsessionid
    WebLogicCluster oamvhn1.example.com:14000,oamvhn2.example.com:14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

  <Location /CertificationCallbackService>
   SetHandler weblogic-handler
   WLCookieName JSESSIONID
   WebLogicCluster oamvhn1.example.com:14000,oamvhn2.example.com:14000
   WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/
oam_component.log"
   WLProxySSL ON
   WLProxySSLPassThrough ON
 </Location>

  <Location /ucs>
   SetHandler weblogic-handler
   WLCookieName oamjsessionid
   WebLogicCluster soavhn1.example.com:7003,soavhn2.example.com:7003
   WLLogFile /tmp/web_log.log
   WLProxySSL ON
   WLProxySSLPassThrough ON
  </Location>

  <Location /FacadeWebApp>
   SetHandler weblogic-handler
   WLCookieName oamjsessionid
   WebLogicCluster oamvhn1.example.com:14000,oamvhn2.example.com:14000
   WLLogFile /tmp/web_log.log
   WLProxySSL ON
   WLProxySSLPassThrough ON
  </Location>

  <Location /iam/governance/configmgmt>
   SetHandler weblogic-handler
   WLCookieName oamjsessionid
   WebLogicCluster oamvhn1.example.com:14000,oamvhn2.example.com:14000
   WLLogFile /tmp/web_log.log
   WLProxySSL ON
   WLProxySSLPassThrough ON
  </Location>

  <Location /iam/governance/scim/v1>
   SetHandler weblogic-handler
   WLCookieName oamjsessionid
   WebLogicCluster oamvhn1.example.com:14000,oamvhn2.example.com:14000
   WLLogFile /tmp/web_log.log
   WLProxySSL ON
   WLProxySSLPassThrough ON>
```

```
  </Location>

  <Location /iam/governance/token/api/v1>
   SetHandler weblogic-handler
   WLCookieName oamjsessionid
   WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
   WLLogFile /tmp/web_log.log
   WLProxySSL ON
   WLProxySSLPassThrough ON
  </Location>

  <Location /OIGUI>
   SetHandler weblogic-handler
   WLCookieName oamjsessionid
   WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
   WLLogFile /tmp/web_log.log
   WLProxySSL ON
   WLProxySSLPassThrough ON
  </Location>

  <Location /iam/governance/applicationmanagement>
   SetHandler weblogic-handler
   WLCookieName oamjsessionid
   WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
   WLLogFile /tmp/web_log.log
   WLProxySSL ON
   WLProxySSLPassThrough ON
  </Location>

  <Location /iam/governance/adminservice/api/v1>
   SetHandler weblogic-handler
   WLCookieName oamjsessionid
   WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
   WLLogFile /tmp/web_log.log
   WLProxySSL ON
   WLProxySSLPassThrough ON
  </Location>

  <Location /iam/governance/selfservice/api/v1>
   SetHandler weblogic-handler
   WLCookieName oamjsessionid
   WebLogicCluster
oamvhn1.example.com:14000,oamvhn2.example.com:14000
   WLLogFile /tmp/web_log.log
   WLProxySSL ON
   WLProxySSLPassThrough ON
  </Location>
```

2. Save the file on both `WEBHOST1` and `WEBHOST2`.
3. Stop and start the Oracle HTTP Server instances on both `WEBHOST1` and `WEBHOST2`.

4. Start system components, such as Oracle HTTP Server by using the `startComponent` script:
   - (UNIX) *OHS_INSTANCE_HOME*`/bin/startComponent.sh ohs1`
   - (Windows) *OHS_INSTANCE_HOME*`\bin\startComponent.sh ohs1`

   You can start system components in any order.

## Verifying the Domain-Specific-Component Configurations Upgrade

To verify that the domain-specific-component configurations upgrade was successful, sign in to the Administration console and the Oracle Enterprise Manager Fusion Middleware Control and verify that the version numbers for each component is 12.2.1.3.0.

To sign in to the Administration Console, go to: `http://`*administration_server_host*`:`*administration_server_port*`/console`

To sign in to the Administration Console in an EDG deployment, see Validating the Virtual Server Configuration and Access to the Consoles.

To sign in to Oracle Enterprise Manager Fusion Middleware Control Console, go to: `http://`*administration_server_host*`:`*administration_server_port*`/em`

> **Note:**
>
> - After upgrade, ensure you run the administration tools from the new 12*c* Oracle home directory and not from the previous Oracle home directory.
> - During the upgrade process, some OWSM documents, including policy sets and predefined documents such as policies and assertion templates, may need to be upgraded. If a policy set or a predefined document is upgraded, its version number is incremented by 1.
> - In the site-specific configuration, the WebLogic and EM consoles must be accessible with the URLs either directly or through proxy URLs.

# Performing Post-Upgrade Tasks

After performing the upgrade of Oracle Access Manager to 12c (12.2.1.3), you should complete the tasks summarized in this section, if required.

This section includes the following tasks:

- WebGates Configuration Fails during Authentication
  WebGates configured with the `hmacEnabled=true` in environments where `globalHMACEnabled` is not set to `true` fails during authentication.
- Updating the java.security File
  If you have multiple components of Oracle Identity and Access Management (Oracle Access Manager, Oracle Identity Manager, WebGates and so on) deployed, until you upgrade all of the components to 12*c* (12.2.1.3.0), you must update the `java.security` file with the changes described in this section.

## WebGates Configuration Fails during Authentication

WebGates configured with the `hmacEnabled=true` in environments where `globalHMACEnabled` is not set to `true` fails during authentication.

To solve this issue, apply patch 12.2.1.3.181016 or later.
For more information, see Upgrading to OHS/OTD 12c WebGate.

## Updating the java.security File

If you have multiple components of Oracle Identity and Access Management (Oracle Access Manager, Oracle Identity Manager, WebGates and so on) deployed, until you upgrade all of the components to 12*c* (12.2.1.3.0), you must update the `java.security` file with the changes described in this section.

To do this:

1. Open the `java.security` file located at *`JAVA_HOME`*`/jre/lib/security/` in an editor.
2. Remove `TLSv1`, `TLSv1.1`, `MD5withRSA` from the following key:

   ```
   key - jdk.tls.disabledAlgorithms
   ```

3. Remove `MD5` from the following key:

   ```
   key - jdk.certpath.disabledAlgorithms
   ```

For more information on possible upgrade scenarios, see Troubleshooting Security Policy Issues When Upgrading.

# Performing the Post-Patch Install Steps

After completing the upgrade, you have to perform the post-patch installation steps.

The post-patch installation steps comprises the following:

- Running the Poststart Command to Confirm Successful Binary Patching
- Performing a Clean Restart of the Servers

## Running the Poststart Command to Confirm Successful Binary Patching

Use the variables and the instructions in the Stack Patch Bundle README.txt file to run the `poststart` command for your product, as shown below:

```
$ ./spbat.sh -type oig -phase poststart -mw_home /
<INSTALLATION_DIRECTORY>/IAM12c -spb_download_dir /<DOWNLOAD_LOCATION>/
IDM_SPB_12.2.1.4.200714 -log_dir /<DOWNLOAD_LOCATION>/OIGlogs
```

For details, see Doc ID 2657920.1.

# Performing a Clean Restart of the Servers

Restart all the servers including the Administration Server and any Managed Servers. See Starting Servers and Processes .

5

# Upgrading Oracle Access Manager Multi-Data Center Environments

You can upgrade Oracle Access Manager deployed across multi-data centers (MDC) from 11g Release 2 (11.1.2.3.0) to 12*c* (12.2.1.3.0).

In a multi-data center environment, where you have two OAM deployments replicating traffic, you must do the following:

- Stop Replication
- Direct all traffic to one of the deployments
- Upgrade the other deployment(s)
- Direct traffic to the newly upgraded deployment
- Upgrade the remaining deployment
- Re-establish replication

> **Note:**
>
> To upgrade Oracle Access Manager MDC environments to 12*c* (12.2.1.3.0), ensure that all of the data centers (DC) are at the same Patch Set level.

When you plan to upgrade to 12*c* (12.2.1.3.0), you can choose to have zero down time by stopping the data center that needs to be upgraded, and routing all the traffic to the other data centers. Once the upgrade has been completed on one data center, it can start and function as an independent data center. You can then redirect all the traffic to the upgraded data center. MDC Single Sign-On works between 11*g* and 12*c* Servers if backward compatibility flag is enabled. Therefore, all of the servers (upgraded and non-upgraded ones) can continue to participate in MDC.

> **Note:**
>
> For information about enabling the backward compatibility flag, see Modifying Backward Compatibility Flag in *Administering Oracle Access Manager*.

- About the Oracle Access Manager Multi-Data Center Topology
  The sample Oracle Access Manager Multi-Data Center topology has two data centers — Master data center and Clone data center.
- Roadmap for Upgrading Oracle Access Manager MDC Setup
  Use the upgrade roadmap to upgrade your Oracle Access Manager multi-data center setup to 12*c* (12.2.1.3.0).

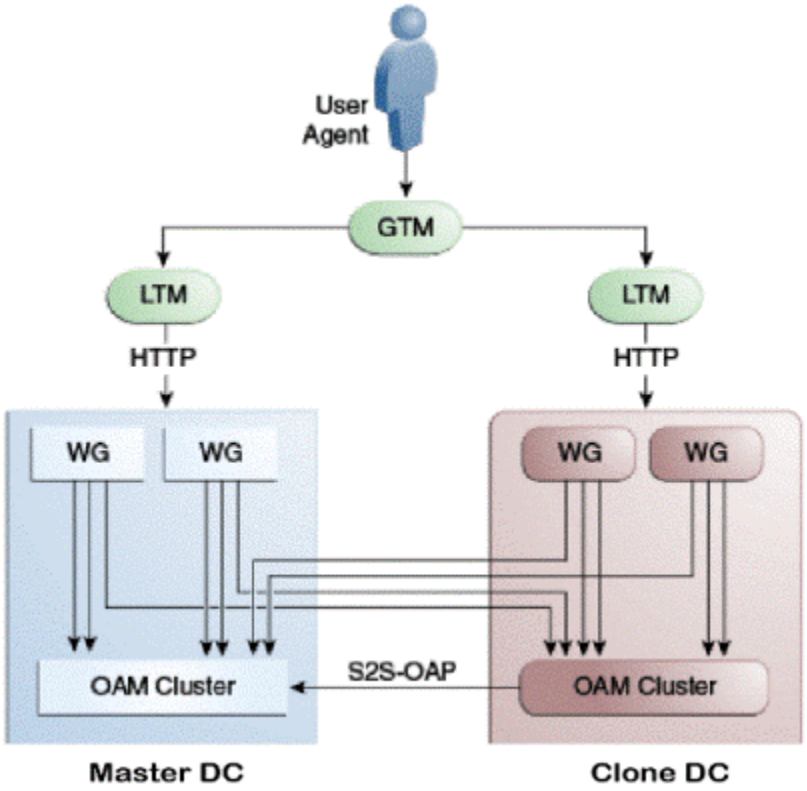
- Backing Up the Existing MDC Environment
  Before you begin with the upgrade, take a back up of your existing environment.
- Enabling Write Permission to Master and Clones (If Necessary)
  Before you start the upgrade, you must enable modifications to the system and policy configurations on both Master and Clones.
- Disabling and Deleting All Replication Agreements Between Master and Clone
  Disable all replication agreements between the Master and the Clone data centers.
- Redirecting Traffic to Master Data Center
  An in-line upgrade procedure is used to upgrade the Clone data center which requires downtime. Therefore, all traffic must be rerouted to the Master data center.
- Upgrading Oracle Access Manager on Clone Data Center
  Upgrade Oracle Access Manager on Clone data center to 12*c* (12.2.1.3.0) after you redirect the traffic to Master data center.
- Redirecting Traffic to Clone Data Center
  An in-line upgrade procedure is used to upgrade the Master data center which requires downtime. Therefore, all traffic must be rerouted to the Clone data centers (also referred to as, the backup data centers or the secondary data centers).
- Upgrading Oracle Access Manager on Master Data Center
  Upgrade Oracle Access Manager on Master data center to 12*c* (12.2.1.3.0) after you redirect the traffic to clone data center.
- Freezing all Changes to Clones (if Necessary)
  After you upgrade Oracle Access Manager on all of the Clone data center(s), it is recommended that you freeze the changes to the Clone data center(s). This is to avoid any inadvertent writes.
- Syncing Access Metadata
  Oracle Access Manager metadata stored in Unified Data Model (UDM) needs to be synced from Master to Clone.
- Creating Replication Agreement
  Create the replication agreement again after upgrading the Master and the Clone data centers.
- Updating the java.security File
  If you have multiple components of Oracle Identity and Access Management (Oracle Access Manager, Oracle Identity Manager, WebGates and so on) deployed, until you upgrade all of the components to 12*c* (12.2.1.3.0), you must update the `java.security` file with the changes described in this section.
- Bringing up the Master and Clone Data Centers Online
  After successful upgrade, both Master and Clone data centers can be brought up online. Traffic can be routed to both data centers based on existing routing rules.

# About the Oracle Access Manager Multi-Data Center Topology

The sample Oracle Access Manager Multi-Data Center topology has two data centers — Master data center and Clone data center.

The procedure in this chapter describes how to upgrade Oracle Access Manager in a MDC setup similar to the reference topology provided in this section. You can use this upgrade procedure to upgrade your environment with any number of data centers.

**Figure 5-1 Oracle Access Manager in Multi—Data Center Setup**




This figure shows a Master data center and a Clone data center, each of them including a full Access Manager installation. In this topology, GTM refers to the global load balancer, LTM refers to the local load balancer, and WG refers to the WebGate. The S2S OAP is the Oracle Access Protocol.

# Roadmap for Upgrading Oracle Access Manager MDC Setup

Use the upgrade roadmap to upgrade your Oracle Access Manager multi-data center setup to 12*c* (12.2.1.3.0).

**Table 5-1 Oracle Access Manager MDC Upgrade Roadmap**

| Task | For More Information |
|---|---|
| Review the Oracle Access Manager multi-data center topology. | See About the Oracle Access Manager Multi-Data Center Topology |
| Back up your existing environment. | See Backing Up the Existing MDC Environment |

**Table 5-1 (Cont.) Oracle Access Manager MDC Upgrade Roadmap**

| Task | For More Information |
|---|---|
| Enable write permission to Master and Clone data centers, if not already done. | See Enabling Write Permission to Master and Clones (If Necessary) |
| Disable and delete all replication agreements between Master and Clone data centers. | See Disabling and Deleting All Replication Agreements Between Master and Clone |
| Redirect the traffic to the Master data center. | See Redirecting Traffic to Master Data Center |
| Upgrade Oracle Access Manager on Clone data center. | See Upgrading Oracle Access Manager on Clone Data Center |
| Redirect the traffic to the Clone data center. | See Redirecting Traffic to Clone Data Center |
| Upgrade Oracle Access Manager on Master data center. | See Upgrading Oracle Access Manager on Master Data Center |
| Freeze all changes to the Master and Clones, if required. | See Freezing all Changes to Clones (if Necessary) |
| Sync the access UDM data by exporting the access store data from Master data center and importing it on the Clone data center. | See Syncing Access Metadata |
| Create the replication agreement again. | See Creating Replication Agreement |
| Upgrade the java.security file. | See Updating the java.security File |
| Bring up the Master and Clone data centers online. | See Bringing up the Master and Clone Data Centers Online |

# Backing Up the Existing MDC Environment

Before you begin with the upgrade, take a back up of your existing environment.

After stopping all the servers, you must back up the following on every data center before proceeding with the upgrade process:

- *ORACLE_HOME*: the Oracle Home directory.
- Oracle Access Manager Domain Home directory on all OAM hosts.
- Following Database schemas:
  - Oracle Access Manager schema
  - Audit and any other dependent schema

For more information about backing up schemas, see *Oracle Database Backup and Recovery User's Guide*.

# Enabling Write Permission to Master and Clones (If Necessary)

Before you start the upgrade, you must enable modifications to the system and policy configurations on both Master and Clones.

Complete the following:

1. Go to the *ORACLE_HOME*/common/bin directory.
   For example: /home/oracle/oam/ORACLE_IDM/common/bin
2. Run the following command on Master and Clone data centers:
   setMultiDataCenterWrite(WriteEnableFlag="true")

# Disabling and Deleting All Replication Agreements Between Master and Clone

Disable all replication agreements between the Master and the Clone data centers.

See Disabling Automated Policy Synchronization in the *Administrator's Guide for Oracle Access Manager*.

# Redirecting Traffic to Master Data Center

An in-line upgrade procedure is used to upgrade the Clone data center which requires downtime. Therefore, all traffic must be rerouted to the Master data center.

This is usually achieved by directing your load balancer to send all requests to the Master Site. Contact your network administrator to perform this task.

# Upgrading Oracle Access Manager on Clone Data Center

Upgrade Oracle Access Manager on Clone data center to 12*c* (12.2.1.3.0) after you redirect the traffic to Master data center.

To upgrade Oracle Access Manager on Master data center, follow the instructions described in Upgrading Oracle Access Manager Highly Available Environments.

# Redirecting Traffic to Clone Data Center

An in-line upgrade procedure is used to upgrade the Master data center which requires downtime. Therefore, all traffic must be rerouted to the Clone data centers (also referred to as, the backup data centers or the secondary data centers).

This is usually achieved by directing your load balancer to send all requests to the Master Site. Contact your network administrator to perform this task.

# Upgrading Oracle Access Manager on Master Data Center

Upgrade Oracle Access Manager on Master data center to 12*c* (12.2.1.3.0) after you redirect the traffic to clone data center.

To upgrade Oracle Access Manager on Master data center, follow the instructions described in Upgrading Oracle Access Manager Highly Available Environments.

# Freezing all Changes to Clones (if Necessary)

After you upgrade Oracle Access Manager on all of the Clone data center(s), it is recommended that you freeze the changes to the Clone data center(s). This is to avoid any inadvertent writes.

To freeze the changes, complete the following on the Clone data center(s):

1. Go to *ORACLE_HOME*/common/bin.
2. Run the following command:
   `SetMultiDataCenterWrite(WriteEnableFlag="false")`

# Syncing Access Metadata

Oracle Access Manager metadata stored in Unified Data Model (UDM) needs to be synced from Master to Clone.

You can sync the access metadata using the WLST commands - `exportAccessStore` and `importAccessStore`. These commands need to be executed after you upgrade all of the data centers and before creating the new replication agreement. This exports the UDM artifacts created till that point, from the Master data center and imports them in the Clone data center(s).
To sync the UDM metadata, complete the following steps:

1. Go to the *ORACLE_HOME*/common/bin directory.
2. Run the following WLST command on the Master data center to create a ZIP file containing the UDM metadata:

   ```
   exportAccessStore(toFile="/master/location/dc1metadata.zip",
   namePath="/")
   ```

3. Copy *dc1metadata.zip* to each of the upgraded Clone data centers.
4. Run the following WLST command on the each of the Clone data centers to import the UDM metadata:

   ```
   importAccessStore(fromFile="/clone/location/dc1metadata.zip",
   namePath="/")
   ```

# Creating Replication Agreement

Create the replication agreement again after upgrading the Master and the Clone data centers.

To create the replication agreement, run the following command:

> **Note:**
>
> Ensure that Master & Clone data centers REST endpoints are up and running, before you run this command.

```
curl -u <repluser> -H 'Content-Type: application/json' -X POST 'https://
supplier.example.com/oam/services/rest/_replication/setup' -d
'{"name":"DC12DC2", "source":"DC1","target":"DC2","documentType":"ENTITY"}'
```

For more information about creating a replication agreement, see Creating a Replication Agreement in the *Administrator's Guide for Oracle Access Manager*.

# Updating the java.security File

If you have multiple components of Oracle Identity and Access Management (Oracle Access Manager, Oracle Identity Manager, WebGates and so on) deployed, until you upgrade all of the components to 12*c* (12.2.1.3.0), you must update the `java.security` file with the changes described in this section.

To do this:

1. Open the `java.security` file located at *`JAVA_HOME`*`/jre/lib/security/` in an editor.
2. Remove `TLSv1`, `TLSv1.1`, `MD5withRSA` from the following key:

   ```
   key - jdk.tls.disabledAlgorithms
   ```

3. Remove `MD5` from the following key:

   ```
   key - jdk.certpath.disabledAlgorithms
   ```

For more information on possible upgrade scenarios, see Troubleshooting Security Policy Issues When Upgrading.

# Bringing up the Master and Clone Data Centers Online

After successful upgrade, both Master and Clone data centers can be brought up online. Traffic can be routed to both data centers based on existing routing rules.

Consult your network infrastructure team or refer to the network infrastructure documentation to accomplish the traffic re-routing.

6

# Upgrading OIM-OAM Integrated Environments set up Manually

You can upgrade Oracle Identity Manager (OIM), Oracle Access Manager (OAM) integrated split domain highly available environments that are set up manually, from 11*g* Release 2 (11.1.2.3.0) to 12*c* (12.2.1.3.0) using the upgrade procedure described in this section.

> **Note:**
>
> The product Oracle Identity Manager is referred to as Oracle Identity Manager (OIM) and Oracle Identity Governance (OIG) interchangeably in the guide.

**Topics**

- About the OIM-OAM Integrated HA Topology Set Up Manually
  The sample topology is based on the split domain four node topology described in the *Enterprise Deployment Guide for Oracle Identity and Access Management* 11*g* Release 2 (11.1.2.3.0), that is deployed manually.
- Supported Starting Points for Integrated HA Upgrade
  Review the supported starting points for each of the components in your integrated environment in order to upgrade to 12*c* (12.2.1.3.0). If the components are in earlier versions, upgrade them to the version that is supported for 12*c* upgrade.
- Roadmap for Upgrading OIM-OAM Integrated Highly Available Environments Set Up Manually
  Refer to the roadmap for upgrading Oracle Identity Manager and Oracle Access Manager integrated highly available 11.1.2.3.0 environments that was set up manually, to 12*c* (12.2.1.3.0).

## About the OIM-OAM Integrated HA Topology Set Up Manually

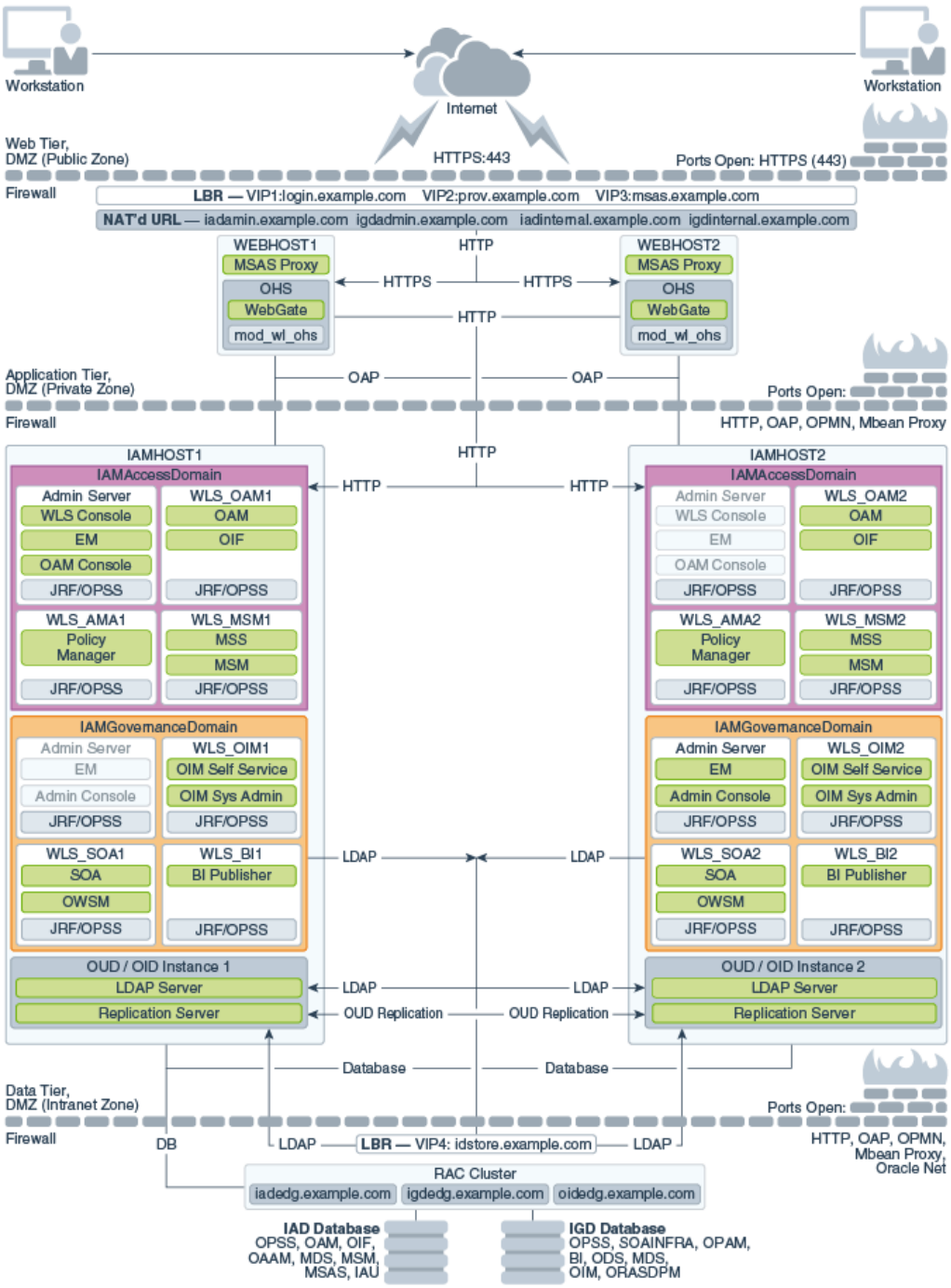
The sample topology is based on the split domain four node topology described in the *Enterprise Deployment Guide for Oracle Identity and Access Management* 11*g* Release 2 (11.1.2.3.0), that is deployed manually.

See Enterprise Deployment Guide for Oracle Identity and Access Management.

This topology and the accompanying procedures in this chapter are provided to serve as an example for upgrading a highly available, integrated Oracle Identity and Access Management environment. Your specific Oracle Identity and Access Management installation will vary, but this topology and upgrade procedure demonstrates the key elements of the upgrade process, which can be applied to your specific environment.

For a complete description of the topology diagram, refer to the Enterprise Deployment Guide for Oracle Identity and Access Management in the 11*g* Release 2 (11.1.2.3.0) Documentation Library.

**Figure 6-1 OIM-OAM Integrated Topology Set Up Manually**




# Supported Starting Points for Integrated HA Upgrade

Review the supported starting points for each of the components in your integrated environment in order to upgrade to 12*c* (12.2.1.3.0). If the components are in earlier versions, upgrade them to the version that is supported for 12*c* upgrade.

The following table lists the versions that are supported for upgrade of an integrated highly available environments.

**Table 6-1 Supported Starting Point for Integrated HA Upgrade**

| Component | Supported Starting Point |
|---|---|
| Oracle Identity Manager | 11*g* Release 2 (11.1.2.3.0) |
| Oracle Access Manager | 11*g* Release 2 (11.1.2.3.0) |
| Oracle SOA Suite | 11*g* Release 1 (11.1.1.9.0) |
| Oracle WebLogic Server | 10.3.6 |

Oracle Adaptive Access Manager is not part of the Oracle Identity and Access Management suite for 12*c* (12.2.1.3.0), and hence will not be upgraded to 12*c*. Oracle Adaptive Access Manager 11.1.2.3.0 is compatible with Oracle Access Manager 12*c* (12.2.1.3.0).

# Roadmap for Upgrading OIM-OAM Integrated Highly Available Environments Set Up Manually

Refer to the roadmap for upgrading Oracle Identity Manager and Oracle Access Manager integrated highly available 11.1.2.3.0 environments that was set up manually, to 12*c* (12.2.1.3.0).

The following table describes the tasks that you must perform to upgrade an OIM-OAM integrated topology described in About the OIM-OAM Integrated HA Topology Set Up Manually.

**Table 6-2 Tasks for Upgrading Integrated Environments Set Up Manually**

| Task | Documentation |
|---|---|
| Review the OIM-OAM integrated topology. | See About the OIM-OAM Integrated HA Topology Set Up Manually. |
| Review the supported starting points for integrated environment upgrade. | See Supported Starting Points for Integrated HA Upgrade. |
| Ensure that the LDAP server and the Oracle Access Manager have the same lockout value configured before you start the upgrade. That is, the lockout threshold of libOVD, OAM, and LDAP should be the same, else the lock and unlock use cases fail after upgrade.<br>This is applicable for a OIM-OAM integrated single node setup as well. | See Setting the LockoutThreshold in Active Directory in the *Oracle Fusion Middleware Deployment Guide for Oracle Identity and Access Management* for 11*g* Release 2 (11.1.2.3.0). |
| If you have configured Node Manager, ensure that the Node Manager is stopped before you proceed with the upgrade. | See Stopping Servers and Processes. |

**Table 6-2 (Cont.) Tasks for Upgrading Integrated Environments Set Up Manually**

| Task | Documentation |
|---|---|
| Check if Oracle Access Manager (OAM) is integrated with Oracle Identity Manager (OIM) in a single domain<br>If Oracle Access Manager is integrated with Oracle Identity Manager (OIM), and if both the products are in a same domain, a separate OAM domain needs to be cloned that works with OIM in the source domain. It is the cloned OAM domain that needs to be upgraded to 12c. | See Checking if OAM is in a Different Domain to OAAM and OIM.<br>Also, complete any necessary pre-upgrade tasks for Oracle Access Manager.<br>See Completing the Pre-Upgrade Tasks for Oracle Access Manager. |
| Upgrade Oracle Access Manager to 12*c* (12.2.1.3.0).<br>You can choose to upgrade Oracle Identity Manager first too. | See Upgrading Oracle Access Manager Highly Available Environments. |
| Upgrade Oracle Identity Manager to 12*c* (12.2.1.3.0). | See Upgrading Oracle Identity Manager Highly Available Environments. |

**Note:**

If you encounter any issues during upgrade, see Troubleshooting the Oracle Access Manager Upgrade.

7

# Upgrading OIM-OAM Integrated Environments set up Using Life Cycle Management Tool

If you had set up an Oracle Identity Manager – Oracle Access Manager integrated environment in 11*g* Release 2 (11.1.2.3.0) using the Life Cycle Management (LCM) tool, follow the instructions in this chapter to upgrade the same to 12*c* (12.2.1.3.0).

> **Note:**
>
> The product Oracle Identity Manager is referred to as Oracle Identity Manager (OIM) and Oracle Identity Governance (OIG) interchangeably in the guide.

**Topics**



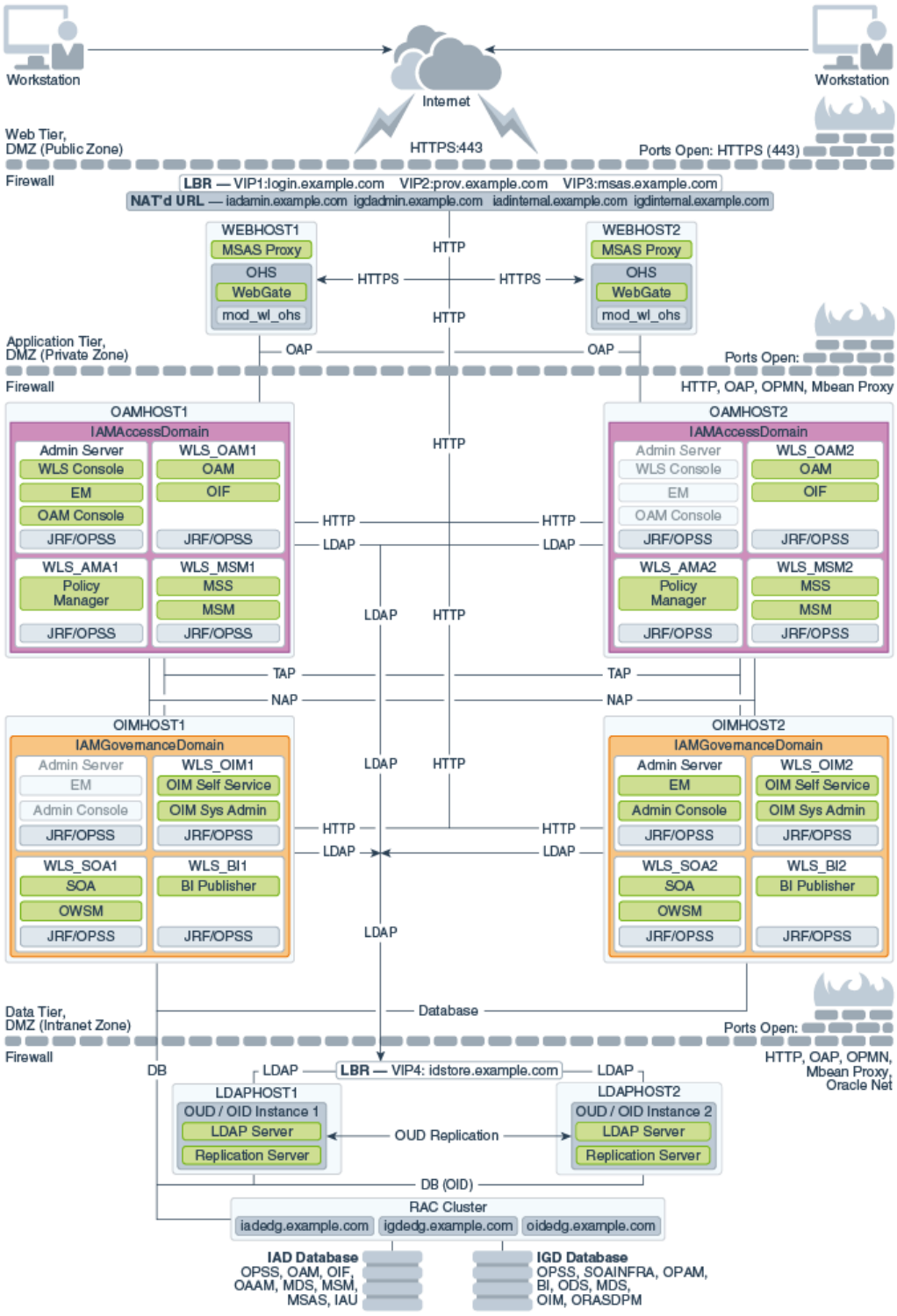
## About the OIM-OAM Integrated HA Topology Set Up Using LCM Tool

The sample topology is based on the split domain eight node topology described in the *Enterprise Deployment Guide for Oracle Identity and Access Management* 11*g* Release 2 (11.1.2.3.0), that is deployed using the Life Cycle Management (LCM) tool.

This topology and the accompanying procedures in this chapter are provided to serve as an example for upgrading a highly available, integrated Oracle Identity and Access Management environment. Your specific Oracle Identity and Access Management installation will vary, but this topology and upgrade procedure demonstrates the key elements of the upgrade process, which can be applied to your specific environment.

For a complete description of the topology diagram, refer to the *Enterprise Deployment Guide in the Oracle Identity and Access Management* in the 11*g* Release 2 (11.1.2.3.0) Documentation Library.

**Figure 7-1 OIM-OAM Integrated Topology Set Up Using LCM Tool**



# Supported Starting Points

Review the supported starting points for each of the components in your integrated environment in order to upgrade to 12*c* (12.2.1.3.0). If the components are in earlier versions, upgrade them to the version that is supported for 12*c* upgrade.

The following table lists the versions that are supported for upgrade of an integrated highly available environments.

**Table 7-1 Supported Starting Points for Integrated HA Upgrade**

| Component | Supported Starting Point |
|---|---|
| Oracle Identity Manager | 11*g* Release 2 (11.1.2.3.0) |
| Oracle Access Manager | 11*g* Release 2 (11.1.2.3.0) |
| Oracle SOA Suite | 11*g* Release 1 (11.1.1.9.0) |
| Oracle WebLogic Server | 10.3.6 |

Oracle Adaptive Access Manager is not part of the Oracle Identity and Access Management suite for 12*c* (12.2.1.3.0), and hence will not be upgraded to 12*c*. Oracle Adaptive Access Manager 11.1.2.3.0 is compatible with Oracle Access Manager 12*c* (12.2.1.3.0).

# Roadmap for Upgrading OIM-OAM Integrated Environments set up Using Life Cycle Management Tool

Refer to the roadmap in this section for upgrading Oracle Identity Manager and Oracle Access Manager integrated highly available 11.1.2.3.0 environments, set up using Life Cycle Management (LCM) tool, to 12*c* (12.2.1.3.0).

The following table describes the tasks that you must perform to upgrade an integrated topology described in About the OIM-OAM Integrated HA Topology Set Up Using LCM Tool.

**Table 7-2 Tasks for Upgrading Integrated Environments Set Up Using LCM Tool**

| Task | Documentation |
|---|---|
| Review the OIM-OAM integrated topology. | See About the OIM-OAM Integrated HA Topology Set Up Using LCM Tool. |
| Review the supported starting points for integrated environment upgrade. | See Supported Starting Points. |
| Ensure that the LDAP server and the Oracle Access Manager have the same lockout value configured before you start the upgrade. That is, the lockout threshold of libOVD, OAM, and LDAP should be the same, else the lock and unlock use cases fail after upgrade.<br>This is applicable for a OIM-OAM integrated single node setup as well. | See Setting the LockoutThreshold in Active Directory in the *Oracle Fusion Middleware Deployment Guide for Oracle Identity and Access Management* for 11*g* Release 2 (11.1.2.3.0). |
| If you have configured Node Manager, ensure that the Node Manager is stopped before you proceed with the upgrade. | See Stopping Servers and Processes. |

**Table 7-2 (Cont.) Tasks for Upgrading Integrated Environments Set Up Using LCM Tool**

| Task | Documentation |
|---|---|
| Check if Oracle Access Manager (OAM) is integrated with Oracle Identity Manager (OIM) in a single domain<br>If Oracle Access Manager is integrated with Oracle Identity Manager (OIM), and if both the products are in a same domain, a separate OAM domain needs to be cloned that works with OIM in the source domain. It is the cloned OAM domain that needs to be upgraded to 12c. | See Checking if OAM is in a Different Domain to OAAM and OIM.<br>Also, complete any necessary pre-upgrade tasks for Oracle Access Manager.<br>See Completing the Pre-Upgrade Tasks for Oracle Access Manager. |
| Upgrade the Oracle Identity Manager on OIMHOST1 shared domain to 12*c* (12.2.1.3.0).<br>Do NOT start the servers after you upgrade. | See Upgrading Oracle Identity Manager Single Node Environments. |
| Take a backup and delete the contents of the private domain.<br>It is recommended that you perform this step, or the `soa-infra` application continues to be in `Prepared` state instead of `active` state, post upgrade. | See soa-infra Application is in 'Prepared' State Post Upgrade. |
| Pack the Oracle Identity Manager shared domain and unpack it into the private domain on OIMHOST1 and OIMHOST2. | See Replicating the Domain Configurations on OIMHOST2.<br>**Note:** Use the pack and unpack commands as described in the above section to pack the OIM shared domain and unpack it to the private domain on OIMHOST1 and OIMHOST2. |
| Start the Oracle SOA Suite Managed Servers and Oracle Identity Manager servers on OIMHOST1.<br>You must start the Administration Server from the shared domain and the Managed Servers from the private domain.<br>When you start the Oracle SOA Suite Managed Servers for the first time after upgrade, ensure that you do it with Business Process Management (BPM) property and Administration Server URL. | See, Starting the Servers.<br>If bootstrapping fails when you start the Oracle Identity Manager servers for the first time, follow the instructions described in OIM Bootstrap for DEPLOYSOACOMPOSITES Task Fails After Upgrade to resolve this, and then start the servers. |
| Restart the Administration Server, Oracle SOA Suite Managed Servers, and the Oracle Identity Manager Managed Servers on OIMHOST1 and start the Oracle SOA Suite Managed Servers, and the Oracle Identity Manager Managed Servers on OIMHOST2.<br>When you restart the Oracle SOA Suite Managed Servers for the second time after upgrade, ensure that you do it without Business Process Management (BPM) property. | See Stopping Servers and Processes for stopping the servers.<br>See, Starting the Servers for starting the servers. |
| Upgrade the Oracle Access Manager on OAMHOST1 shared domain to 12*c* (12.2.1.3.0).<br>Do not start the servers after you upgrade. | See Upgrading Oracle Access Manager Single Node Environments. |

**Table 7-2 (Cont.) Tasks for Upgrading Integrated Environments Set Up Using LCM Tool**

| Task | Documentation |
|---|---|
| Pack the Oracle Access Manager shared domain and unpack it to the private domain on OAMHOST1 and OAMHOST2. | See Replicating the Domain Configurations on each OAMHOST.<br>**Note:** Use the pack and unpack commands as described in the above section to pack the OAM shared domain and unpack it to the private domain on OAMHOST1 and OAMHOST2. |
| Start the Administration Server and the Oracle Access Manager Managed Servers. | See Starting Servers and Processes . |

> **Note:**
>
> If you encounter any issues when upgrading Oracle Access Manager, see Troubleshooting the Oracle Access Manager Upgrade for troubleshooting tips.

# Part II

# Out-of-Place Cloned Upgrade of Oracle Access Manager

An out-of-place cloned upgrade is described as creating a copy of your existing system on a new hardware, and then performing an in-place upgrade on the clone.

This part contains the following chapter:

- Cloning Oracle Access Manager Environment
  The out-of-place upgrade procedure discussed in this guide explains how to perform a cloned upgrade of Oracle Access Manager 11g to Oracle Access Manager 12c.

8

# Cloning Oracle Access Manager Environment

The out-of-place upgrade procedure discussed in this guide explains how to perform a cloned upgrade of Oracle Access Manager 11g to Oracle Access Manager 12c.

This chapter includes the following topics:



## Cloning the Database

You can take a copy of your existing environment and then upgrade that copy. If you encounter issues during the upgrade, you will have the existing environment as a fallback.

For more information, see Performing an Upgrade via a Cloned Environment.



### Methods for Cloning Databases

There are different methods of cloning a database and each method has its own merits.

> **Note:**
>
> Oracle Identity and Access Management 12c does not support Oracle Access Manager and Oracle Identity Manager configured to use the same database schema prefix. Before you upgrade, if both products co-exist and share the same database schemas, you must first split the database into two different prefixes and schema sets.

You can use the following options to clone the database:

**Option 1 – Database Export Import**

- Suitable for smaller sized databases.
- Allows movement between versions. For example, 12.1.0.3 to 19c.
- Allows movement into Container Databases/Private Databases.
- Is a complete copy; redoing the exercise requires data to be deleted from the target each time.

- No ongoing synchronization.
- During cut-over the source system will need to be frozen for updates.

**Option 2 – Duplicate Database Using RMAN**

- Suitable for databases of any size.
- Takes a back up of an entire database.
- The database version and patch level should be the same on both the source and destination.
- Database upgrades will need to be performed as a separate task.
- CDP/PDB migration will have to be done as a separate exercise.
- No ongoing synchronization.
- During cut-over, you should freeze the source system for updates.

**Option 3 – Dataguard Database**

- Suitable for databases of any size.
- Takes a back up of an entire database.
- Database upgrades will need to be performed as a separate task.
- CDP/PDB migration will have to be done as a separate exercise.
- Ongoing synchronisation; Database can be opened to test the upgrade and closed again to keep data synchronized with the source system.

> **Note:**
>
> You should choose the solution based on your requirements.

## Cloning the Database Using the Export/Import Method

On your 11g environment, export the data from your database to an export file.

To export the data, do the following:

1. Install an Oracle database of the version you want to use. This database can be a single instance database, a real applications cluster (RAC) database, a standard database, or a container database with OAM in a separate pluggable databse (PDB).
2. Make a directory on the source and the destination target hosts.
3. Create a database directory object pointing to this location on the source and destination databases.
4. Export the source database.

> **Note:**
>
> If you are using a RAC database, make sure you have a TNS connection which is forced to a specific instance/PDB unless you want to create the directories on each node. IADUPG is an example of a RCU prefix.

5. Copy the generated file to the destination database host.
6. Extract DDL from the source database. The import will only import the data you have extracted from the source database, it will not create any tablespaces or users, and not having those present will cause the import to fail. This can be resolved by extracting the DDL for these objects from the database. To do this:
   a. Create a file called extract_ddl.sql using an editor of your choice, with the following content:

```
set pages 0
set feedback off
set heading off
set long 5000
set longchunksize 5000
set lines 200
set verify off
exec dbms_metadata.set_transform_param
(dbms_metadata.session_transform, 'SQLTERMINATOR', true);
exec dbms_metadata.set_transform_param
(dbms_metadata.session_transform, 'PRETTY', true);
accept PREFIX char prompt 'Enter RCU Prefix:'
accept PDBNAME char prompt 'Enter PDB:'
spool ddl.sql
select 'alter session set container=&&PDBNAME;'
from dual
/
SELECT DBMS_METADATA.GET_DDL('TABLESPACE',Tablespace_name)
from  dba_tablespaces
where tablespace_name like '&&PREFIX%'
/
set lines 600
SELECT DBMS_METADATA.GET_DDL('USER',USERNAME)
from DBA_USERS
where USERNAME like '&&PREFIX%'
/
set lines 200
SELECT DBMS_METADATA.GET_GRANTED_DDL ('SYSTEM_GRANT',USERNAME)

from DBA_USERS
where USERNAME like '&&PREFIX%'
and USERNAME NOT LIKE '%_IAU_APPEND'
and USERNAME NOT LIKE '%_IAU_VIEWER'
/
SELECT DBMS_METADATA.GET_GRANTED_DDL ('OBJECT_GRANT',USERNAME)

from DBA_USERS
where USERNAME like '&&PREFIX%'
and USERNAME NOT LIKE '%TLOGS'
```

```
and USERNAME NOT LIKE '%JMS'
/

spool off
set pages 0
set feedback off
set heading off
set long 5000
set longchunksize 5000
set lines 200
set verify off
exec dbms_metadata.set_transform_param
(dbms_metadata.session_transform, 'SQLTERMINATOR', true);
exec dbms_metadata.set_transform_param
(dbms_metadata.session_transform, 'PRETTY', true);
accept PREFIX char prompt 'Enter RCU Prefix:'
accept PDBNAME char prompt 'Enter PDB:'
spool ddl.sql
select 'alter session set container=&&PDBNAME;'
from dual
/
SELECT DBMS_METADATA.GET_DDL('TABLESPACE',Tablespace_name)
from  dba_tablespaces
where tablespace_name like '&&PREFIX%'
/
set lines 600
SELECT DBMS_METADATA.GET_DDL('USER',USERNAME)
from DBA_USERS
where USERNAME like '&&PREFIX%'
/
set lines 200
SELECT DBMS_METADATA.GET_GRANTED_DDL ('SYSTEM_GRANT',USERNAME)

from DBA_USERS
where USERNAME like '&&PREFIX%'
and USERNAME NOT LIKE '%_IAU_APPEND'
and USERNAME NOT LIKE '%_IAU_VIEWER'
/
SELECT DBMS_METADATA.GET_GRANTED_DDL ('OBJECT_GRANT',USERNAME)

from DBA_USERS
where USERNAME like '&&PREFIX%'
and USERNAME NOT LIKE '%TLOGS'
and USERNAME NOT LIKE '%JMS'
/

spool off
```

> **Note:**
>
> The lines in **Bold** are applicable only if your target database is a PDB. This SQL assumes that all the objects are created using the RCU prefix. If you have created objects without the prefix (for example tablespaces/users for JMS or TLogs), you will need to add these manually.

b. Execute the file in SQL Plus:

```
SQL> @extract_ddl
```

7. Copy the generated file to the destination database host.
8. Create TNS entry for the Pluggable Database in the target system, if necessary.
9. Validate that the target database meets all of the criteria of Oracle Access Manager. See Installing and Configuring the Oracle Access Management Software.
10. Create a database restore point to roll back the transaction, if required.
11. Create the Tablespaces/Users for Oracle Access Manager.

    To do this execute the script (`ddl.sql`) you generated earlier (in step 6).

    Execute the file in SQL Plus:

    ```
    SQL> @ddl
    ```

    Carefully review the output and correct errors, if any.
12. Import the data into the destination database. This database need not be at the same database version as the source.

    ```
    export ORACLE_BASE=/u01/app/oracle
    export ORACLE_HOME=${ORACLE_BASE}/product/12.2.0.1/dbhome_1
    export GRID_HOME=/u01/app/12.2.0.1/grid
    export PATH=$PATH:$ORACLE_HOME/bin:$ORACLE_HOME/OPatch
    export DB_NAME=iamcdb_phx1g8
    export ORACLE_SID=iamcdb

    impdp \"SYS/Password@IADPDB AS SYSDBA\" DIRECTORY=orcl_full
    DUMPFILE=oam_system.dmp LOGFILE=oam_system_imp.log FULL=YES;
    impdp \"SYS/Password@IADPDB AS SYSDBA\" DIRECTORY=orcl_full
    DUMPFILE=full_oam.dmp LOGFILE=full_oam_imp.log FULL=YES;
    ```
13. Create a database service in the target system with the same name as the primary.

    ```
    srvctl add service -db iamcdb_phx1g8 -service onpremservice -rlbgoal
    SERVICE_TIME -clbgoal SHORT -pdb iadpdb
    srvctl start service -db iamcdb_phx1g8 -service onpremservice
    srvctl status service -db iamcdb_phx1g8 -service onpremservice
    ```

    After you have imported the schemas, it is important to check that the following query returns rows that are consistent with your deployment. This table should have been

imported as part of the steps above. If it fails to do so, you must populate the table with values from your source system.

```
set linesize 100
col comp_id for a10
col comp_name for a50
col version for a10
select comp_id, comp_name, version, status, upgraded from
system.schema_version_registry;
```

## Cloning the Database Using RMAN

Clone the database from the source environment to the target environment by using RMAN. See Transferring Data with RMAN.

# Cloning the Oracle Binaries

Following options are available for cloning the Oracle binaries:

- Using your preferred backup/restore tools to archive and transfer the MW_HOME binaries and OraInventory directories.
- Using the Oracle FMW T2P process.

This section includes the following topics:

- Using Backup/Restore Tools to Clone the Access Domain
- Cloning the Oracle Binaries Using T2P

## Using Backup/Restore Tools to Clone the Access Domain

> **Note:**
>
> You can take a back up with the domain and NodeManagers online or offline. However, Oracle recommends to execute the backup with all FMW processes shutdown.

**Take a backup:**

Complete the following steps to take a backup of your source environment binaries and Oracle Inventory:

1. Using your preferred backup tool, take a backup of the following locations on the source site:
   - `oraInventory`
   - `MW_HOME`

For example, a command on `OAMHOST1` may appear as follows:

```
tar cfzP /u01/oracle/backups/oamhost1_binaries.tar.gz /u01/oracle/
oraInventory MW_HOME
```

2. Repeat the command on any supplementary nodes using the separate product binary volumes.

> **Note:**
>
> When using the shared filesystem volumes for the Oracle products `MW_HOME` locations, you should **take** only the binary backups from one host per volume.

For example, a command on `OAMHOST2` may appear as follows:

```
tar cfzP /u01/oracle/backups/oamhost2_binaries.tar.gz /u01/oracle/
oraInventory MW_HOME
```

3. Copy the resulting backup files to their appropriate target environment hosts.

**Restore the backup**

Using your preferred extraction tool, extract the backup to your target environment nodes.

> **Note:**
>
> When using the shared filesystem volumes for the Oracle products `MW_HOME` locations, you should **restore** only the binary backups to one host per volume.

For example:

On OAMHOST1, run the following command:

```
tar xvfzP oamhost1.tar.gz
```

On OAMHOST2, run the following command:

```
tar xvfzP oamhost2.tar.gz
```

## Cloning the Oracle Binaries Using T2P

You can use this method as an alternative to the backup/restore method.

Move a copy of the Middleware home for the component or suite from the source environment to the target environment using the `copyBinary` and `pasteBinary` scripts. See Moving the Middleware Home and the Binary Files.

# Cloning the Configuration

Following options are available for cloning the configuration:

- Using your preferred backup/restore tools to clone the configuration.
- Using the T2P process.
- Using Backup/Restore Tools to Clone the Access Domain
- Cloning the Configuration Using T2P
- Starting the OAM Domain

# Using Backup/Restore Tools to Clone the Access Domain

> **Note:**
>
> You can take a back up with the domain and Node Managers online or offline. However, Oracle recommends to execute the backup with all FMW processes shutdown.

**Take a backup:**

Following steps are available to take a backup of the source environment binaries and Oracle Inventory:

1. Using your preferred backup tool, take a backup of the following locations on the source site:
   - Application Server domain home (`ASERVER_HOME`)
   - Managed Server domain home if you have a separate location as described in the EDG (`MSERVER_HOME`)
   - `Keystores`
   - `Nodemanager`

   > **Note:**
   >
   > If you have a combined `DOMAIN_HOME` rather than a segregated one, as described in the Enterprise Deployment Guide, include `DOMAIN_HOME` rather than `ASERVER_HOME` and `MSERVER_HOME`.

   For example, a command on `OAMHOST1` may appear as follows:

```
tar cfvzP /u01/oracle/config/backups/oamhost1_accessdomain.tar.gz \
   ASERVER_HOME \
   MSERVER_HOME \
   /u01/oracle/config/keystores \
   /u01/oracle/config/nodemanager/OAMHOST1 \
   /u01/oracle/config/nodemanager/OAMHOST2 \
   /u01/oracle/config/nodemanager/IADADMINVHN \
   /u01/oracle/runtime/domains/IAMAccessDomain
```

2. Repeat the command on any supplementary nodes. For example, a command on `OAMHOST2` may appear as follows:

```
tar cfzP /u01/oracle/backups/oamhost2_accessdomain.tar.gz /u02/private/
oracle/config/domains/IAMAccessDomain
```

3. Copy the resulting backup files to their appropriate target environment hosts.
4. Delete any lock and log files in the domain that have been replicated from the source environment.
   - Remove any lock files for all `NodeManager` folders on the appropriate cloned environment hosts by running the following command:

     ```
     find /u01/oracle/config/nodemanager -type f -name "*.lck" -exec rm -f {}
     \;
     ```

   - Remove any lock files from the `ASERVER_HOME` and `MSERVER_HOME` folders on the appropriate cloned environment hosts by running the following command:

     > **Note:**
     >
     > If you have a combined `DOMAIN_HOME` rather than a segregated one as described in the Enterprise Deployment Guide, include `DOMAIN_HOME` rather than `ASERVER_HOME` and `MSERVER_HOME`.

     For example, on `OAMHOST1`, run the following command:

     ```
     find  ASERVER_HOME \
         -type f \( -name "*.lck" -or -name "*.lok" \) -print -exec rm -f
     {} \;
     find  MSERVER_HOME \
         -type f \( -name "*.lck" -or -name "*.lok" \) -print -exec rm -f
     {} \;
     ```

     For example, on `OAMHOST2`, run the following command:

     ```
     find  MSERVER_HOME \
         -type f \( -name "*.lck" -or -name "*.lok" \) -print -exec rm -f
     {} \;
     ```

   - Optionally, remove the old log files from the `NodeManager` and Managed Server folders in the cloned domain:

     For example, on `OAMHOST1`, run the following command:

     ```
     find /u01/oracle/config/nodemanager/OIMHOST1 \
         -type f \( -name '*.log' -or -name '*.out' \) -print -exec rm -f
     {} \;
     find /u01/oracle/config/nodemanager/OIMHOST2 \
         -type f \( -name '*.log' -or -name '*.out' \) -print -exec rm -f
     {} \;
     ```

```
find /u01/oracle/config/nodemanager/IGDADMINVHN \
    -type f \( -name '*.log' -or -name '*.out' \) -print -exec
rm -f {} \;

find ASERVER_HOME/servers/AdminServer/logs \
    -type f ! -size 0c -print -exec rm -f {} \+

find MSERVER_HOME/servers/*/logs \
    -type f ! -size 0c -print -exec rm -f {} \+
```

For example, on `OAMHOST2`, run the following command:

```
find MSERVER_HOME/servers/*/logs \ -type f ! -size 0c -print -exec
rm -f {} \+
```

**Restore the Access Domain in the Cloned Environment**

Using your preferred extraction tool, extract the backup to your target environment nodes.

For example:

On OAMHOST1, run the following command:

```
tar xvfzP oamhost1_accessdomain.tar.gz
```

On OAMHOST2, run the following command:

```
tar xvfzP oamhost2_accessdomain.tar.gz
```

# Cloning the Configuration Using T2P

You can clone the configuration using the T2P method. This method is an alternative to the backup/recovery option. The advantage of using T2P is that it enables you to change the host names during the process. You can move a copy of the configuration of components such as UMS (User Messaging Service) messaging preferences, Oracle Identity Management configuration files, and so on, by using the following scripts:

- `copyConfig`
- `extractMovePlan`
- `pasteConfig`

To modify the host name or ports that is specific to the new environment, see Moving Oracle Fusion Middleware Components

> **Note:**
>
> Before running `pasteConfig` on the target environment, connect to the cloned database and verify that all the schemas/data from the source environment are present.

## Starting the OAM Domain

After successfully restoring the backup to the target environment instances, do the following to start the domain:

- Start the Node Manager for the Administration Server.
- Start the Node Manager for the Managed Servers (if different).
- Start the administration server.
- Start the OAM managed servers.
- Start the policy manager managed servers.

# Upgrading the Cloned Environment

After cloning the environment, you should perform some sanity checks to ensure that it is working as desired. After verifying that the environment is functioning as expected, take a backup of the environment and perform the upgrade as described in In-Place Upgrade of Oracle Access Manager.

A

# Troubleshooting the Oracle Access Manager Upgrade

If you encounter errors while upgrading Oracle Access Manager to 12c (12.2.1.3), review the following troubleshooting procedures.

- WebGates Configuration Fails during Authentication
  During authentication, WebGates configured with the user-defined parameter `hmacEnabled=true` in environments where the server-side `globalHMACEnabled` property is set to `false`, fails.
- Activation State is set as FAILED when Restarting the Admin Server
  After you upgrade the domain component configurations and start the Admin server, the activation state is set as `FAILED`.
- AMInitServlet Fails to Preload when Restarting OAM Managed Server
  After you upgrade the domain component configurations and start the OAM managed server, `AMInitServlet` fails to preload.
- CFGFWK-60928: Invalid Existing Node Manager Home Directory
  When you reconfigure the domain, an invalid existing node manager home directory error is displayed.
- File Not Found Exception when Starting the OAM Managed Server
  After you upgrade the domain component configurations and start the server a `File Not Found` exception is displayed.
- Internal Server Error: The Server Encountered an Unknown Error
  When you perform an upgrade and access a protected resource, an Internal Server Error is displayed.
- Invalid OAM Keystore Configuration: oam_admin Fails
  After you upgrade the domain component configurations and start the admin server, the `oam_admin` deployment state is displayed as `fail`.
- Upgrade Assistant Readiness Check Fails: Common Infrastructure Services (DEV_STB)
  When you reconfigure the domain, the upgrade assistant readiness check fails because of the common infrastructure services (`DEV_STB`).
- Upgrade Assistant Readiness Check Fails: Missing System and Object Privileges
  When you reconfigure the domain, the upgrade assistant readiness check fails because of missing system and object privileges for OPSS Schema (`DEV_IAU`).
- Upgrade Assistant Readiness Check Fails: Oracle WSM Datasource Connection Details
  When you reconfigure the domain, the upgrade assistant readiness check fails with the exception `Failed to read the Oracle WSM datasource connection details.`
- Readiness Check for OAM Configuration Upgrade Fails
  Before you run the readiness check for Oracle Access Manager for the first time, ensure that you have removed the IAMSuiteAgent security provider.
- Error When Starting SSL Enabled OAM Managed Server After Upgrade
  If SSL is enabled for Oracle Access Manager Managed Servers, the SSL port for the Administration Server must be changed manually before starting the servers.

- Readiness Check for OPSS Schema Fails
  When you upgrade Oracle Access Manager 11.1.2.3.0 environments that is upgraded from 11*g* Release 2 (11.1.2.1.0), the readiness check for Oracle Platform Security Services (OPSS) schema fails with the following exception"
- OAM Upgrade Fails With InvalidKeyException
  Oracle Access Manager upgrade fails with InvalidKeyException if Java JSE Policy is not upgraded.
- OWSM Error Messages in the Reconfiguration Logs
  During the Oracle Access Manager (OAM) upgrade, when you reconfigure the OAM domain, Oracle Web Services Manager (OWSM) error messages are seen in the reconfig logs.
- OAM Console Shows No Application Domains After Upgrade
  After you upgrade Oracle Access Manager (OAM) in an integrated setup where you have deployed Oracle Identity Manager, Oracle Access Manager, Oracle Unified Directory, and Oracle Adaptive Access Manager, when you search for application domains on OAM console, it shows no result.
- Troubleshooting Security Policy Issues When Upgrading

# WebGates Configuration Fails during Authentication

During authentication, WebGates configured with the user-defined parameter `hmacEnabled=true` in environments where the server-side `globalHMACEnabled` property is set to `false`, fails.

To solve the issue, apply patch 12.2.1.3.180414 or later and secure the communication between WebGates and the OAM server. After you apply the patch, ensure that `globalHMACEnabled` is set to `true`. This setting ensures that the OAM server rejects requests coming from the unpatched WebGates.

# Activation State is set as `FAILED` when Restarting the Admin Server

After you upgrade the domain component configurations and start the Admin server, the activation state is set as `FAILED`.

```
Caused By: oracle.security.am.install.AMInstallException: Invalid
Simple
Mode Artifacts at
oracle.security.am.install.startup.AMKeyStoreValidator.execute(AMKeySto
reValid
ator.java:70)at
oracle.security.am.install.startup.OamInstallTopologyConfigListener.doM
andator
yValidations(OamInstallTopologyConfigListener.java:114)
```

To solve the error, complete the following steps:

1. In the 11g environment, open to the `oam-config.xml` file and copy the value of `sslGlobalPassphrase`.

2. In the 12c environment, open to the `oam-config.xml` file and replace the value of `sslGlobalPassphrase` with the value that you copied from the 11g environment.

For more information about how to import or export `oam-config.xml` from database, see Doc ID 2310234.1.

## `AMInitServlet` Fails to Preload when Restarting OAM Managed Server

After you upgrade the domain component configurations and start the OAM managed server, `AMInitServlet` fails to preload.

The following error message is displayed:

```
Caused By: oracle.security.am.common.utilities.exception.AmRuntimeException:
Fail to decrypt oamkeystore data with cipher key from OAM config
(/DeployedComponent/Server/NGAMServer/Profile/ssoengine/CipherKey)
at oracle.security.am.engines.sso.adapter.OAMSessionConfiguration$Config
Listener.configurationChanged(OAMSessionConfiguration.java:295)
```

To solve the error, complete the following steps:

1. In the 11g environment, open to the `oam-config.xml` file and copy the value of `cipherKey`.
2. In the 12c environment, open to the `oam-config.xml` file and replace the value of `cipherKey` with the value that you copied from the 11g environment.

For more information about how to import or export `oam-config.xml` from database, see Doc ID 2310234.1.

## CFGFWK-60928: Invalid Existing Node Manager Home Directory

When you reconfigure the domain, an invalid existing node manager home directory error is displayed.

The following message is displayed:

```
CFGFWK-60928: Invalid existing node manager home directory.
Existing node manager home directory does not exist or is not accessible.
```

To solve the issue, select **Create New Configuration** in the Node Manager screen.

# File Not Found Exception when Starting the OAM Managed Server

After you upgrade the domain component configurations and start the server a `File Not Found` exception is displayed.

This is a known issue. Ignore the following `File Not Found` exception:

```
[2019-09-04T05:52:24.349+00:00] [wls_oam1] [WARNING] [J2EE JMX-46714]
[oracle.as.jmx.framework.wls.spi.ComponentMBeans] [tid:
[ACTIVE].ExecuteThread: '4' for queue: 'weblogic.kernel.Default
(self-tuning)'] [userId: <WLS Kernel>] [ecid:
ab946520-e9e8-498c-89f6-5e9e0f055f40-00000007,0] [partition-name:
DOMAIN]
[tenant-name: GLOBAL] Error parsing MBean descriptor file
"fmwconfig/mbeans/oamconfig_mbeans.xml".[[
java.io.FileNotFoundException: The Config MBean jar file
"C:\Oracle\Middleware_IAM\user_projects\domains\oam_domain\config\fmwco
nfig\mb
eans\${OAM_ORACLE_HOME}\server\lib\jmx\configmgmt.jar" does not exist.

[2019-09-04T05:52:26.693+00:00] [wls_oam1] [WARNING] [J2EE JMX-46714]
[oracle.as.jmx.framework.wls.spi.ComponentMBeans] [tid:
[ACTIVE].ExecuteThread: '4' for queue: 'weblogic.kernel.Default
(self-tuning)'] [userId: <WLS Kernel>] [ecid:
ab946520-e9e8-498c-89f6-5e9e0f055f40-00000007,0] [partition-name:
DOMAIN]
[tenant-name: GLOBAL] Error parsing MBean descriptor file
"fmwconfig/mbeans/t2p_mbeans.xml".[[
java.io.FileNotFoundException: The Config MBean jar file
"C:\Oracle\Middleware_IAM\user_projects\domains\oam_domain\config\fmwco
nfig\mb
eans\${OAM_ORACLE_HOME}\server\lib\jmx\was-t2p.jar" does not exist.
```

# Internal Server Error: The Server Encountered an Unknown Error

When you perform an upgrade and access a protected resource, an Internal Server Error is displayed.

When accessing a protected resource, the following error is displayed due to incorrect OAM datasource name:

```
Internal Server Error. The server encountered an unknown error,
possibly due
to misconfiguration.
Contact the server administrator:[no address given]
```

To solve the issue, complete the following steps:

1. Validate OAM instance name, host, port, and proxy port. If incorrect values are displayed, then update the values.
2. Validate LBR host and port. If incorrect values are displayed, then update the values.
3. Delete OAM admin and managed server cache.
4. Restart the admin and managed servers.
5. Copy the artifacts to webgate and restart the OHS instance.

# Invalid OAM Keystore Configuration: `oam_admin` Fails

After you upgrade the domain component configurations and start the admin server, the `oam_admin` deployment state is displayed as `fail`.

This configuration change is applicable only if all the following are true:

- `oam_admin` deployment state is displayed as `fail`.
- **ProductRelease** version in `oam-config`, stored in db, is not updated to 12.2.1.3.0.
- OAM admin logs does not contain the following info level log message:

```
First Time startup after upgrade detected. Starting upgrade.
```

To solve the issue, complete the following steps:

1. Go to the `upgradeStatus.properties` file located at *<domain_home>*`/config/fmwconfig/`.
2. Change the `isUpgrade` value to `True`.
3. Save and close the file.

# Upgrade Assistant Readiness Check Fails: Common Infrastructure Services (DEV_STB)

When you reconfigure the domain, the upgrade assistant readiness check fails because of the common infrastructure services (`DEV_STB`).

The following error message is displayed:

```
STBUPG-05610: Exception thrown while testing the readiness of
cie schema upgrade :java.lang.NullPointerException
```

To solve this issue, apply Patch 22096827. You can download this patch from My Oracle Support.

# Upgrade Assistant Readiness Check Fails: Missing System and Object Privileges

When you reconfigure the domain, the upgrade assistant readiness check fails because of missing system and object privileges for OPSS Schema (`DEV_IAU`).

To solve the issue, complete the following steps:

1. Provide the `SCHEMA_VERSION_REGISTRY` grant to following audit schemas:
   - `DEV_IAU`
   - `DEV_IAU_APPEND`
   - `DEV_IAU_VIEWER`

   For example: `GRANT SELECT ON "SYSTEM"."SCHEMA_VERSION_REGISTRY" TO "DEV_IAU"`
2. Start the Upgrade Assistant.
   - (UNIX) `./ua -readiness`
   - (Windows) `ua.bat -readiness`

# Upgrade Assistant Readiness Check Fails: Oracle WSM Datasource Connection Details

When you reconfigure the domain, the upgrade assistant readiness check fails with the exception `Failed to read the Oracle WSM datasource connection details.`

The following exception is displayed:

```
[2019-09-22T23:14:50.828-04:00] [WSM] [INCIDENT_ERROR] []
[upgrade.WSM.WSMPLUGIN] [tid: 105] [ecid:
33feff0a-000a-49d1-88c7-2ae23c9712d9-00000002,0] [[
oracle.ias.update.exception.UpgradeException: WSMERROR-00015:
Failed to read the Oracle WSM datasource connection details.
at oracle.wsm.lifecycle.upgrade.impl.WSMUpgradePlugin.
initializePluginData(WSMUpgradePlugin.java:386)
at oracle.wsm.lifecycle.upgrade.impl.WSMUpgradePlugin.
readiness(WSMUpgradePlugin.java:150)
at oracle.ias.update.plugin.Plugin.readiness(Plugin.java:595)
```

To solve this error, complete the steps mentioned in Doc ID 2289605.1.

# Readiness Check for OAM Configuration Upgrade Fails

Before you run the readiness check for Oracle Access Manager for the first time, ensure that you have removed the IAMSuiteAgent security provider.

When you run the readiness check for the first time, the check fails for OAM configuration upgrade with the following error:

```
Remove the IAMSuiteAgent security provider as per EDG guide
```

To remove the IAMSuiteAgent security provider, do the following:

1. Log in to the Oracle WebLogic Server Administration Console using the following URL: `http://host.mycompany.com/console`
2. Select **Security Realms** from the **Domian Structure** menu.
3. Click **Myrealm**.

4. Go to the **Providers**tab.
5. Click **Lock and Edit** from the **Change Center** menu.
6. From the list of authentication providers, select **IAMSuiteAgent**.
7. Click **Delete**.
8. Click **Yes** to confirm the deletion.
9. Click **Activate Changes** from the **Change Center** menu to apply the changes.

# Error When Starting SSL Enabled OAM Managed Server After Upgrade

If SSL is enabled for Oracle Access Manager Managed Servers, the SSL port for the Administration Server must be changed manually before starting the servers.

This issue occurs when you upgrade Oracle Identity Manager (OIM) and Oracle Access Manager (OAM) integrated environments. If the SSL port is not updated for the SSL enabled Oracle Access Manager Managed Server, the following exception is displayed when you start the Managed Server:

```
<Error> <Server>  <idmr2ps3> <AdminServer> <[ACTIVE] ExecuteThread: '11'
for  queue: 'weblogic.kernel.Default (self-tuning)'> <<WLS Kernel>> <>
<303f1768-cdd2-4e0c-9b1e-564a32e22aa1-00000056> <1494577396454>  <[severity-
value: 8]
[rid: 0] [partition-id: 0] [partition-name: DOMAIN] >  <BEA-002606> <The
server is unable to
create a server socket for listening on  channel "DefaultSecure[iiops]". The
address x.x.x.x
might be incorrect  or another process is using port 7503:
java.net.BindException: Address  already in use>
```

The following exception is seen in the Administration Server log file:

```
<Error> <Server> <idmr2ps3>  <AdminServer>
<DynamicJSSEListenThread[DefaultSecure]>
<<WLS Kernel>> <>
<1880691887b793b2:4b6e5462:15ba94a4abd:-8000-0000000000000015>
<1493194022003>
<BEA-002606> <Unable to create a server socket for listening  on channel
"DefaultSecure".
The address x.x.x.x might be incorrect or  another process is using port
7503: java.net.BindException: Address already  in use.>
```

To resolve this issue, do the following:

1. Change the SSL port of the Administration Server from `7503` to another free port, for example, `7505`, on the WebLogic Administration Console.
2. Edit the startManagedWebLogic.sh file located at DOMAIN_HOME/bin/ to change the port from `7503` to `7505`.

In an OIM and OAM integrated environment, you must use different SSL ports for OIM Administration Server and OAM Administration Server.

# Readiness Check for OPSS Schema Fails

When you upgrade Oracle Access Manager 11.1.2.3.0 environments that is upgraded from 11*g* Release 2 (11.1.2.1.0), the readiness check for Oracle Platform Security Services (OPSS) schema fails with the following exception”

```
Starting schema test:  SEQUENCE_TEST  Test that the Oracle Platform
Security
Services schema sequence and its properties are valid
EXCEPTION JPSCHANGELOG_SEQ sequence is missing
EXCEPTION JPSDN_SEQ sequence is missing
EXCEPTION JPSATTRS_SEQ sequence is missing
Completed schema test: SEQUENCE_TEST --> Test that the Oracle Platform
Security
Services schema sequence and its properties are valid +++ FAIL  .
Starting schema test:  TEST_REQUIRED_TABLES  Test that the schema
contains
all the required tables
EXCEPTION    Schema is missing a required table: JPS_ENTITY_LOCK
Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema
contains all the required tables +++ FAIL
```

This is a known issue for a multi-step (chain) upgrade. This exception can be ignored.

# OAM Upgrade Fails With InvalidKeyException

Oracle Access Manager upgrade fails with InvalidKeyException if Java JSE Policy is not upgraded.

The following exception is displayed:

```
oracle.security.jps.JpsException:
oracle.security.jps.service.keystore.KeyStoreServiceException:
Failed to perform cryptographic operation
Caused by: java.security.InvalidKeyException: Illegal key size
```

To resolve this issue, upgrade the Java JSE policy using the instructions described in Upgrading Java JSE Policy.

# OWSM Error Messages in the Reconfiguration Logs

During the Oracle Access Manager (OAM) upgrade, when you reconfigure the OAM domain, Oracle Web Services Manager (OWSM) error messages are seen in the reconfig logs.

The following error messages are seen in the reconfig logs:

```
2017-07-23 10:49:11,791 SEVERE [18]
oracle.wsm.common.logging.WsmMessageLogger - Following validation
errors were
encountered while validating document
```

```
"/assertiontemplates/oracle/http_pkinit_over_ssl_template" :
2017-07-23 10:49:11,868 SEVERE [18]
oracle.wsm.common.logging.WsmMessageLogger - Following validation errors
were
encountered while validating document
"/assertiontemplates/oracle/http_kinit_over_ssl_template" :
2017-07-23 10:49:35,462 SEVERE [18]
oracle.wsm.common.logging.WsmMessageLogger - Following validation errors
were
encountered while validating document
"/policies/oracle/multi_token_over_ssl_client_policy" :
2017-07-23 10:49:35,562 SEVERE [18]
oracle.wsm.common.logging.WsmMessageLogger - Following validation errors
were
encountered while validating document
"/policies/oracle/multi_token_client_policy" :
```

The errors are caused because of the corrupted custom documents which need to be either removed or fixed before upgrade.

This does not impact the functionality of OWSM functionality, and hence can be ignored.

# OAM Console Shows No Application Domains After Upgrade

After you upgrade Oracle Access Manager (OAM) in an integrated setup where you have deployed Oracle Identity Manager, Oracle Access Manager, Oracle Unified Directory, and Oracle Adaptive Access Manager, when you search for application domains on OAM console, it shows no result.

The following is shown as the search result:

```
not able to search any application domain.
```

To resolve this, ensure that you have specified the right values for `OUDAuthenticator` on the Administration Console, post upgrade, by doing the following:

1. Log in to the WebLogic Administration Console using the following URL:.

   `http://`*`adminserver_host`*`:`*`adminserver_port`*`/console`
2. Click **Realm**, and select **Providers**.
3. Click **OUDprovider**.
4. Ensure that you have the following values set for the OUD group configuration of OUDprovider:
   - Static Group Object Class: groupOfUniqueNames
   - Static Member DN Attribute: uniqueMember
   - Static Group DN from Member DN filter: (&(uniqueMember=%M) (objectclass=groupOfUniqueNames))

# Troubleshooting Security Policy Issues When Upgrading

OAM 12c has an improved security posture and leverages the capabilities added in the underlying infrastructure. OAM 12c is certified with JDK 8, and based on the JDK 8 update used, its behavior may vary. More details about specific JDK 8 updates and their corresponding Java policies can be found in Release Notes for JDK 8 and JDK 8 Update Releases.

Oracle Access Protocol (OAP) version 5 has improved security for WebGate and server communication. OAP version 5 is used for communication between 12c WebGates and 12c OAM Servers.

- Modifying the Java Security Posture
- Upgrade Scenarios for OAM

# Modifying the Java Security Posture

OAM Server 12c supports TLS1.2 and SHA-2. For compatibility with older products (including Webgate, OIM, and OAAM), relax the OAM security posture by making the following changes to the java.security policy:

1. Remove TLSv1, TLSv1.1, MD5withRSA from the following key:

   ```
   key - jdk.tls.disabledAlgorithms
   ```

2. Remove MD5 from the following key:

   ```
   key - jdk.certpath.disabledAlgorithms
   ```

# Upgrade Scenarios for OAM

An upgraded OAM environment can result in the following cases:

- If WebGate is upgraded and the OAM Server is not, then SSL communication between them uses TLSv1 with MD5 certificates.
- If OAM Server is upgraded and WebGate is not, then SSL communication between them fails, as the OAM Server rejects MD5 certificates and doesn't support TLSv1. In this case, you need to modify the Java security policy to enable TLSv1, TLSv1.1 and MD5.
- If both OAM Server and WebGate are upgraded, edit the WebGate profile and copy the WebGate artifacts to the WebGate config folder. SSL communication between the OAM Server and WebGates will use TLSv1.2 with SHA-2 certificates.

**WebGates**

12c PS2/R2PS3 WebGates that employ version 4 of the OAP protocol will continue to work with OAM 12c. However, these WebGates must be upgraded to leverage the full capability of 12c. To upgrade the WebGates:

1. Stop the WebGates (OHS/OTD)
2. Upgrade WebGate binaries to 12c PS3
3. Edit WebGate profile and register the updated profile
4. Copy the WebGate artifacts to the WebGate config folder
5. Start the WebGates (OHS/OTD)

### Multi-Data Center

If an upgrade results in a 12c Master server and an 11g clone server (or vice versa), then SSL communication between the servers fails. To enable communication between these servers, modify the java.security policy to enable TLSv1, TLSv1.1, and MD5 as suggested above.

### Client Certificates

OAM Server 12c rejects older client/user X.509 certificates that don't adhere to JDK 8 security requirements. See Release Notes for JDK 8 and JDK 8 Update Releases for MD5- and TLS-related restrictions for the JDK 8 update specific to the system. This behavior is governed by the JDK 8 java.security policy. To ensure acceptance of older client/user X.509 certificates, modify the java.security policy to enable TLSv1, TLSv1.1, and MD5 as described above.

### Federation

For scenarios that involve Service Provider (SP) or Identity Provider (IDP) registration, the certificates used may undergo the same limitations as that for Client Certificates listed above.

Note that federation agreements will break if the Token Signing Certificate is changed. As a result, the 11g security posture is carried forward after upgrading, which may require enabling the legacy algorithms (TLSv1, TLSv1.1, and MD5), as described above. The use of SHA-2 certificates is supported.

### OIC

Similar to Federation, changing the OAuth Token Signing Certificate breaks existing trust relationships. As a result, the 11g security posture is carried forward after upgrading, which may require enabling the legacy algorithms (TLSv1, TLSv1.1, and MD5), as described above. The use of SHA-2 certificates is supported.

B

# Updating the JDK After Installing and Configuring an Oracle Fusion Middleware Product

Consider that you have a JDK version jdk1.8.0_121 installed on your machine. When you install and configure an Oracle Fusion Middleware product, the utilities, such as Configuration Wizard (`config.sh|exe`), OPatch, or RCU point to a default JDK, for example, jdk1.8.0_121. After some time, Oracle releases a new version of the JDK, say jdk1.8.0_131 that carries security enhancements and bug fixes. From 12*c* (12.2.1.3.0) onwards, you can upgrade the existing JDK to a newer version, and can have the complete product stack point to the newer version of the JDK.

You can maintain multiple versions of JDK and switch to the required version on need basis.

- About Updating the JDK Location After Installing an Oracle Fusion Middleware Product
  The binaries and other metadata and utility scripts in the Oracle home and Domain home, such as RCU or Configuration Wizard, use a JDK version that was used while installing the software and continue to refer to the same version of the JDK. The JDK path is stored in a variable called JAVA_HOME which is centrally located in .globalEnv.properties file inside the *`ORACLE_HOME`*`/oui` directory.

## About Updating the JDK Location After Installing an Oracle Fusion Middleware Product

The binaries and other metadata and utility scripts in the Oracle home and Domain home, such as RCU or Configuration Wizard, use a JDK version that was used while installing the software and continue to refer to the same version of the JDK. The JDK path is stored in a variable called JAVA_HOME which is centrally located in .globalEnv.properties file inside the *`ORACLE_HOME`*`/oui` directory.

The utility scripts such as config.sh|cmd, launch.sh, or opatch reside in the *ORACLE_HOME*, and when you invoke them, they refer to the JAVA_HOME variable located in .globalEnv.properties file. To point these scripts and utilities to the newer version of JDK, you must update the value of the JAVA_HOME variable in the .globalEnv.properties file by following the directions listed in Updating the JDK Location in an Existing Oracle Home .

To make the scripts and files in your Domain home directory point to the newer version of the JDK, you can follow one of the following approaches:

- Specify the path to the newer JDK on the Domain Mode and JDK screen while running the Configuration Wizard.

  For example, consider that you installed Oracle Fusion Middleware Infrastructure with the JDK version 8u191. So while configuring the WebLogic domain with the Configuration Assistant, you can select the path to the newer JDK on the Domain Mode and JDK screen of the Configuration Wizard. Example: `/scratch/jdk/jdk1.8.0_131`.

- Manually locate the files that have references to the JDK using `grep` (UNIX) or `findstr` (Windows) commands and update each reference. See Updating the JDK Location in an Existing Oracle Home .

> **Note:**
>
> If you install the newer version of the JDK in the same location as the existing JDK by overwriting the files, then you don't need to take any action.

- Updating the JDK Location in an Existing Oracle Home
  The `getProperty.sh|cmd` script displays the value of a variable, such as JAVA_HOME, from the .globalEnv.properties file. The `setProperty.sh|cmd` script is used to set the value of variables, such as OLD_JAVA_HOME or JAVA_HOME that contain the locations of old and new JDKs in the .globalEnv.properties file.
- Updating the JDK Location in an Existing Domain Home
  You must search the references to the current JDK, for example jdk1.8.0_121 manually, and replace those instances with the location of the new JDK.

# Updating the JDK Location in an Existing Oracle Home

The `getProperty.sh|cmd` script displays the value of a variable, such as JAVA_HOME, from the .globalEnv.properties file. The `setProperty.sh|cmd` script is used to set the value of variables, such as OLD_JAVA_HOME or JAVA_HOME that contain the locations of old and new JDKs in the .globalEnv.properties file.

The `getProperty.sh|cmd` and `setProperty.sh|cmd` scripts are located in the following location:

(UNIX) *ORACLE_HOME*`/oui/bin`
(Windows) *ORACLE_HOME*`\oui\bin`
Where, *ORACLE_HOME* is the directory that contains the products using the current version of the JDK, such as jdk1.8.0_121.

To update the JDK location in the .globalEnv.properties file:

1. Use the `getProperty.sh|cmd` script to display the path of the current JDK from the JAVA_HOME variable. For example:

   (UNIX) *ORACLE_HOME*`/oui/bin/getProperty.sh JAVA_HOME`
   (Windows) *ORACLE_HOME*`\oui\bin\getProperty.cmd JAVA_HOME`
   `echo JAVA_HOME`
   Where JAVA_HOME is the variable in the .globalEnv.properties file that contains the location of the JDK.
2. Back up the path of the current JDK to another variable such as OLD_JAVA_HOME in the .globalEnv.properties file by entering the following commands:

   (UNIX) *ORACLE_HOME*`/oui/bin/setProperty.sh -name OLD_JAVA_HOME -value` *specify_the_path_of_current_JDK*
   (Windows) *ORACLE_HOME*`\oui\bin\setProperty.cmd -name OLD_JAVA_HOME -value` *specify_the_path_of_current_JDK*

This command creates a new variable called OLD_JAVA_HOME in the .globalEnv.properties file, with a value that you have specified.

3. Set the new location of the JDK in the JAVA_HOME variable of the .globalEnv.properties file, by entering the following commands:

   (UNIX) *ORACLE_HOME*/oui/bin/setProperty.sh -name JAVA_HOME -value *specify_the_location_of_new_JDK*
   (Windows) *ORACLE_HOME*\oui\bin\setProperty.cmd -name JAVA_HOME -value *specify_the_location_of_new_JDK*

   After you run this command, the JAVA_HOME variable in the .globalEnv.properties file now contains the path to the new JDK, such as jdk1.8.0_131.

## Updating the JDK Location in an Existing Domain Home

You must search the references to the current JDK, for example jdk1.8.0_121 manually, and replace those instances with the location of the new JDK.

You can use the `grep` (UNIX) or `findstr` (Windows) commands to search for the jdk-related references.

You'll likely be required to update the location of JDK in the following three files:

(UNIX) *DOMAIN_HOME*/bin/setNMJavaHome.sh
(Windows) *DOMAIN_HOME*\bin\setNMJavaHome.cmd

(UNIX) *DOMAIN_HOME*/nodemanager/nodemanager.properties
(Windows) *DOMAIN_HOME*\nodemanager\nodemanager.properties

(UNIX) *DOMAIN_HOME*/bin/setDomainEnv.sh
(Windows) *DOMAIN_HOME*\bin\setDomainEnv.cmd