Oracle® Fusion Middleware

Upgrading Oracle HTTP Server

14c (14.1.2.0.0)
F85546-03
June 2025

Oracle Fusion Middleware Upgrading Oracle HTTP Server, 14c (14.1.2.0.0)

F85546-03

Primary Author: Oracle Corporation

Contents

3 Upgrading a Standalone Oracle HTTP Server

4 Upgrading a Managed Oracle HTTP Server

Preface

This preface contains the following sections:

Audience

This manual is intended for Oracle Fusion Middleware system administrators who are responsible for upgrading Oracle Fusion Middleware. It is assumed that the readers of this manual have knowledge of the following:

- Oracle Fusion Middleware 12*c* (12.2.1.4.0) or Oracle Fusion Middleware 14*c* (14.1.1.0.0) system administration and configuration information for the existing deployment
- The configuration and expected behavior of the system or systems being upgraded

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at https://www.oracle.com/corporate/accessibility/.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit https://support.oracle.com/portal/ or visit Oracle Accessibility Learning and Support if you are hearing impaired.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Documents

Upgrade documentation is organized by tasks in the 14c (14.1.2.0.0) documentation library. The task-specific pages provide direct links to common upgrade procedures and related documentation.

You can refer the Oracle Fusion Middleware Library for additional information.

- For installation information, see Fusion Middleware Installation Documentation.
- For upgrade information, see Fusion Middleware 14*c* Upgrade Documentation.

- For administration-related information, see Fusion Middleware 14*c* Administration Documentation.
- For release-related information, see Fusion Middleware 14*c* Release Notes.

Conventions

The following text conventions are used in this document:

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 HTTP Server to 14c (14.1.2.0.0)

Before you begin, review all introductory information to understand the standard upgrade topologies and upgrade paths for Oracle HTTP Server 14c (14.1.2.0.0)

About the Starting Points for an Oracle HTTP Server Upgrade

You can upgrade a managed (collocated) Oracle HTTP Server deployment from Oracle Fusion Middleware 12c (12.2.1.4.0) and a standalone Oracle HTTP Server 12c (12.2.1.4.0) and 14 c (14.1.1.0.0) to 14c (14.1.2.0.0)

Supported release starting points are:

- Oracle HTTP Server 12c (12.2.1.4) and and 14 c (14.1.1.0.0) for standalone servers
- Oracle HTTP Server 12*c* (12.2.1.4) for managed servers

About the Oracle HTTP Server Standard Topologies

The steps to upgrade Oracle HTTP Server to 14c (14.1.2.0.0) depend on the existing production topology.

As a result, it is difficult to provide exact upgrade instructions for every possible Oracle HTTP Server installation. Therefore, this upgrade documentation provides instructions for upgrading several typical Oracle HTTP Server topologies. These typical topologies are referred to as *standard upgrade 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 HTTP Server topologies.

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.

For a list of products and features available in Oracle Fusion Middleware Release 14.1.2.0.0, see Products and Features Available in Oracle Fusion Middleware in *Understanding Interoperability and Compatibility*.

When performing the upgrade of your hardware or software, verify that your Oracle Fusion Middleware software is certified to support the new operating system or computer hardware. For more information, refer to the following resources:

- Oracle Fusion Middleware Supported System Configurations
- Oracle® Fusion Middleware System Requirements and Specifications

Determining whether Oracle HTTP Server is Standalone or Managed (Collocated)

Oracle HTTP Server is the web server component for Oracle Fusion Middleware. It provides a listener for Oracle WebLogic Server and the framework for hosting static pages, dynamic pages, and applications over the Web. If you configure Oracle HTTP Server in a WebLogic Server domain, it is considered a **managed** Oracle HTTP Server because you can manage the Oracle HTTP Server instances like any other elements of the WebLogic Server domain using Enterprise Manager Fusion Middleware Control, the WLST command-line interface, or the Node Manager. If you install the Oracle HTTP software in a separate Oracle home, without installing the Oracle Fusion Middleware Infrastructure, it is considered a **standalone** deployment.
To determine whether you are upgrading a managed or a standalone Oracle HTTP Server:

Check the element `extention-template-ref` and its attribute `name` in the file *`DOMAIN_HOME`*`/init-info/domain-info.xml`. If you find an element with the name **Oracle HTTP Server (Standalone)**, then it is a standalone Oracle HTTP Server. And if you find an element with name **Oracle HTTP Server (Collocated)**, then it is collocated.

> **Note:**
>
> This guide covers the procedures for upgrading both managed and standalone OHS deployments. Perform only those tasks that apply to your environment.

2

Preparing to Upgrade Oracle HTTP Server

Upgrade is performed while the servers are down. The pre-upgrade tasks are often time-consuming. Oracle recommends that you plan and prepare your environment for upgrade by completing these pre-upgrade tasks, so that you have a successful upgrade and a limited downtime.

Use the following checklist to make sure you complete the pre-upgrade tasks:

Pre-Upgrade Checklist

The Pre-Upgrade Checklist identifies tasks that can be performed before you begin your upgrade to ensure that you have a successful upgrade and limited downtime.

Upgrades are performed while the servers are down. This checklist is meant to identify important — and often time-consuming — pre-upgrade tasks that you can perform before the upgrade to limit your downtime. The more preparation you can do before you begin the upgrade process, the less time you spend offline.

> **Note:**
>
> The pre-upgrade procedures you perform depend on the configuration of your existing system, the components you are upgrading, and the environment that 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.**

This table describes the Pre-Upgrade Checklist. It lists all the required components and describes them in detail.

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

Task	Description
Required Create a complete backup of your existing environment.	Back up all system-critical files 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. See Creating a Complete Backup. • Make sure that your backup includes the schema version registry table. See Backing Up the Schema Version Registry Table. • If you have modified or customized any of the startup scripts or any of the configuration files in your existing domain (for example, setting a value for the cookie-path property), you need to copy them to the temporary directory location (outside of the existing domain) during the upgrade, and redeploy them after the upgrade.

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

Task	Description
Optional Clone your production environment to use as an upgrade testing platform.	In addition to creating a complete backup of your system files, Oracle strongly recommends that you clone your production environment. This environment can be used to test the upgrade.
Required Verify that you install and upgrade your product on a supported hardware and software configuration. **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 documents. Also make sure to use a supported JDK version before you install the product distributions. Oracle recommends that you verify this information right before you start the upgrade as the certification requirements are frequently updated. **Make sure that you have applied the latest patches to your components before you upgrade.** See Verifying Certification and System Requirements.
Optional Create a Non-SYSDBA user to run the Upgrade Assistant with necessary privileges.	Oracle recommends that you create the FMW user to run the Upgrade Assistant. The FMW user can run the Upgrade Assistant without any system administration privileges. See Creating a Non-SYSDBA User to Run the Upgrade Assistant.
Required Linux and UNIX Operating System users must set their `DISPLAY` environment variables before starting the Fusion Middleware tools.	Setting the DISPLAY Environment Variable If the `DISPLAY` environment variable is not set up properly to allow for GUI mode, you may encounter an error.

Creating a Complete Backup

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

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

The Upgrade Assistant Prerequisites screen prompts you to acknowledge that backups have been performed before you proceed with the actual upgrade. However, note that 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 14c (14.1.2.0.0) in *Planning an Upgrade of Oracle Fusion Middleware*

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.

> **Note:**
>
> This step is only required for managed or collocated domains. Standalone domains will not have this 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 *`EXISTING_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*.

Cloning Your Source Environment for Testing

Create a copy of your source environment, upgrade the cloned environment, verify that the upgraded components work as expected, and then (and only then) upgrade your environment.

Cloning your source environment for testing is recommended, but not required.

Upgrades cannot be reversed. In most cases, if an error occurs, you must stop the upgrade and restore the entire environment from backup and begin the upgrade process from the beginning. Identifying potential upgrade issues in a cloned environment can eliminate unnecessary downtime.

> **Note:**
>
> It is beyond the scope of this document to describe the cloning procedures for all components and operating systems. Cloning procedures are component and operating system-specific. At a high level, you install the pre-upgrade version of your component domain on a test machine, create the required schemas using the Repository Creation Utility (RCU), and perform the upgrade.

Additional benefits of running an upgrade in a cloned environment include the following:

- Uncover and correct any upgrade issues.
- Practice completing an end-to-end upgrade.
- Understand the upgrade performance and how purge scripts can help.
- Understand the time required to complete the upgrade.
- Understand the database resource usage (such as temporary tablespace; PGA, and so on).

> **Note:**
>
> You can run the pre-upgrade Readiness Check on the cloned environment to help identify potential upgrade issues with your data, but you must perform a complete test upgrade on a cloned environment to ensure a successful upgrade.

Verifying Certification and System Requirements

Review the certification matrix and system requirements documents to verify that your environment meets the necessary requirements for installation. You may be required to upgrade your operating system, hardware or other software packages.

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

> **WARNING:**
>
> Make sure that your current environment has been patched to the latest patch set *before* you begin the upgrade. Certifications are based on fully patched environments, unless stated otherwise.

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 14c (14.1.2.0.0).

Verify System Requirements and Specifications

It is important to use both the System Requirements and Specifications document and the Oracle Fusion Middleware Certification Matrix 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 Oracle Fusion Middleware Certification matrix are met. For example, if the Certification Matrix indicates that your product is certified for installation on 64-Bit Oracle Linux 8, the System Requirements and Specifications document should be used to verify that your Oracle Linux 8 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 Technology Network (OTN).

> **Note:**
>
> Do not attempt an upgrade if you are unable to meet the minimum system requirements.

Specifically, you can use the Oracle Fusion Middleware System Requirements and Specifications document to verify the following:

- Processor Requirements
- Java Development Kit (JDK) Requirements
- General Memory and Disk Space Requirements
- Product-Specific Memory and Disk Space Requirements
- Network Requirements
- UNIX Operating System Requirements
- Windows Operating Systems Requirements
- Virtualization Requirements
- Database Requirements

What if my operating system is not supported?

If you are running your environment on an unsupported operating system, you will need to create a supported environment before you begin your upgrade. Do not attempt an upgrade on an unsupported operating system.

Use the migration steps for your environment.

Migrating a Standalone Domain from an Unsupported Operating System

If your standalone domain is installed on an unsupported operating system, then you will need to perform an out-of-place upgrade on a machine running a supported operating system.

You cannot perform an upgrade to 14c (14.1.2.0.0) if your existing deployment is not running on a supported operating system. You must migrate your existing deployment to a supported operating sytem before you begin an upgrade.

> **Note:**
>
> This process assumes the existing Oracle home and domain home were deployed on an Oracle Linux 7 machine and the same directory structure will be replicated on the new machine. These steps assume you are upgrading to Oracle Linux 8, but consult the Certification Matrix for 14c (14.1.2.0.0) for a complete list of supported operating systems.

1. Pack the Oracle HTTP Server 12c (12.2.1.4) domain using `pack.sh` on the Oracle Linux 7 host. Change the code example below to reflect your actual directory structure, file names, usernames and passwords.

```
cd 12214_ORACLE_HOME/oracle_common/common/bin/
```

```
./pack.sh -domain=../user_projects/domains/base_domain -template=/scratch/
OHS12214.jar -template_author=<user_name> -template_name=OHS_domain
```

2. Copy the packed jar to the Oracle Linux 8 host.

> **Note:**
>
> Do not unpack the jar file. At this stage you are just copying the file to a temporary location on the new Oracle Linux 8 host until it is time to unpack the domain into the new 14c (14.1.2.0.0) Oracle Home.

3. Install Oracle HTTP Server 14c (14.1.2.0.0) on the Oracle Linux 8 host using the same directory structure that was used on the Oracle Linux 7 host. See Installing the Standalone Oracle HTTP Server.

> **Note:**
>
> The Oracle home and domain home path used on the Oracle Linux 7 machine should be available for use on the Oracle Linux 8 machine.

4. Invoke the `unpack.sh` script using the jar file you packed in Step 1. This recreates the 12.2.1.4.0 domain on the Oracle Linux 8 machine.
Change the code example below to reflect your actual directory structure, file names, usernames and passwords.

```
cd 1412_ORACLE_HOME/oracle_common/common/bin/
```

```
./unpack.sh -domain=../user_projects/domains/base_domain -template=/
scratch/OHS12214.jar -user_name=weblogic -password=<enter your password>
```

5. Run the 14c (14.1.2.0.0) Reconfiguration Wizard from the Oracle Linux 8 Oracle Home as described in Reconfiguring the Existing Domain with the Reconfiguration Wizard. When prompted, be sure to provide the location of the unpacked 12.2.1.4.0 domain on the Oracle Linux 8 machine.
6. Run the 14c (14.1.2.0.0) Upgrade Assistant from the Oracle Linux 8 Oracle home as described in Upgrading Standalone System Component Configurations Using the Upgrade Assistant. When prompted, be sure to provide the location of the unpacked 12.2.1.4.0 domain on the Oracle Linux 8 machine.

Migrating a Managed Domain from an Unsupported Operating System

If you are currently running your managed or collocated Oracle Fusion Middleware domain on an unsupported operating system, then you must migrate your existing environment to a supported operating system before you upgrade.

After the migration, validate that all of your existing Oracle Fusion Middleware 12c (12.2.1.4.0) software is working properly on the updated machine, and only then perform the upgrade to Oracle Fusion Middleware 14c (14.1.2.0.0).

In these tasks, *host* refers to the existing unsupported source machine and *target* refers to the new supported 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.

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 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, 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 Remote Console. See Starting and Stopping Administration and Managed Servers and Node Manager.
>
> As of release 14c (14.1.2.0.0), the WebLogic Server Administration Console has been removed. For comparable functionality, you should use the WebLogic Remote Console. For more information, see Oracle WebLogic Remote Console.

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

> **Note:**
>
> It is important that you stop the following servers in the correct order.

Step 1: Stop System Components

To stop system components, such as Oracle HTTP Server, use the `stopComponent` script:

- (UNIX) *`EXISTING_DOMAIN_HOME`*`/bin/stopComponent.sh` *`component_name`*
- (Windows) *`EXISTING_DOMAIN_HOME`*`\bin\stopComponent.cmd` *`component_name`*

You can stop system components in any order.

Step 2: Stop Any Managed Servers

To stop a WebLogic Server Managed Server, use the `stopManagedWebLogic` script:

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

When prompted, enter your user name and password.

Step 3: Stop the Administration Server

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

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

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

Step 4: Stop Node Manager

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

Alternatively, after setting 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 Oracle WebLogic Server*.

Back Up All Files from the Host Machine

Make sure that you have created a complete backup of your entire 12c (12.2.1.4.0) 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 operating system upgrade takes place on the same machine, there is a risk of corrupting the source environment if the upgrade fails.
> For general information about creating a complete backup of your existing environment, 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:

- *12c_DOMAIN_HOME*
- *12c*`/nodemanager` directory located in *12c_ORACLE_HOME*`/wlserver/common/`

The following steps explain how to use the **pack** command to create a domain template jar file. This is only one method that can be used to create a backup. Consult your own backup and recovery plans to choose the backup method that best suits your deployment.

1. Pack the domain that was created on the unsupported host using the pack command as follows:

```
cd ORACLE_HOME/oracle_common/common/bin/
```

```
./pack.sh -domain=/scratch/username/<product>_12214/user_projects/domains/
base_domain -template=/scratch/<product>.jar - template_author=<user_name>
-template_name=<product>_domain
```

2. Copy the domain template jar file that you just created to the new supported host.
Do not unpack the jar file. At this stage you are just copying the file to a temporary location on the new host until it is time to unpack the domain into the new 14.1.2 Oracle Home. To simplify the unpack process, consider recreating the exact same directory structure that you used in your 12.2.1.4 domain. This will ensure that the file is not overwritten.

> **Note:**
>
> Do not proceed with the upgrade until you have a complete backup.

Set Up the Target Machine with the 12*c* 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.

Copy the Contents of the Domain Template to the New Target Host

Unpack the contents of the generated domain template jar file on the target host. The directory structure on the target machine must be identical to the structure of the host machine.

1. On the target machine, navigate to the new Oracle home.

   ```
   cd 1412_ORACLE_HOME/oracle_common/common/bin/
   ```

2. Use the unpack command to copy the files to the new target:

   ```
   ./unpack.sh -domain=/scratch/<username>/<product>_12214/user_projects/
   domains/base_domain -template=/scratch/<product>.jar -user_name=weblogic -
   password=<enter your password>
   ```

Install the 14c (14.1.2.0.0) Product Distributions on the Target Machine

Oracle recommends an Out-of-Place approach for upgrade. Therefore, you must install the 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 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 12c (12.2.1.4.0) Node Manger files that you backed up from the source 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 14c (14.1.2.0.0).

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 14c (14.1.2.0.0).

> **Note:**
>
> If your database version is no longer supported, you must upgrade to a supported version before starting an upgrade.

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

If your JDK is not supported, or you do not have a JDK installed, you must download the required Java SE JDK before you begin.

Refer to the Oracle Fusion Middleware Supported System Configurations information on the Oracle Technology Network (OTN) 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`.

Verify the Database User for the WLSSchemaDataSource Data Source

This step is required if your existing domain has a `WLSSchemaDataSource` data source.

If your domain has the `WLSSchemaDataSource` data source, then you will need to verify which database user is assigned to it. If `<PREFIX>_WLS_RUNTIME` is assigned to it, then you need to change that to `<PREFIX>_WLS` .

This change is necessary due to the following changes:

- The 14c (14.1.2.0.0) Upgrade Assistant uses the information in the `WLSSchemaDataSource` data source,when a domain-based schema upgrade is performed. That upgrade will fail if the `<PREFIX>_WLS` database user is not assigned to the `WLSSchemaDataSource`, or if `<PREFIX>_WLS` is not entered as the "Schema User Name" on the "WLS Schema" page of the Upgrade Assistant.
- Oracle recommends that you use the 12c Oracle WebLogic Administration Console to change the database user to `<PREFIX>_WLS` in the `WLSSchemaDataSource` data source. Doing this will avoid the Upgrade Assistant failure, and also allow the Reconfiguration Wizard to pre-populate fields with the correct values.
- The `<PREFIX>_WLS_RUNTIME` database user is reserved for use with a new `WLSRuntimeSchemaDataSource`, which was introduced in 14c (14.1.2.0.0). This new `WLSRuntimeSchemaDataSource` will be created when the 14c (14.1.2.0.0) Reconfiguration Wizard (reconfig.sh) is used to upgrade the domain.

You can use your Oracle WebLogic 12*c* Administration Console to change the user in the WLSSchemaDataSource from `<PREFIX>_WLS_RUNTIME` to `<PREFIX>_WLS`.

1. Log in the 12*c* (12.2.1.4.0) Administration Console.
2. In the administration console under Domain Structure, expand **Services** (by clicking the **+** next to it). Then click **Data Sources**.
3. If the user in Properties field contains `<PREFIX>_WLS_RUNTIME` , change it to`<PREFIX>_WLS`.
4. Save the change.
5. Use the Change Center to commit the change, if your domain is running in production mode.

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 14c (14.1.2.0.0) 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.

Purging Unused Data

Purging unused data and maintaining a purging methodology before an upgrade can optimize the upgrade process.

Some components have automated purge scripts. If you are using purge scripts, wait until the purge is complete before starting the upgrade process. The upgrade may fail if the purge scripts are running while using the Upgrade Assistant to upgrade your schemas.

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.

Note:

In this example we are using the name `FMW` for our non-SYSDBA administrator. Substitute `FMW` with your admin name.

When granting privileges, make sure that you specify your actual user names and password for the schemas in your domain.

```
CREATE USER FMW IDENTIFIED BY "<FMW password>";
GRANT pdb_dba TO FMW;
GRANT MANAGE SCHEDULER TO FMW;
GRANT USE ON EDITION ORA$BASE TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON DBMS_LOB TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON DBMS_OUTPUT TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON DBMS_STATS TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON sys.dbms_aq TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON sys.dbms_aqadm TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON sys.dbms_aqin TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON sys.dbms_aqjms TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON utl_file TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON dbms_lock TO FMW WITH GRANT OPTION;
GRANT SELECT ON sys.V_$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_$SESSION TO FMW WITH GRANT OPTION;
GRANT SELECT ON dba_scheduler_jobs TO FMW WITH GRANT OPTION;
GRANT SELECT ON dba_scheduler_job_run_details TO FMW WITH GRANT OPTION;
GRANT SELECT ON dba_scheduler_running_jobs TO FMW WITH GRANT OPTION;
GRANT SELECT ON dba_aq_agents TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON sys.DBMS_SHARED_POOL TO FMW WITH GRANT OPTION;
GRANT SELECT ON dba_2pc_pending TO FMW WITH GRANT OPTION;
GRANT SELECT ON dba_pending_transactions TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON DBMS_FLASHBACK TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON dbms_crypto TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON dbms_job TO FMW WITH GRANT OPTION;
GRANT SELECT ON dba_scheduler_job_classes TO FMW WITH GRANT OPTION;
GRANT SELECT ON SYS.DBA_DATA_FILES TO FMW WITH GRANT OPTION;
GRANT SELECT ON SYS.V_$ASM_DISKGROUP TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON SYS.DBMS_ASSERT 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 dba_objects TO FMW WITH GRANT OPTION;
GRANT SELECT ON dba_queue_subscribers TO FMW WITH GRANT OPTION;
GRANT SELECT ON dba_subscr_registrations TO FMW WITH GRANT OPTION;
GRANT EXECUTE ON DBMS_RLS TO FMW WITH GRANT OPTION;
GRANT READ ON CTXSYS.CTX_PENDING TO FMW WITH GRANT OPTION;
GRANT SELECT ON SYS.V_$PARAMETER TO FMW WITH GRANT OPTION;
GRANT CREATE PROCEDURE TO FMW;
GRANT SELECT ON dba_users TO FMW WITH GRANT OPTION;
GRANT ALL ON sys.v_$parameter TO FMW WITH GRANT OPTION;
```

Setting the DISPLAY Environment Variable

If the `DISPLAY` environment variable is not set up properly to allow for GUI mode, you may encounter an error.

Linux and UNIX Operating System Users:

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, set the `DISPLAY` environment variable to the system name or IP address of your local workstation, and rerun Upgrade Assistant.

If you continue to receive these errors after setting `DISPLAY`, try launching another GUI tool, such as `vncconfig`. If you see the same errors, your `DISPLAY` environment variable may still not be set correctly.

3
Upgrading a Standalone Oracle HTTP Server

This chapter describes the process to upgrade a Standalone Oracle HTTP Server from a Standalone Oracle HTTP Server 12*c* (12.2.1.4.0) release.
This chapter contains the following sections:

About the Standalone Oracle HTTP Server Upgrade Process

Review the flowchart and roadmap for an overview of the upgrade process for Oracle HTTP Server.

The following table describes the tasks that must be completed to upgrade a standalone Oracle HTTP Server.

Table 3-1 Tasks for Upgrading Standalone Oracle HTTP Server

Task	Description
Required Verify that you are upgrading a standalone Oracle HTTP Server.	To determine which Oracle HTTP Server you have in your existing environment, see Determining whether Oracle HTTP Server is Standalone or Managed (Collocated).
Required Complete the pre-upgrade tasks.	The pre-upgrade tasks include cloning your production environment, verifying system requirements and certifications. For a complete list of pre-upgrade tasks, see Preparing to Upgrade Oracle HTTP Server.
Required Install the Standalone Oracle HTTP Server.	Run the installation program to install the software. Select the installation type **Standalone Oracle HTTP Server (managed independently of WebLogic server)**. This transfers the software to your system and creates a new Oracle home directory. See Installing the Standalone Oracle HTTP Server.
Required Shut down the servers in the existing environment.	See Stopping Servers and Processes .
Required Reconfigure your existing domain.	Run the Reconfiguration Wizard from the new installation to reconfigure the existing domain. See About Reconfiguring the Domain.
Required Upgrade the standalone system component configurations.	See Upgrading Standalone System Component Configurations.
Required Restart the servers and processes.	See Starting Servers and Processes.
Required Verify the upgrade.	Your Oracle HTTP Server should continue to function as expected. If you have post-upgrade issues, you should troubleshoot the installation and retry the upgrade. See Verifying the Upgrade.

Installing the Standalone Oracle HTTP Server

Before you begin your upgrade, download the Oracle HTTP Server 14c (14.1.2.0.0) distribution on the target system and install it by using the Oracle Universal Installer.

To install the standalone Oracle HTTP Server:

1. Sign in to the target system.
2. Download the following from Oracle Technology Network or Oracle Software Delivery Cloud to your target system:

   Oracle HTTP Server (UNIX: `fmw_14.1.2.0.0_ohs_linux64.bin`), (Windows: `setup_fmw_14.1.2.0.0_ohs_win64.exe`)
3. Change to the directory where you downloaded the 14c (14.1.2.0.0) product distribution.
4. Ensure that the system, patch, kernel, and other requirements are met as specified in theRoadmap for Verifying Your System Environment.
5. Enter the following command to start the installation program:

   (UNIX) `./fmw_14.1.2.0.0_ohs_linux64.bin`

   (Windows) `setup_fmw_14.1.2.0.0_ohs_win64.exe`
6. 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.
7. On the Welcome screen, review the information to make sure that you have met all the prerequisites. Click **Next**.
8. 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**.
9. On the Installation Location screen, specify the location for the Oracle home directory and click **Next**.

   As you are installing the standard installation topology for Oracle HTTP Server in a standalone domain, you can specify an Oracle home directory of your choice. However, ensure that you install the software in a new Oracle home.

See Selecting Directories for Installation and Configuration in *Planning an Installation of Oracle Fusion Middleware*.

10. On the Installation Type screen, select **Standalone HTTP Server (Managed independently of WebLogic server)** and click **Next**.
11. 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).
12. On the Security Updates screen, indicate how you would like to receive security updates if you already have an Oracle Support account.

    If you do not have an Oracle Support account and you are sure that you want to skip this step, clear the check box and verify your selection in the follow-up dialog box.
13. 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.
14. On the Installation Progress screen, when the progress bar displays 100%, click **Finish** to dismiss the installer, or click **Next** to see a summary.
15. 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.

You have installed the Oracle HTTP Server in a standalone mode.

Stopping Standalone System Components

Before you run the Upgrade Assistant to upgrade your schemas and configurations, you must shut down the pre-upgrade environment.

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

> **Note:**
>
> It is important that you stop the following servers in the correct order.

Step 1: Stop System Components

To stop system components, such as Oracle HTTP Server, use the `stopComponent` script:

- (UNIX) *`EXISTING_DOMAIN_HOME`*`/bin/stopComponent.sh` *`component_name`*
- (Windows) *`EXISTING_DOMAIN_HOME`*`\bin\stopComponent.cmd` *`component_name`*

You can stop system components in any order.

Step 2: Stop Node Manager

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

Alternatively, after setting 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 Oracle WebLogic Server*.

About Reconfiguring the Domain

Run the Reconfiguration Wizard to reconfigure your domain component configurations to 14c (14.1.2.0.0).

> **Note:**
>
> The Reconfiguration Wizard does not update any of your own applications that are included in the domain.

When you reconfigure a standalone domain, the following occurs:

- The domain version number in the `config.xml` file for the domain is updated to the version of the target Oracle HTTP Server.
- The reconfiguration template for Oracle HTTP Server is automatically selected and applied to the domain. This template defines any reconfiguration tasks that are required to make the standalone domain compatible with the target Oracle HTTP Server version.
- The 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.

Backing Up the Domain

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

1. Create a backup of the domain directory.
2. 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 copy 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. Sign in to the system on which the domain resides.
2. Open the command shell (on UNIX operating systems) or open a command prompt window (on Windows operating systems).
3. Go to the `oracle_common/common/bin` directory:
   - (UNIX) *`NEW_ORACLE_HOME`*`/oracle_common/common/bin`
   - (Windows) *`NEW_ORACLE_HOME`*`\oracle_common\commom\bin`
4. 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.

   > **Note:**
   >
   > When you run this command, the following error message might appear to indicate that the default cache directory is not valid:
   >
   > ```
   > *sys-package-mgr*: can't create package cache dir
   > ```
   >
   > You can change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`. For example:
   >
   > `CONFIG_JVM_ARGS=-Dpython.cachedir=`*`valid_directory`*

Reconfiguring the Existing Domain with the Reconfiguration Wizard

Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing 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).
   - Scripts and other such files that support Oracle HTTP Server 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 14c (14.1.2.0.0) is 17.0.12 and later. Click **Next**.

   > **Note:**
   >
   > You cannot change the **Domain Mode** at this stage. Your domain will retain its pre-upgrade domain mode. If you want to change the domain to secure mode, then after the upgrade see Changing Domain Mode Post Upgrade.

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

4. The Node Manager screen is only displayed if the domain you are reconfiguring is currently using a per host Node Manager.

   On the Node Manager screen, select the Node Manager configuration to use for the reconfigured domain. The resulting configuration depends on the combination of options you select for **Node Manager Type** and **Node Manager Configuration**.

Table 3-2 Field Descriptions for Node Manager Screen

Option	Description
Per Domain Default Location	If you select this option, the Node Manager home is redefined to *DOMAIN_NAME*`/nodemanager` and you cannot edit the Node Manager home.
Per Domain Custom Location	Select this option if you want the per domain Node Manager configuration files to be created in a specific location for this domain. Specify the directory in the **Node Manager Home** field, or click **Browse** to use the navigation tree to select the location. The specified directory must be empty. The `nodemanager.properties` and `nodemanager.domains` files are created in this directory.
Node Manager Home	If you selected the **Per Domain Custom Location** option, click **Browse** to navigate to the directory location that you want to use to store the per domain Node Manager configuration.

Table 3-2 (Cont.) Field Descriptions for Node Manager Screen

Option	Description
Manual Node Manager Setup	If you select this option, creation of the Node Manager configuration for the domain is skipped and the remaining fields cannot be modified. If you want to use Node Manager in the domain, you must manually configure Node Manager as described in Completing the Node Manager Configuration. The reconfigured domain will still use a per host Node Manager configuration. You should also select this option if your existing domain is not configured to use Node Manager and you do not want to use Node Manager in the reconfigured domain. For more information about Node Manager configuration, see Administering Node Manager for Oracle WebLogic Server.
Node Manager Configuration	Select one of the following two options. These fields are not available if you selected **Manual Node Manager Setup**.
Create New Configuration	A per domain Node Manager configuration is automatically created for the reconfigured domain using default settings in `nodemanager.properties`. If necessary, you can modify `nodemanager.properties` after the domain has been successfully reconfigured.
Migrate Existing Configuration	The existing per host Node Manager configuration is migrated to a per domain configuration for the reconfigured domain. This does not include environment-specific settings for `ListenAddress`, `ListenPort`, `StartScriptName`, `JavaHome`, and `LogFile`.
Node Manager Home	If you selected the **Migrate Existing Configuration** option, enter or browse to the Node Manager home directory that you want to migrate to the reconfigured domain.

Apply Oracle Recommended Defaults	If you selected the **Migrate Existing Configuration** option, select this check box if you want to use Oracle-recommended defaults in the `nodemanager.properties` file. Deselect this check box if you want to continue using the settings in the `nodemanager.properties` file being migrated.
	Oracle-recommended properties with default values are as follows:
	`LogLimit=0`
	`AuthenticationEnabled=true`
	`LogLevel=INFO`
	`DomainsFileEnabled=true`
	`NativeVersionEnabled=true`
	`LogToStderr=true`
	`SecureListener=true`
	`LogCount=1`
	`StopScriptEnabled=false`
	`QuitEnabled=false`
	`LogAppend=true`
	`StateCheckInterval=500`
	`CrashRecoveryEnabled=false`
	`StartScriptEnabled=true`
	`LogFormatter=weblogic.nodemanager.server.LogFormatter`
	`ListenBacklog=50`

Table 3-2 (Cont.) Field Descriptions for Node Manager Screen

Option	Description
Node Manager Credentials: Username and Password	Specify the user name and password that you want to use to start Node Manager in the reconfigured domain.

5. 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.

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

   During this process:

   - Domain information is extracted, saved, and updated.
   - Scripts and other such files that support Oracle HTTP Server are updated.

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

7. 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. If the reconfiguration is successful, it displays `Domain Configuration 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 for further operations.

Upgrading Standalone System Component Configurations

Use the Upgrade Assistant to upgrade the standalone agent's component configurations.

Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 14c (14.1.2.0.0).

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 set the character encoding, run the following:
>
> UNIX operating systems:
>
> ```
> export UA_PROPERTIES="-Dfile.encoding=UTF-8 ${UA_PROPERTIES}"
> ```
>
> Windows operating systems:
>
> ```
> set UA_PROPERTIES=-Dfile.encoding=UTF-8 %UA_PROPERTIES%
> ```

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`

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

Upgrade Assistant Parameters

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

Table 3-3 Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-readiness`	Required for readiness checks **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. Schemas and configurations are checked. 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. The value must be a positive integer in the range 1 to 8. The default is 4.

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

Parameter	Required or Optional	Description
`-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. 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: • `TRACE` • `NOTIFICATION` • `WARNING` • `ERROR` • `INCIDENT_ERROR` The default logging level is `NOTIFICATION`. 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. The default locations are: (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/logs` *ORACLE_HOME*`/oracle_common/upgrade/temp` (Windows) *ORACLE_HOME*`\oracle_common\upgrade\logs` *ORACLE_HOME*`\oracle_common\upgrade\temp`
`-help`	Optional	Displays all of the command-line options.

Upgrading Standalone System Component Configurations Using the Upgrade Assistant

Navigate through the screens in the Upgrade Assistant to upgrade standalone system component configurations.

To upgrade the existing domain using 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 Standalone Components screen, select **Standalone System Component Configurations**.

   Select **Update an Existing Domain** and enter the location of the existing domain in the **Domain Directory** field. You can also click **Browse** to select the existing domain directory by using the navigation tree. 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 Instance Directories screen, select the correct source and click **Next**.
6. On the Examine screen, review the status of the Upgrade Assistant as it examines each standalone component, verifying that the standalone 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 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.

7. 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.

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 components 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. 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.

Verifying the Upgrade

You can verify that the upgrade is successful if you are able to start the Node Manager and the Standalone Oracle HTTP Server properly.

If you experience post-upgrade issues, you need to troubleshoot the installation and retry the upgrade. See Troubleshooting Oracle HTTP Server in *Administrator's Guide for Oracle HTTP Server*.

If you are not able to start the newly upgraded environment, a possible cause could be the use of MD5 certificates in your Oracle wallet. To check whether you are using MD5 signatures and how to replace them with SHA-2 certificates, see Replacing Certificate Signed Using MD5 Algorithm with Certificate Signed Using SHA-2 Algorithm .

Starting Node Manager

Use the Node Manager scripts in the new domain home to start Node Manager.

To start Node Manager, use the `startNodeManager script`:

(UNIX)

```
NEW_DOMAIN_HOME/bin/startNodeManager.sh
```

(Windows)

```
NEW_DOMAIN_HOME\bin\startNodeManager.cmd
```

Starting the Standalone Oracle HTTP Server

After the upgrade, start the standalone Oracle HTTP Server.

To start the standalone Oracle HTTP Server:

1. Go to the DOMAIN_HOME/bin directory:

   (UNIX) ORACLE_HOME/user_projects/domains/<*domain name*>/bin

   (WINDOWS) ORACLE_HOME\user_projects\domains\<*domain name*>\bin
2. Enter the command to start the standalone server.

   On UNIX operating systems:

   `./startComponent.sh` *`ohs_name`*

   On Windows operating systems:

   `startComponent.cmd` *`ohs_name`*

   See Starting and Stopping System Components in *Administering Oracle Fusion Middleware*.

4

Upgrading a Managed Oracle HTTP Server

This chapter describes the procedure to upgrade a managed Oracle HTTP Server from 12*c* (12.2.1.4.0) to 14c (14.1.2.0.0) .

This procedure assumes that you have an existing Fusion Middleware Infrastructure installed and configured (WebLogic Domain either in Full-JRF or Restricted JRF mode) on your system.

> **Note:**
>
> You can install and configure a managed/collocated Oracle HTTP Server in either Full-JRF or Restricted-JRF domain. The key difference between the two is that there is no database dependency in the case of the Restricted-JRF mode. See About Full-JRF and Restricted-JRF Modes.
>
> You have to upgrade the database schemas only for the Full-JRF mode, if required.

About the Managed Oracle HTTP Server Upgrade Process

Review the flowchart and roadmap for an overview of the upgrade process for Managed Oracle HTTP Server.

Table 4-1 describes the tasks that must be completed to upgrade a Managed Oracle HTTP Server.

Table 4-1 Tasks for Upgrading Managed Oracle HTTP Server

Task	Description
Required Verify that you are upgrading a Managed Oracle HTTP Server.	To determine which Oracle HTTP Server you have in your existing environment, see Determining whether Oracle HTTP Server is Standalone or Managed (Collocated).
Required If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks.	The pre-upgrade tasks include cloning your production environment, verifying system requirements and certifications, purging unused data, and creating non-SYSDBA user. For a complete list of pre-upgrade tasks, see Preparing to Upgrade Oracle HTTP Server.
Required Download and install the 14.1.2.0.0 Fusion Middleware Infrastructure and Oracle HTTP Server distributions.	The Infrastructure distribution packs the WebLogic Server and the Java Required Files (JRF) that are required to set up the foundation to install other Fusion Middleware products. As per the upgrade topology defined in this guide, you must install the Infrastructure in a new Oracle home. You must install Oracle HTTP Server in the Oracle home that is created when you installed the 14.1.2.0.0 Infrastructure. See Installing the Product Distributions.

Table 4-1 (Cont.) Tasks for Upgrading Managed Oracle HTTP Server

Task	Description
Required Verify the Database User for the WLSSchemaDataSource Data Source	This step is required if your existing domain has a `WLSSSchemaDataSource` data source. See Verify the Database User for the WLSSchemaDataSource Data Source
Optional Run the Readiness Check.	Running the Readiness Check helps using the Upgrade Assistant helps you to determine whether your pre-upgrade environment is ready for upgrade. For the complete procedure, see Running a Pre-Upgrade Readiness Check.
Required Shut down the 12*c* instance.	Before starting the upgrade process, shut down the Administration Server, the Managed Servers, and your existing instance. See Stopping Servers and Processes .
Required Upgrade the existing schemas with the Upgrade Assistant.	See Upgrading Product Schemas.
Required Reconfigure your existing domain.	Run the Reconfiguration Wizard from the 14.1.2.0.0 Oracle HTTP Server installation to reconfigure the existing domain. See Reconfiguring the Existing Domain with the Reconfiguration Wizard.
Required Upgrade your existing domain configurations.	After the installation, you need to use the Upgrade Assistant to upgrade the Oracle HTTP Server and system component infrastructure. See Upgrading Domain Components Using the Upgrade Assistant.
Required Restart the servers and processes.	The upgrade process is complete. You can now restart the Administration Server, the Managed Servers, and your 14.1.2.0.0 instance. See Starting Servers and Processes.
Required Verify the upgrade.	For verifying the upgrade, see Verifying the Upgrade.

Installing the Product Distributions

Before you begin your upgrade, download the 14c (14.1.2.0.0) product distributions on the target system and install them using Oracle Universal Installer.

> **Note:**
>
> When Infrastructure is required for the upgrade, you must install the Oracle Fusion Middleware Infrastructure distribution first before you install other Fusion Middleware products. If your JDK is not supported, or you do not have a JDK installed, you must download the required Java SE JDK before you begin

To install the 14c (14.1.2.0.0) distributions:

1. Sign in to the target system.
2. Download the following from Oracle Technology Network or Oracle Software Delivery Cloud to your target system:
   - Oracle Fusion Middleware Infrastructure (`fmw_14.1.2.0.0_infrastructure_generic.jar`)
   - Oracle HTTP Server (UNIX: `fmw_14.1.2.0.0_ohs_linux64.bin`), (Windows: `setup_fmw_14.1.2.0.0_ohs_win64.exe`)
3. Check that your machines meet the following requirements:
   - Ensure that the system, patch, kernel, and other requirements are met as specified in *Installing and Configuring Oracle HTTP Server*.
4. On UNIX platforms, if the `/etc/oraInst.loc` file exists, check that its contents are correct. Specifically, check that the inventory directory is correct and that you have write permissions for that directory.

   If the `/etc/oraInst.loc` file does not exist, you can skip this step.
5. Change to the directory where you downloaded the 14c (14.1.2.0.0) product distribution.
6. Start the installation program for Oracle Fusion Middleware Infrastructure:
   - (UNIX) *JDK_HOME*`/bin/java -jar fmw_14.1.2.0.0_infrastructure_generic.jar`
   - (Windows) *JDK_HOME*`\bin\java -jar fmw_14.1.2.0.0_infrastructure_generic.jar`
7. 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.

8. On the Welcome screen, review the information to make sure that you have met all the prerequisites. Click **Next**.
9. 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**.
10. On the Installation Location screen, specify the location for the Oracle home directory and click **Next**.

Since you are installing the standard installation topology for a collocated Oracle HTTP Server in a WebLogic Server domain, enter the path to an existing Oracle Fusion Middleware Infrastructure Oracle home.

See Selecting Directories for Installation and Configuration in *Planning an Installation of Oracle Fusion Middleware*.

11. On the Installation Type screen, select the following:
    - For Infrastructure, select **Fusion Middleware Infrastructure**
    - For Oracle HTTP Server, select **Collocated HTTP Server (Managed through WebLogic Server)**

    Click **Next**.
12. 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).
13. 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.
14. On the Installation Progress screen, when the progress bar displays 100%, click **Finish** to dismiss the installer, or click **Next** to see a summary.
15. 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.
16. 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) `./fmw_14.1.2.0.0_ohs_linux64.bin`

    (Windows) `setup_fmw_14.1.2.0.0_ohs_win64.exe`

You have installed the Oracle HTTP Server in a collocated mode.

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.

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

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 **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. Schemas and configurations are checked. Do not use this parameter if you have specified the `-examine` parameter.

Table 4-2 (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. 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. 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: • `TRACE` • `NOTIFICATION` • `WARNING` • `ERROR` • `INCIDENT_ERROR` The default logging level is `NOTIFICATION`. 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 4-2 (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. The default locations are: (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/logs` *ORACLE_HOME*`/oracle_common/upgrade/temp` (Windows) *ORACLE_HOME*`\oracle_common\upgrade\logs` *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 domain directory.
   - Specify schema credentials to connect to the selected schema: **Database Type**, **DBA User Name**, and **DBA Password**. 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**.

     > **Note:**
     >
     > The Upgrade Assistant automatically enables default credentials. If you are unable to connect, make sure that you manually enter the credentials for your schema before you continue.

   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 4-3 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_common/upgrade/logs`	The directory location of the generated log file.	No action required.
Domain Directory	Displays the domain location	No action required.
Readiness report location `/oracle_common/upgrade/logs`	The directory location of the generated readiness report.	No action required.
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.

Table 4-3 (Cont.) Readiness Report Elements

Report Information	Description	Required Action
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.

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 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, 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 Remote Console. See Starting and Stopping Administration and Managed Servers and Node Manager.
>
> As of release 14c (14.1.2.0.0), the WebLogic Server Administration Console has been removed. For comparable functionality, you should use the WebLogic Remote Console. For more information, see Oracle WebLogic Remote Console.

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

> **Note:**
>
> It is important that you stop the following servers in the correct order.

Step 1: Stop System Components

To stop system components, such as Oracle HTTP Server, use the `stopComponent` script:

- (UNIX) *EXISTING_DOMAIN_HOME*`/bin/stopComponent.sh` *component_name*
- (Windows) *EXISTING_DOMAIN_HOME*`\bin\stopComponent.cmd` *component_name*

You can stop system components in any order.

Step 2: Stop Any Managed Servers

To stop a WebLogic Server Managed Server, use the `stopManagedWebLogic` script:

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

When prompted, enter your user name and password.

Step 3: Stop the Administration Server

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

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

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

Step 4: Stop Node Manager

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

Alternatively, after setting 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 Oracle WebLogic Server*.

Upgrading Product Schemas

After stopping servers and processes, use the Upgrade Assistant to upgrade your 12.2.1.4.0 schemas to the 14c (14.1.2.0.0) release of Oracle Fusion Middleware.

> **Note:**
>
> If your domain has the `WLSSchemaDataSource` data source, then you will need to verify which database user is assigned to it. If `<`*PREFIX*`>_WLS_RUNTIME` is assigned to it, then you need to change that to `<`*PREFIX*`>_WLS` . For more information, see Verify the Database User for the WLSSchemaDataSource Data Source.

> **Note:**
>
> As of 14c (14.1.2.0.0) the following schema changes have been made to help you prepare for an optional zero downtime upgrade to a future release:
>
> - Schemas created prior to 14c (14.1.2.0.0) with editions disabled and then upgraded to 14c (14.1.2.0.0) will become editions enabled.
> - Schemas created in 14c (14.1.2.0.0) will be created with editions enabled.

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.

Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 14c (14.1.2.0.0).

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 set the character encoding, run the following:
>
> UNIX operating systems:
>
> ```
> export UA_PROPERTIES="-Dfile.encoding=UTF-8 ${UA_PROPERTIES}"
> ```
>
> Windows operating systems:
>
> ```
> set UA_PROPERTIES=-Dfile.encoding=UTF-8 %UA_PROPERTIES%
> ```

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`

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

Upgrade Assistant Parameters

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

Table 4-4 Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-readiness`	Required for readiness checks **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. Schemas and configurations are checked. 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. 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. 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: • `TRACE` • `NOTIFICATION` • `WARNING` • `ERROR` • `INCIDENT_ERROR` The default logging level is `NOTIFICATION`. 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 4-4 (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. The default locations are: (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/logs` *ORACLE_HOME*`/oracle_common/upgrade/temp` (Windows) *ORACLE_HOME*`\oracle_common\upgrade\logs` *ORACLE_HOME*`\oracle_common\upgrade\temp`
`-help`	Optional	Displays all of the command-line options.

Upgrading the Product Schemas Using the Upgrade Assistant

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

Caution: Complete all required prerequisites before you begin the upgrade. The upgrade may fail if you do not complete the tasks described in Preparing to Upgrade Oracle HTTP Server. For example, as of 14.1.2.0.0 if your existing domain has a `WLSSchemaDataSource` data source, then you must verify which database user is assigned to it. If `<`*PREFIX*`>_WLS_RUNTIME` is assigned to it, then you need to change that to `<`*PREFIX*`>_WLS`. For more information see Verify the Database User for the WLSSchemaDataSource Data Source.

To upgrade schemas:

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 All Schemas screen, select the following option:

   - **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.

- **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 14c (14.1.2.0.0) components. Do not upgrade schemas that are currently being used to support components that are not included in Oracle Fusion Middleware 14c (14.1.2.0.0).

On the Domain Directory screen, enter the 12*c* WebLogic domain directory. Click **Browse** and use the navigation tree to select the 12*c* WebLogic domain directory. The Upgrade Assistant requires the 12*c* domain location to access the jps-config.xml file.

Click **Next**.

3. The Component List screen provides a list of installed Oracle Fusion Middleware components that have schemas that can be upgraded. When you select a component, the schemas and any dependencies are automatically selected.
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 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 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.

6. On the Upgrade Summary screen, review the summary of the schemas that will be upgraded and/or created.

   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 **Next** .

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 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**.

8. 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 *NEW_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. Be sure to replace <*PREFIX*> with your schema prefix.

```
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, EDITION NAME, VERSION, STATUS, UPGRADED FROM
SCHEMA_VERSION_REGISTRY where owner like '<PREFIX>_%';
```

In the query result:

- Verify that the `EDITION NAME` column appears as `ORA$BASE`.
- 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 14.1.2.0.0.

> **Note:**
>
> 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.

About Reconfiguring the Domain

Run the Reconfiguration Wizard to reconfigure your domain component configurations to 14c (14.1.2.0.0).

> **Note:**
>
> If the source is a clustered environment, run the Reconfiguration Wizard on the primary node only.

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.

Backing Up the Domain

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

1. Create a backup of the domain directory.
2. 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 copy 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. Sign in to the system on which the domain resides.
2. Open the command shell (on UNIX operating systems) or open a command prompt window (on Windows operating systems).
3. Go to the `oracle_common/common/bin` directory:
   - (UNIX) *`NEW_ORACLE_HOME`*`/oracle_common/common/bin`
   - (Windows) *`NEW_ORACLE_HOME`*`\oracle_common\commom\bin`
4. 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.

> **Note:**
>
> When you run this command, the following error message might appear to indicate that the default cache directory is not valid:
>
> ```
> *sys-package-mgr*: can't create package cache dir
> ```
>
> You can change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`. For example:
>
> ```
> CONFIG_JVM_ARGS=-Dpython.cachedir=valid_directory
> ```

Reconfiguring the Domain with the Reconfiguration Wizard

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

To reconfigure the domain:

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).
- Scripts and other 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 14c (14.1.2.0.0) is 17.0.12 and later. Click **Next**.

> **Note:**
>
> You cannot change the **Domain Mode** at this stage. Your domain will retain its pre-upgrade domain mode. If you want to change the domain to secure mode, then after the upgrade see Changing Domain Mode Post Upgrade.

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

4. On the JDBC Data Sources screen, configure the JDBC data sources defined in your domain source.

   The JDBC data sources associated with the products for which you are creating the domain are listed in the lower half of the screen. A JDBC data source contains a pool of database connections that are created when the data source instance is created, deployed or targeted, or at server startup. Applications look up a data source on the JNDI tree, and then request a connection. When the applications no longer need the connections, they return the connections to the connection pool in the data source.

   From the **Data Source Name** drop-down list, select the data source(s) for which you want to specify the settings. The values that you specify are displayed in the appropriate columns in the data source list, for the selected data source.

   For Oracle RAC Configuration for data sources, you can select one of the three options:

   - Convert to GridLink
   - Convert to RAC multi data source
   - Don’t convert

   For more information about each option, click **Help**.

   After specifying the details, click **Next**.

   If you do not select any data sources on the JDBC Data Sources screen, the following warning displays:

   ```
   Missing Driver

   Click Ok to proceed without verification, click Cancel to return to
   the JDBC Data Sources page.
   ```

   In this case, if you click **Ok**, the data sources are not verified.

5. On the JDBC Data Sources Test screen, select the check box for the data source connection you configured on the JDBC Data Sources screen and click **Test Selected Connections** to test the data source connection.

> **Note:**
>
> To test the database connections, the database to which you are connecting must be running. If you do not want to test the connections at this time, do not select any data sources. Click **Next** to continue.

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

   Note: <*PREFIX*> is the RCU schema prefix of the 12.2.1.4 domain that is being upgraded.

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

   The Reconfiguration Wizard uses this connection to automatically update 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.

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

7. 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**.

8. The Node Manager screen is only displayed if the domain you are reconfiguring is currently using a per host Node Manager.

   On the Node Manager screen, select the Node Manager configuration to use for the reconfigured domain. The resulting configuration depends on the combination of options you select for **Node Manager Type** and **Node Manager Configuration**.

Table 4-5 Field Descriptions for Node Manager Screen

Option	Description
Per Domain Default Location	If you select this option, the Node Manager home is redefined to *DOMAIN_NAME*/nodemanager and you cannot edit the Node Manager home.
Per Domain Custom Location	Select this option if you want the per domain Node Manager configuration files to be created in a specific location for this domain. Specify the directory in the **Node Manager Home** field, or click **Browse** to use the navigation tree to select the location. The specified directory must be empty. The nodemanager.properties and nodemanager.domains files are created in this directory.

Table 4-5 (Cont.) Field Descriptions for Node Manager Screen

Option	Description
Node Manager Home	If you selected the **Per Domain Custom Location** option, click **Browse** to navigate to the directory location that you want to use to store the per domain Node Manager configuration.
Manual Node Manager Setup	If you select this option, creation of the Node Manager configuration for the domain is skipped (all remaining fields cannot be modified), and if you want to use Node Manager in the domain, you must manually configure Node Manager as described in Completing the Node Manager Configuration. The reconfigured domain will still use a per host Node Manager configuration. You should also select this option if your existing domain is not configured to use Node Manager and you do not want to use Node Manager in the reconfigured domain. For more information about Node Manager configuration, see Administering Node Manager for Oracle WebLogic Server.
Node Manager Configuration	Select one of the following two options. These fields are not available if you selected **Manual Node Manager Setup**.
Create New Configuration	A per domain Node Manager configuration will be automatically created for the reconfigured domain using default settings in nodemanager.properties. If necessary, you can modify nodemanager.properties after the domain has been successfully reconfigured.
Migrate Existing Configuration	The existing per host Node Manager configuration will be migrated to a per domain configuration for the reconfigured domain. This does not include environment-specific settings for ListenAddress, ListenPort, StartScriptName, JavaHome, and LogFile.
Node Manager Home	If you selected the **Migrate Existing Configuration** option, enter or browse to the Node Manager home directory that you want to migrate to the reconfigured domain.

Table 4-5 (Cont.) Field Descriptions for Node Manager Screen

Option	Description

Apply Oracle Recommended Defaults	If you selected the **Migrate Existing Configuration** option, select this check box if you want to use Oracle-recommended defaults in the nodemanager.properties file. Deselect this check box if you want to continue using the settings in the nodemanager.properties file being migrated.
	Oracle-recommended properties with default values are as follows:
	`LogLimit=0`
	`AuthenticationEnabled=true`
	`LogLevel=INFO`
	`DomainsFileEnabled=true`
	`NativeVersionEnabled=true`
	`LogToStderr=true`
	`SecureListener=true`
	`LogCount=1`
	`StopScriptEnabled=false`
	`QuitEnabled=false`
	`LogAppend=true`
	`StateCheckInterval=500`
	`CrashRecoveryEnabled=false`
	`StartScriptEnabled=true`
	`LogFormatter=weblogic.nodemanager.server.LogFormatter`
	`ListenBacklog=50`

Node Manager Credentials: Username, Password	Specify the username and password that you want to use to start Node Manager in the reconfigured domain.

9. On the Advanced Configuration screen, you can select all categories for which you want to perform advanced configuration. For each category you select, the appropriate configuration screen is displayed to allow you to perform advanced configuration.

> **Note:**
>
> The categories that are listed on the Advanced Configuration screen depend on the resources defined in the templates you selected for the domain.

For this upgrade, select none of the options and click **Next**.

10. 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.

11. 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**.

12. 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 14c (14.1.2.0.0).

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 set the character encoding, run the following:
>
> UNIX operating systems:
>
> ```
> export UA_PROPERTIES="-Dfile.encoding=UTF-8 ${UA_PROPERTIES}"
> ```
>
> Windows operating systems:
>
> ```
> set UA_PROPERTIES=-Dfile.encoding=UTF-8 %UA_PROPERTIES%
> ```

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`

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

Upgrade Assistant Parameters

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

Table 4-6 Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-readiness`	Required for readiness checks **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. Schemas and configurations are checked. 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. 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. Do not specify this parameter if you have specified the `-readiness` parameter.

Table 4-6 (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-logLevel` *attribute*	Optional	Sets the logging level, specifying one of the following attributes: • `TRACE` • `NOTIFICATION` • `WARNING` • `ERROR` • `INCIDENT_ERROR` The default logging level is `NOTIFICATION`. 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. The default locations are: (UNIX) *ORACLE_HOME*`/oracle_common/upgrade/logs` *ORACLE_HOME*`/oracle_common/upgrade/temp` (Windows) *ORACLE_HOME*`\oracle_common\upgrade\logs` *ORACLE_HOME*`\oracle_common\upgrade\temp`
`-help`	Optional	Displays all of the command-line options.

Upgrading Domain Components Using the Upgrade Assistant

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 14c (14.1.2.0.0), 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 WebLogic 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 *Planning an Upgrade of Oracle Fusion Middleware* 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.

> **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.

Verifying the Domain-Specific-Component Configurations Upgrade

To verify that the domain-specific-component configurations upgrade was successful, sign in to the Remote Console and verify that the version numbers for each upgraded component is 14.1.2.0.0.

> **Note:**
>
> Before you can access the Hosted WebLogic Remote Console, you must deploy the hosted WebLogic Remote Console. For more information, see the Remote Console Online Help.

To sign in to the Remote Console, go to: `http://hostname:port/rconsole` or for HTTPS, `https://hostname:port/rconsole`.

> **Note:**
>
> After a successful upgrade, make sure you run the administration tools from the new 14c (14.1.2.0.0) 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.
>
> If you created the FMW user to run the Upgrade Assistant, ensure that you delete the account after verifying your upgrade was successful.

Starting Servers and Processes

After a successful upgrade, restart 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 processes using the WLST command-line utility or a script. You can also use the Oracle Fusion Middleware Control and the Oracle WebLogic Server Remote Console. See Starting and Stopping Administration and Managed Servers and Node Manager.
>
> As of release 14c (14.1.2.0.0), the WebLogic Server Administration Console has been removed. For comparable functionality, you should use the WebLogic Remote Console. For more information, see Oracle WebLogic Remote Console.

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

> **Note:**
>
> Depending on your existing security settings, you may need to perform additional configuration before you can manage a domain with secured production mode enabled. For more information, see Connecting to the Administration Server using WebLogic Remote Console
>
> .

Step 1: Start the Administration Server

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

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

> **Note:**
>
> When using secured production mode, you must provide additional parameters to start the Administration Server. See Connecting to the Administration Server using WLST in *Administering Security for Oracle WebLogic Server*.

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

Step 2: Start Node Manager

To start Node Manager, use the `startNodeManager` script:

- (UNIX) *NEW_DOMAIN_HOME*`/bin/startNodeManager.sh`
- (Windows) *NEW_DOMAIN_HOME*`\bin\startNodeManager.cmd`

Step 3: Start Any Managed Servers

To start a WebLogic Server Managed Server, use the `startManagedWebLogic` script:

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

> **Note:**
>
> When using secured production mode, you must provide additional parameters to start the Managed Servers. See Starting Managed Servers using a Start Script in *Administering Security for Oracle WebLogic Server*.

> **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.

Step 4: Start System Components

To start system components, such as Oracle HTTP Server, use the `startComponent` script:

- (UNIX) *NEW_DOMAIN_HOME*`/bin/startComponent.sh` *component_name*
- (Windows) *NEW_DOMAIN_HOME*`\bin\startComponent.cmd` *component_name*

You can start system components in any order.

Starting the Oracle HTTP Server

You can use Enterprise Manager Fusion Middleware Control to start, stop, and restart Oracle HTTP Server.

You can start the Oracle HTTP Server with the Node Manager by following the procedure mentioned in Running Oracle HTTP Server Remotely in *Administering Oracle HTTP Server*.

To start an Oracle HTTP Server using Enterprise Manager Fusion Middleware Control:

1. Go to the Oracle HTTP Server home page.
2. Log in to the Enterprise Manager Console page with the proper credentials.
3. From the Oracle HTTP Server menu:
   a. Select **Control**.
   b. Select **Start Up** from the **Control** menu.

   Or
4. From the Target Navigation tree:
   a. Right-click the Oracle HTTP Server instance you want to start.
   b. Select **Control**.
   c. Select **Start Up** from the **Control** menu.

Verifying the Upgrade

If you can log in to the WebLogic Server Remote Console, the Enterprise Manager Fusion Middleware Control, and the OHS Home Page, then your upgrade is successful.

To verify the upgrade:

1. Access the WebLogic Server Remote Console:

   For HTTP: http://*hostname*:port/rconsole

   For HTTPS: https://*hostname*:port/rconsole
2. Access the OHS default home page by using the following URL:

   http://*hostname*:port
3. Access Enterprise Manager Fusion Middleware Control by using the following URL:

   http://*hostname*:port/em

Post-Upgrade Tasks

Perform the post upgrade tasks that apply to your deployment.

> **Note:**
>
> Perform only those tasks that are applicable for your specific configuration.

Changing Domain Mode Post Upgrade

After the upgrade, your domain retains its original pre-upgrade domain security mode settings. If you want to change the domain mode, to enable enhanced security, for example, you must explicitly change the settings using the WebLogic Remote Console or by modifying the `DomainMBean`.

If your domain is currently set to Production Mode, and you want to enable added security, then after the upgrade use the WebLogic Remote Console to change the domain mode and enable the Secured Production Mode. Change the Domain Mode in *Oracle WebLogic Remote Console Online Help*.

> **Caution:**
>
> Changes to the domain mode require a full domain restart - a rolling restart is not sufficient. You must stop all managed servers before you attempt to change the domain mode.

When upgrading a domain to 14c (14.1.2.0.0), if there is no explicit secure mode setting, then the Reconfiguration Wizard will explicitly set secure mode to *disabled* in the upgraded domain. This is to preserve the behavior that was present in the original domain. If there is an explicit secure mode setting, it will be preserved in the upgraded domain. For more information, see Understand How Domain Mode Affects the Default Security Configuration in *Securing a Production Environment for Oracle WebLogic Server*.

> **Note:**
>
> Secured Production Mode enforces more restrictive and stringent security settings to ensure less vulnerability to threats. To make sure that your domain is secure, after enabling Secured Production Mode, you will have to choose the security configuration options that are appropriate for the environment in which the domain runs, such as obtaining and storing certificates, protecting user accounts, and securing the network on which the domain runs. If these options are not properly configured, you will be blocked from using WebLogic Server.
>
> After you have created your WebLogic domain, several key steps remain to ensure its integrity such as selecting appropriate security configurations. For more information, see Securing the Domain After You Have Created It in *Administering Security for Oracle WebLogic Server*.

Removing Version Information from htdocs Folder

This step is only required if you have a managed (collocated) environment that uses an OHS instance created using online WLST command `ohs_createInstance`.

The script `version_suppressor.pl` should be executed to remove the version information from files located in the /htdocs folder. This script has to be run from the host where the runtime directory of the managed OHS instance exists.

The script is located in the $ORACLE_HOME/ohs/common/bin directory.

5

Upgrading a Clustered Environment

Describes the process of upgrading to a multi-node environment and performing post-upgrade configuration tasks.

> **Note:**
>
> If the Oracle wallet configured with OHS is not located on the same machine where the Upgrade Assistant is being invoked, then the wallets cannot be taken care of during the upgrade process. You must perform the following steps to ensure that the wallets are available.

Propagating Domain Configuration to Another Host

After verifying that the upgrade was successful, use these steps to propagate the newly upgraded files to another host.

After you have completed your single node upgrade on `HOST1`, use these steps to propagate the newly upgraded files to another node (in this example the secondary host is called `HOST2`).

Executing the pack command on the server where the Admin Server and one of the Managed Servers is installed.

In our sample topology, you would execute the following on `HOST1`:

`cd` */14c_ORACLE_HOME*`/oracle_common/common/bin`

`./pack.sh -domain=/`*12c_DOMAIN_HOME* `-template=domainupgradetemplate.jar -template_name=domainupgradetemplate -managed=true`

In this example:

- *14c_ORACLE_HOME* refers the actual path to the 14*c* Oracle home directory (the installation directory for the 14c (14.1.2.0.0)bits).
- Replace *12c_DOMAIN_HOME* with the actual path to the upgraded domain directory.
- `domainupgradetemplate.jar` is a sample name for the jar file you are creating, which will contain the domain configuration files.
- `domainupgradetemplate` is the name assigned to the domain template file.
- By default, the `domainupgradetemplate` is created in the current directory where you ran the pack command. In this example, it would be created in the following directory, but you can specify a full path for the template jar file as part of the `-template` argument to the pack command:

  ORACLE_COMMON_HOME`/common/bin/`

The `pack` command creates a template archive (.jar) file that contains a snapshot of either an entire domain or a subset of a domain. You can use a template that contains a subset of a domain to create a Managed Server domain directory hierarchy on a remote machine.

Executing the unpack Command from the 12c Oracle Home on HOST2.

Make sure that the Administration and Managed Servers are still stopped and then execute the `unpack` command to create a full domain (or a subset of a domain) used for a Managed Server domain directory on the remote machine. You may use unpack only with a template compatible with your current installation.

> **Note:**
>
> Do not attempt to unpack the domain on top of an existing domain. Oracle recommends that you unpack the contents of the domain upgrade template jar file into a new domain location.
> It is important to note that even if you use the `-overwrite_domain=true` argument when unpacking the domain, the contents of the existing domain will remain in place and will cause issues with when starting the servers. For this reason, Oracle recommends that you unpack the domain template jar file into a new location, or, manually delete the contents of the existing location before you unpack.

A sample unpack command code snippet is shown below.

cd */12c_ORACLE_HOME/*`oracle_common/common/bin`

`./unpack.sh -template=domainupgradetemplate.jar - domain=`*NEW_DOMAIN_LOCATION*

In this example:

- *12c_ORACLE_HOME* refers the actual path to the 12c Oracle home directory, the installation directory for the 14c (14.1.2.0.0).
- Replace *NEW_DOMAIN_LOCATION* with the actual path to the upgraded domain directory.
- `domainupgradetemplate.jar` is a sample name for the jar file you are creating, which will contain the domain configuration files.
- `domainupgradetemplate` is the name assigned to the domain template file.

Copying the template file created on HOST 1 to HOST2.

After you perform a complete upgrade of your deployment on HOST1, and you have completed any post-upgrade configurations that apply to your new environment, you must copy the domain template to HOST2.
Use the following command to copy from HOST1 the domain upgrade template JAR file created during the upgrade.

```
scp domaintemplate.jar company@HOST2:14c_ORACLE_HOME/oracle_common/common/bin
```

Completing the following verification steps after the unpack.

1. Verify that settings for WLS_HOME and ORACLE_HOME located in the setDomainEnv.sh script from the 12*c* domain are pointing to 14c (14.1.2.0.0).
2. Start the Node Manager, WebLogic Administration Server, and the Managed Servers on HOST1 and HOST2 in the following order:
   a. On HOST1 and HOST2, start the Node Manager.
   b. On HOST1, start the WebLogic Administration Server.
   c. On HOST1 and HOST2, start the Managed Servers.

For more information, see Starting Servers and Processes. Carefully review the order in which Managed Servers should be started.

A

Replacing Certificate Signed Using MD5 Algorithm with Certificate Signed Using SHA-2 Algorithm

Oracle strongly recommends that you refrain from using a certificate signed with Message Digest 5 Algorithm (MD5), because the security of MD5 algorithm has been compromised. Therefore, you must replace the certificate signed using MD5 algorithm with a certificate signed with Secure Hashing Algorithm 2 (SHA-2). By default, certificates signed using MD5 algorithm are no longer supported in Oracle HTTP Server.

How to Check whether Certificate Signed with MD5 Algorithm is Present in the Wallet?

You can use the orapki utility to display whether your wallet contains a certificate signed with MD5 algorithm.

To check whether the MD5 certificate is present in your wallet:

1. Locate orapki and the keytool path:

   (orapki) *ORACLE_HOME*/oracle_common/bin/orapki

   (Keytool) *ORACLE_HOME*/oracle_common/jdk/jre/bin

2. Set the JAVA_HOME environment variable as follows:

   ```
   export JAVA_HOME=ORACLE_HOME/oracle_common/jdk/jre/
   ```

3. Enter the following command to display the contents of the wallet:

   ```
   orapki wallet display -wallet wallet_location
   ```

   Sample command:

   ```
   ORACLE_HOME/oracle_common/bin/orapki wallet display -wallet /
   scratch/ohs14.1.2.0.0_install/walletohs2/
   ```

   Output:

   ```
   Oracle PKI Tool : Version 14.1.2.0.0
   Copyright (c) 2004, 2024, Oracle and/or its affiliates. All rights
   reserved.

   Requested Certificates:
   User Certificates:
   Subject: CN=www.xyx.com,C=IN
   Trusted Certificates:
   Subject: CN=www.xyx.com,C=IN
   ```

4. Identify the domain name (DN) of the certificate present in the wallet. In this example, Subject of User Certificates is the DN of the certificate present in the wallet; which is `CN=www.xyx.com,C=IN`.

5. Export the certificate present in the wallet as shown in the following example:

   ```
   orapki wallet export -wallet wallet_Location -dn 'DN_string' -cert
   certificate_file
   ```

   Sample command:

   ```
   ORACLE_HOME/oracle_common/bin/orapki wallet export -wallet /
   scratch/ohs14.1.2.0.0_install/walletohs2/ -
   dn  'CN=www.xyx.com,C=IN' -cert wallet.cert
   ```

6. Use the keytool to check the signature algorithm used to sign the *certificate_file* that you exported in the previous step by entering the following command:

   ```
   ORACLE_HOME/oracle_common/jdk/jre/bin/keytool -printcert -file
   certificate_file
   ```

   Sample command:

   ```
   /scratch/ohs14.1.2.0.0_install/oracle_common/jdk/jre/bin/keytool -
   printcert -file wallet.cert
   ```

   Output:

   ```
   Owner: CN="Self-Signed Certificate for ohs3 ", OU=OAS, O=ORACLE,
   L=REDWOODSHORES, ST=CA, C=US
   Issuer: CN="Self-Signed Certificate for ohs3 ", OU=OAS, O=ORACLE,
   L=REDWOODSHORES, ST=CA, C=US
   Serial number: cd7081c47adb9ff867da01e3fe383e0f
   Valid from: Fri Jul 01 04:21:40 PDT 2024 until: Sat Jun 19 04:21:40 PDT
   2066
   Certificate fingerprints:
    MD5: 89:4D:C4:B4:28:7E:D4:0A:93:CA:E4:97:0C:F0:CE:86
    SHA1: 53:FB:D6:58:0D:09:85:E9:6D:55:E2:92:E7:87:08:B8:65:92:15:9E
    Signature algorithm name: MD5withRSA
    Version: 1
   ```

   Signature algorithm name: MD5withRSA implies that MD5 algorithm is being used to sign the certificate present in the wallet.

Removing Certificate Signed with MD5 Algorithm from the Wallet

If MDS is present in your wallet, the signature algorithm name is displayed as `MDSwithRSA`. You must replace this certificate with certificate signed using SHA2 algorithm.

To remove a certificate signed with MD5 algorithm from your wallet:

1. Enter the following command to remove a user certificate signed using MD5 algorithm:

   ```
   orapki wallet  remove -wallet wallet_location -dn 'DN_string' -
   user_cert -auto_login_only
   ```

   Sample command:

   ```
   /scratch/ohs14.1.2.0.0_install/oracle_common/bin/orapki wallet
   remove -wallet /scratch/ohs14.1.2.0.0_install/walletohs2/ -
   dn 'CN=www.xyx.com,C=IN' -user_cert -auto_login_only
   ```

Output:

```
Oracle PKI Tool : Version 14c (14.1.2.0.0)
Copyright (c) 2016, 2024, Oracle and/or its affiliates. All rights
reserved.
```

2. If the user certificate is a self-signed certificate, you need to remove it from the *trusted certificate* and the *requested certificate* list by entering the following commands:

   ```
   orapki wallet remove -wallet wallet_location -dn 'DN_string' -
   trusted_cert -auto_login_only
   ```

   ```
   orapki wallet remove -wallet wallet_location -dn 'DN_string' -
   cert_req -auto_login_only
   ```

   Sample command:

   ```
   /scratch/ohs14.1.2.0.0_install/oracle_common/bin/orapki wallet
   remove -wallet /scratch/ohs14.1.2.0.0_install/walletohs2/ -dn
   'CN=www.xyx.com,C=IN' -trusted_cert -auto_login_only
   ```

   Output:

   ```
   Oracle PKI Tool : Version 14.1.2.0.0
   Copyright (c) 2024, 2016, Oracle and/or its affiliates. All rights
   reserved.
   ```

3. If a wallet has trusted certificate or a certificate request which is signed using MD5 algorithm, remove that certificate by entering the following commands:

   ```
   orapki wallet remove -wallet wallet_location -dn 'DN_string' -
   trusted_cert -auto_login_only remove cert request
   ```

   ```
   orapki wallet remove -wallet wallet_location -dn 'DN_string' -
   cert_req -auto_login_only
   ```

Adding Certificate Signed with SHA-2 Algorithm to the Wallet

If you are using CA-signed user certificate that is signed with MD5 algorithm, contact your certificate authority to a get a new user certificate signed with SHA-2 algorithm and import it in to the wallet.

To add self-signed certificate, signed using SHA-2 algorithm:

1. If certificate is self-signed, enter the following command to add self-signed certificate signed using SHA-2 algorithm:

   ```
   orapki wallet add -wallet wallet_Location  -dn 'DN_String' -keysize
   2048 -sign_alg sha256  -self_signed -validity 9125 -auto_login_only
   ```

   Sample command:

   ```
   /scratch/ohs14.1.2.0.0_install/oracle_common/bin/orapki wallet add
   -wallet /scratch/ohs14.1.2.0.0_install/walletohs2/ -
   dn 'CN=www.xyx.com,C=IN' -keysize 2048 -sign_alg sha256  -
   self_signed -validity 9125 -auto_login_only
   ```

2. Add trusted certificate signed using SHA-2 algorithm to wallet by entering the following command:

   ```
   orapki -wallet add -wallet wallet_location -trusted_cert -cert
   certificate_file -auto_login_only
   ```

3. Add user certificate signed using SHA-2 algorithm to wallet by entering the following command:

   ```
   orapki -wallet add -wallet wallet_location -user_cert -cert
   certificate_file -auto_login_only
   ```

The operation is successful if you see signature algorithm name equal to *SHA256withRSA* when you print your certificate file. Following is a sample output:

```
Owner: CN=www.xyx.com, C=IN
Issuer: CN=www.xyx.com, C=IN
Serial number: f689ec6986c70f973138962eb2f0e5f9
Valid from: Wed May 11 04:01:24 PDT 2024 until: Sat Oct 27 04:01:24 PDT 2025
Certificate fingerprints:
   MD5:  D7:0F:CB:00:A7:04:33:DA:2F:8A:AD:C9:2A:9E:DF:26
   SHA1: D4:6C:51:DB:53:B5:F5:C7:60:8D:8B:95:68:E6:B0:5E:E8:00:ED:DF
   SHA256:
B1:EF:73:98:EA:6A:1A:60:FF:1F:10:89:8C:B8:16:63:71:03:1B:6E:38:D1:2D:AE:E9:BD:
3E:13:BE:AF:A0:76
   Signature algorithm name: SHA256withRSA
   Version: 1
```

Enabling Support for Certificate Signed with MD5 Algorithm in your 14.1.2.0.0 Oracle HTTP Server Deployment

By default, support of certificate signed with MD5 algorithm has been removed because the security of MD5 algorithm is severely compromised. If you still want to use certificate signed using MD5 algorithm, you can enable the support for a certificate signed with MD5 algorithm by following the procedure in this section. However, enabling support for certificates signed using MD5 algorithm is not recommended.

To enable support for a certificate signed with MD5 algorithm:

1. For Standalone Oracle HTTP Server deployment:
   a. Stop the Oracle HTTP Server instance and the Node Manager.
   b. Change to the following staging directory:

      (UNIX) *ORACLE_HOME*`/user_projects/domains/base_domain/config/fmwconfig/components/OHS/ohs1/`

      (Windows) *ORACLE_HOME*`\user_projects\domains\base_domain\config\fmwconfig\components\OHS\ohs1\`

   c. Open the `ohs.plugins.nodemanager.properties` file in edit mode and add the following line:

      ```
      environment.ORACLE_SSL_ALLOW_MD5_CERT_SIGNATURES = 1
      ```

   d. Restart the Node Manager and the Oracle HTTP Server instance.
2. For Managed Oracle HTTP Server deployment:
   a. Stop the Oracle HTTP Server instance, Node Manager, and the WebLogic Server.
   b. Change to the following staging directory:

      (UNIX) *ORACLE_HOME*`/user_projects/domains/base_domain/config/fmwconfig/components/OHS/ohs1/`

(Windows)
ORACLE_HOME\user_projects\domains\base_domain\config\fmwconfig\components\OHS\ohs1\

c. Open the `ohs.plugins.nodemanager.properties` file in edit mode and add the following line:

```
environment.ORACLE_SSL_ALLOW_MD5_CERT_SIGNATURES = 1
```

d. Restart the WebLogic Server, Node Manager, and the Oracle HTTP Server instance.

You must perform this procedure for each Oracle HTTP Server instance.

B

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/jdk17.0.12.

- 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.

The getProperty.sh|cmd and setProperty.sh|cmd scripts are located in the following location:

(Linux) *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 jdk17.0.12.

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:

   (Linux) *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:

   (Linux) *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:

   (Linux) *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 jdk17.0.12.

Updating the JDK Location in an Existing Domain Home

You must search the references to the current JDK manually, and replace those instances with the location of the new JDK.

You can use the `grep` or `findstr` commands to search for the jdk-related references.

You'll likely be required to update the location of JDK in the following three files:

(Linux) *DOMAIN_HOME*/bin/setNMJavaHome.sh

(Windows) *DOMAIN_HOME*\bin\setNMJavaHome.cmd

(Linux) *DOMAIN_HOME*/nodemanager/nodemanager.properties

(Windows)*DOMAIN_HOME*\nodemanager\nodemanager.properties

(Linux) *DOMAIN_HOME*/bin/setDomainEnv.sh

(Windows)*DOMAIN_HOME*\bin\setDomainEnv.cmd