Red Hat JBoss Enterprise Application Platform 7.1

Using the JBoss Server Migration Tool

For Migrating Server Configurations to Red Hat JBoss Enterprise Application Platform 7.1

Last Updated: 2018-08-13

Red Hat JBoss Enterprise Application Platform 7.1 Using the JBoss Server Migration Tool

For Migrating Server Configurations to Red Hat JBoss Enterprise Application Platform 7.1

Legal Notice

Copyright © 2018 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at
http://creativecommons.org/licenses/by-sa/3.0/
. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

This guide describes how to use the JBoss Server Migration Tool to migrate your application server configuration from a previous release of JBoss EAP to JBoss EAP 7.1.

Table of Contents

CHAPTER 1. INTRODUCTION **6**
1.1. ABOUT RED HAT JBOSS ENTERPRISE APPLICATION PLATFORM 7 6
1.2. ABOUT THE JBOSS SERVER MIGRATION TOOL 6
1.3. ABOUT THE USE OF EAP_HOME IN THIS DOCUMENT 7
JBoss EAP Installation Path 7

CHAPTER 2. BEFORE YOU BEGIN **8**
2.1. UPGRADE YOUR CURRENT SOURCE SERVER INSTALLATION 8
2.2. RUN WITH A CLEAN TARGET SERVER INSTALLATION 8
2.3. CUSTOMIZE THE MIGRATION 8

CHAPTER 3. RUNNING THE JBOSS SERVER MIGRATION TOOL **9**
3.1. RUN THE JBOSS SERVER MIGRATION TOOL IN INTERACTIVE MODE 9
3.2. RUN THE JBOSS SERVER MIGRATION TOOL IN NON-INTERACTIVE MODE 11

CHAPTER 4. REVIEWING THE RESULTS OF JBOSS SERVER MIGRATION TOOL EXECUTION **12**
4.1. REVIEW THE MIGRATED CONFIGURATION FILES 12
4.2. TRACKING MIGRATION TASK EXECUTION 13
4.3. REVIEW THE TASK SUMMARY LOG 14
4.4. REVIEW THE JBOSS SERVER MIGRATION TOOL REPORTS 14
4.4.1. JBoss Server Migration Tool HTML Report 14
4.4.2. JBoss Server Migration Tool XML Report 14

CHAPTER 5. CONFIGURING THE JBOSS SERVER MIGRATION TOOL **16**
5.1. CONFIGURING THE JBOSS SERVER MIGRATION TOOL PROPERTIES 16
5.1.1. Configure the JBoss Server Migration Tool Using the Tool Properties 16
5.1.2. Configure the JBoss Server Migration Tool Using User Properties 16
5.1.3. Configure the JBoss Server Migration Tool Using System Properties 16
5.2. CONFIGURING LOGGING FOR JBOSS SERVER MIGRATION TOOL 17
5.3. CONFIGURING MODULES MIGRATION 17
5.3.1. Modules Environment Properties 17
5.3.2. Configuring Modules Properties 18
5.4. CONFIGURING REPORTING FOR JBOSS SERVER MIGRATION TOOL 18
5.4.1. Configuring the Task Summary Log 18
5.4.2. Configuring the HTML Report 19
5.4.3. Configuring the XML Report 19
5.5. CONFIGURING THE MIGRATION OF THE STANDALONE SERVER CONFIGURATION 19
5.6. CONFIGURING THE MIGRATION OF A MANAGED DOMAIN CONFIGURATION 20
5.7. CONFIGURE THE MIGRATION TASKS PERFORMED BY THE JBOSS SERVER MIGRATION TOOL 21

CHAPTER 6. MIGRATING JBOSS EAP 6.4 CONFIGURATIONS TO JBOSS EAP 7.1 . . . . . **22**
6.1. MIGRATING A JBOSS EAP 6.4 STANDALONE SERVER TO JBOSS EAP 7.1 22
6.1.1. Remove Unsupported Subsystems 23
6.1.2. Migrate Referenced Modules 23
6.1.3. Migrate Referenced Paths 24
6.1.4. Update the Transactions Subsystem 24
6.1.5. Migrate the Jacorb Subsystem 24
6.1.6. Migrate the Web Subsystem 24
6.1.7. Migrate the Messaging Subsystem 24
6.1.8. Update the Infinispan Subsystem 25
6.1.9. Update the EE Subsystem 25
Configuring Concurrency Utilities in the EE Subsystem 26
Configuring Default Resources in the EE Subsystem 26

6.1.10. Update the EJB 3 Subsystem 27
6.1.11. Update the JGroups Subsystem 28
6.1.12. Update the Remoting Subsystem 28
6.1.13. Update the Undertow Subsystem 28
6.1.14. Update the Messaging-ActiveMQ Subsystem 29
6.1.15. Add the Batch JBeret Subsystem 29
6.1.16. Add the Core Management Subsystem 30
6.1.17. Add the Elytron Subsystem 30
6.1.18. Add the Request Controller Subsystem 30
6.1.19. Add the Security Manager Subsystem 30
6.1.20. Add the Singleton Subsystem 30
6.1.21. Set Up HTTP Upgrade Management 30
6.1.22. Set Up the Private Interface 31
6.1.23. Add Socket Binding Port Expressions 31
6.1.24. Migrate Compatible Security Realms 31
6.1.25. Add the Default SSL Server Identity to the ApplicationRealm 31
6.1.26. Migrate Deployments 32
6.1.26.1. Migrate Persistent Deployments 32
6.1.26.2. Migrate Deployment Scanner Deployments 34
6.1.26.3. Migrate Deployment Overlays 35
6.2. MIGRATING A JBOSS EAP 6.4 MANAGED DOMAIN TO JBOSS EAP 7.1 35
6.2.1. Remove Unsupported Subsystems 36
6.2.2. Migrate Referenced Modules 37
6.2.3. Migrate Referenced Paths 37
6.2.4. Update the Transactions Subsystem 38
6.2.5. Migrate the Jacorb Subsystem 38
6.2.6. Migrate the Web Subsystem 38
6.2.7. Migrate the Messaging Subsystem 38
6.2.8. Update the Infinispan Subsystem 38
6.2.9. Update the EE Subsystem 39
Configuring Concurrency Utilities in the EE Subsystem 40
Configuring Default Resources in the EE Subsystem 40
6.2.10. Update the EJB 3 Subsystem 41
6.2.11. Update the JGroups Subsystem 41
6.2.12. Update the Remoting Subsystem 42
6.2.13. Update the Undertow Subsystem 42
6.2.14. Update the Messaging-ActiveMQ Subsystem 43
6.2.15. Add the Batch JBeret Subsystem 43
6.2.16. Add the Core Management Subsystem 43
6.2.17. Add the Elytron Subsystem 44
6.2.18. Add the Request Controller Subsystem 44
6.2.19. Add the Security Manager Subsystem 44
6.2.20. Add the Singleton Subsystem 44
6.2.21. Update the Unsecure Interface 44
6.2.22. Set Up the Private Interface 44
6.2.23. Add Socket Binding Port Expressions 45
6.2.24. Add Socket Binding Multicast Address Expressions 45
6.2.25. Add the Load Balancer Profile 45
6.2.26. Add Host Excludes 45
6.2.27. Remove the PermGen Attributes from the JVM Configurations 46
6.2.28. Migrate Deployments 46
6.2.28.1. Migrate Persistent Deployments 47
6.2.28.2. Migrate Deployment Overlays 48

6.3. MIGRATING A JBOSS EAP 6.4 HOST CONFIGURATION TO JBOSS EAP 7.1 48
6.3.1. Migrate Referenced Modules 48
6.3.2. Migrate Referenced Paths 49
6.3.3. Add the Core Management Subsystem 49
6.3.4. Add the Elytron Subsystem 49
6.3.5. Add the JMX Subsystem to the Host Configuration 49
6.3.6. Remove the unsecure Interface 49
6.3.7. Set Up HTTP Upgrade Management 50
6.3.8. Remove the PermGen Attributes from the JVM Configurations 50
6.3.9. Migrate Compatible Security Realms 50
6.3.10. Add the Default SSL Server Identity to the ApplicationRealm 50

CHAPTER 7. MIGRATING JBOSS EAP 7.0 CONFIGURATIONS TO JBOSS EAP 7.1 51
7.1. MIGRATING A JBOSS EAP 7.0 STANDALONE SERVER TO JBOSS EAP 7.1 51
7.1.1. Remove Unsupported Subsystems 51
7.1.2. Migrate Referenced Modules 51
7.1.3. Migrate Referenced Paths 52
7.1.4. Update the Infinispan Subsystem 52
7.1.5. Update the Undertow Subsystem 53
7.1.6. Add the Core Management Subsystem 54
7.1.7. Add the Elytron Subsystem 54
7.1.8. Add Socket Binding Multicast Address Expressions 54
7.1.9. Migrate Compatible Security Realms 54
7.1.10. Add the Default SSL Server Identity to the ApplicationRealm 54
7.1.11. Migrate Deployments 55
7.1.11.1. Migrate Persistent Deployments 55
7.1.11.2. Migrate Deployment Scanner Deployments 57
7.1.11.3. Migrate Deployment Overlays 58
7.2. MIGRATING A JBOSS EAP 7.0 MANAGED DOMAIN TO JBOSS EAP 7.1 58
7.2.1. Remove Unsupported Subsystems 59
7.2.2. Migrate Referenced Modules 59
7.2.3. Migrate Referenced Paths 60
7.2.4. Update the Infinispan Subsystem 60
7.2.5. Update the Undertow Subsystem 60
7.2.6. Add the Core Management Subsystem 61
7.2.7. Add the Elytron Subsystem 61
7.2.8. Add Socket Binding Multicast Address Expressions 62
7.2.9. Add the Load Balancer Profile 62
7.2.10. Add Host Excludes 62
7.2.11. Migrate Deployments 62
7.2.11.1. Migrate Persistent Deployments 63
7.2.11.2. Migrate Deployment Overlays 64
7.3. MIGRATING A JBOSS EAP 7.0 HOST CONFIGURATION TO JBOSS EAP 7.1 64
7.3.1. Migrate Referenced Modules 65
7.3.2. Migrate Referenced Paths 65
7.3.3. Add the Core Management Subsystem 65
7.3.4. Add the Elytron Subsystem 65
7.3.5. Migrate Compatible Security Realms 66
7.3.6. Add the Default SSL Server Identity to the ApplicationRealm 66

APPENDIX A. REFERENCE MATERIAL 67
A.1. EXAMPLE TASK SUMMARY REPORT 67
A.2. EXAMPLE HTML REPORT 67

A.3. EXAMPLE XML REPORT 69

CHAPTER 1. INTRODUCTION

1.1. ABOUT RED HAT JBOSS ENTERPRISE APPLICATION PLATFORM 7

Red Hat JBoss Enterprise Application Platform 7 (JBoss EAP) is a middleware platform built on open standards and compliant with the Java Enterprise Edition 7 specification.

JBoss EAP provides two operating modes for server instances.

Standalone server

The standalone server operating mode represents running JBoss EAP as a single server instance.

Managed domain

The managed domain operating mode allows for the management of multiple JBoss EAP instances from a single control point.

JBoss EAP includes APIs and development frameworks for quickly developing secure and scalable Java EE applications. Many of the APIs and capabilities that are exposed to applications deployed to JBoss EAP servers are organized into subsystems that are configured in the server configuration files. For example, you configure database access information in the **`datasources`** subsystem so that it can be accessed by applications deployed to JBoss EAP standalone servers or managed domains. The introduction of new features and deprecation of other features can require modification of the server configurations from one release of JBoss EAP to another.

For more information about Red Hat JBoss Enterprise Application Platform, see the Product Documentation for JBoss EAP located on the Red Hat Customer Portal.

1.2. ABOUT THE JBOSS SERVER MIGRATION TOOL

Migrating an existing application server configuration from one release to another is a complex task. To plan and execute a successful migration not only requires a complete understanding the current server configuration, but also knowledge of features and changes in the target server configuration. With a manual migration, you generally copy and edit several configuration files, and then make the updates needed to keep the same behavior in the new release. If this is not done correctly, the new server does not work as expected. This is often because some functionality is not supported by the new server.

The JBoss Server Migration Tool is a Java application that automatically migrates JBoss EAP server configurations with minimal or no interaction required. It is the preferred method to update your JBoss EAP server configuration to include the new features and settings in JBoss EAP 7 while keeping your existing configuration. The JBoss Server Migration Tool reads your existing source server configuration files and adds configurations for any new subsystems, updates the existing subsystem configurations with new features, and removes any obsolete subsystem configurations.

The JBoss Server Migration Tool supports the migration of standalone servers and managed domains for the following configurations.

From JBoss EAP 6.4 or JBoss EAP 7.0 to JBoss EAP 7.1

The JBoss Server Migration Tool ships with JBoss EAP 7.1, so there is no separate download or installation required. You run the tool by executing the **`jboss-server-migration`** script located in the ***`EAP_HOME`*`/bin`** directory. For more information about how to run the tool, see Running the JBoss Server Migration Tool in JBoss EAP 7.1.

It is recommended that you use this version of the JBoss Server Migration Tool to migrate your server configuration to JBoss EAP 7.1 as this version of the tool is supported.

From JBoss EAP 6.4 to JBoss EAP 7.0

For migration to JBoss EAP 7.0, you must download the latest binary distribution of the JBoss Server Migration Tool from the wildfly-server-migration GitHub repository. This open source, standalone version of the tool also supports migrating from several versions of the WildFly server to JBoss EAP. For information about how to install and run this version of the tool, see the JBoss Server Migration Tool *User Guide*.

IMPORTANT

The binary distribution of the JBoss Server Migration Tool is not supported. Rather than use this version to migrate your server configuration to JBoss EAP 7.0, it is recommended that you use the supported version of the tool to migrate your server configuration directly to JBoss EAP 7.1 instead.

1.3. ABOUT THE USE OF EAP_HOME IN THIS DOCUMENT

In this document, the variable ***EAP_HOME*** is used to denote the path to the target server installation. Replace this variable with the actual path to your server installation.

NOTE

EAP_HOME is a replaceable variable, not an environment variable. ***JBOSS_HOME*** is the environment variable used in scripts.

JBoss EAP Installation Path

- If you installed JBoss EAP using the ZIP install method, the install directory is the **`jboss-eap-7.1`** directory where you extracted the ZIP archive.
- If you installed JBoss EAP using the RPM install method, the install directory is **`/opt/rh/eap7/root/usr/share/wildfly/`**.
- If you used the installer to install JBoss EAP, the default path for ***EAP_HOME*** is **`${user.home}/EAP-7.1.0`**:
  - For Red Hat Enterprise Linux, Solaris, and HP-UX: **`/home/`*`USER_NAME`*`/EAP-7.1.0/`**
  - For Microsoft Windows: **`C:\Users\`*`USER_NAME`*`\EAP-7.1.0\`**
- If you used the JBoss Developer Studio installer to install and configure the JBoss EAP server, the default path for ***EAP_HOME*** is **`${user.home}/jbdevstudio/runtimes/jboss-eap`**:
  - For Red Hat Enterprise Linux: **`/home/`*`USER_NAME`*`/jbdevstudio/runtimes/jboss-eap/`**
  - For Microsoft Windows: **`C:\Users\`*`USER_NAME`*`\jbdevstudio\runtimes\jboss-eap`** or **`C:\Documents and Settings\`*`USER_NAME`*`\jbdevstudio\runtimes\jboss-eap\`**

CHAPTER 2. BEFORE YOU BEGIN

2.1. UPGRADE YOUR CURRENT SOURCE SERVER INSTALLATION

Before you migrate your current source server configuration, be sure to apply the most recent patches and updates that are available for that version of the JBoss EAP software. For information about how to upgrade your server configuration, see the *Patching and Upgrading Guide* for JBoss EAP.

2.2. RUN WITH A CLEAN TARGET SERVER INSTALLATION

Because the JBoss Server Migration Tool creates the configuration files based on the configuration of a previous release, it is intended to be run on a clean and unconfigured target server installation. The JBoss Server Migration Tool creates a backup of the target server's configuration files by appending `.beforeMigration` to the file names. It then creates totally new configuration files for the target server using the source server's configuration files, and migrates the configuration to run in target server configuration.

WARNING

When you run the JBoss Server Migration Tool, all changes on the target server made between installation and running the migrate tool are lost.

Also, be aware that if you run the tool against the target server directory more than once, the subsequent runs will overwrite the original target configuration files that were backed up on the first run of the tool. This is because each run of the tool backs up the configuration files by appending `.beforeMigration`, resulting in the loss of any existing backed up configuration files.

2.3. CUSTOMIZE THE MIGRATION

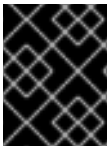
The JBoss Server Migration Tool provides the ability to configure logging, reporting, and the execution of migration tasks. By default, when you run the JBoss Server Migration Tool in non-interactive mode, it migrates the entire server configuration. You can configure the JBoss Server Migration Tool to customize logging and reporting output. You can also configure it to skip any part of the configuration that you do not want to migrate.

For instructions on how to configure properties to control the migration process, see Configuring the JBoss Server Migration Tool.

CHAPTER 3. RUNNING THE JBOSS SERVER MIGRATION TOOL

You can run the JBoss Server Migration Tool in either of the following ways.

- Interactive mode: This mode, which is the default, allows you to choose exactly which configurations you want to migrate.
- Non-interactive mode: This mode allows you to run the tool without prompts.



IMPORTANT

You must stop both the source and the target JBoss EAP servers before you run the JBoss Server Migration Tool.

3.1. RUN THE JBOSS SERVER MIGRATION TOOL IN INTERACTIVE MODE

By default, the JBoss Server Migration Tool runs interactively. This mode allows you to choose exactly which server configurations you want to migrate.

NOTE

Interactive mode does not allow you to choose which subsystems to migrate. For information on how to configure the tool at the subsystem or task level, see Configure the Migration Tasks Performed by the JBoss Server Migration Tool.

The following are the basic steps that are performed for a minimal migration. If the server from which you are migrating includes custom configurations, for example deployments, or if it is missing default resources, the tool provides additional prompts.

1. To run the tool in interactive mode, navigate to the target server installation directory and run the following command, providing the **`source`** argument as the path to the source server installation.

```
$ EAP_HOME/bin/jboss-server-migration.sh --source EAP_PREVIOUS_HOME
```

2. You are prompted to determine if you want to migrate the source server's standalone configurations, which are located in the ***`EAP_PREVIOUS_HOME`*`/standalone/configuration/`** directory, to the target server's standalone configurations, which are located in the ***`EAP_HOME`*`/standalone/configuration/`** directory.

```
Migrate the source's standalone server?
yes/no? yes
```

If you respond with **no**, standalone server migration is skipped and no standalone server configuration files are migrated.

If you respond with **`yes`**, you see the following prompt.

```
Migrate all configurations?
yes/no? yes
```

Respond with **yes** to migrate all of the source server's standalone server configuration files.

Respond with **no** to receive a prompt for each individual **standalone*.xml** configuration file.

3. Next, you are prompted to determine if you want to migrate the source server's managed domain configurations, which are located in the ***EAP_PREVIOUS_HOME*/domain/configuration/** directory, to the target server's managed domain configurations, which are located in the ***EAP_HOME*/domain/configuration/** directory.

```
Migrate the source's managed domain?
yes/no? yes
```

If you respond with **no**, managed domain migration is skipped and no managed domain configuration files are migrated.

If you respond with **yes**, the tool begins migrating the managed domain content of the source server. A ciphered repository is used to store data, such as deployments and deployment overlays, that are referenced by the source server's managed domain and host configurations. Because the source and target servers use a similar content repository, the tool simply copies the data from the source server to the target server and prints the results to the console and the server log.

```
INFO  [ServerMigrationTask#397] Migrating domain content found:
[22/caa450a9ba3b84eaf5a15b6da418b92ce6c98e/content,
23/b62a37ba8a4830622bfcdb960280577cc6796e/content]
INFO  [ServerMigrationTask#398] Resource with path
/__EAP_HOME__/domain/data/content/22/caa450a9ba3b84eaf5a15b6da418b92
ce6c98e/content migrated.
INFO  [ServerMigrationTask#399] Resource with path
/__EAP_HOME__/domain/data/content/23/b62a37ba8a4830622bfcdb960280577
cc6796e/content migrated.
```

4. Next, the migration tool scans the source server for managed domain configuration files, prints the results to the console, and provides the following prompt.

```
Migrate all configurations?
yes/no? yes
```

Respond with **yes** to migrate all of the source server's managed domain configuration files.

Respond with **no** to receive a prompt for each individual managed domain configuration file.

5. Next, the migration tool scans the source server for host configurations files, prints the results to the console, and provides the following prompt.

```
INFO  [ServerMigrationTask#457] Retrieving source's host
configurations...
INFO  [ServerMigrationTask#457] /jboss-eap-
6.4/domain/configuration/host-master.xml
INFO  [ServerMigrationTask#457] /jboss-eap-
```

```
6.4/domain/configuration/host-slave.xml
INFO  [ServerMigrationTask#457] /jboss-eap-
6.4/domain/configuration/host.xml

Migrate all configurations?
yes/no? yes
```

Respond with **yes** to migrate all of the source server's host configuration files.

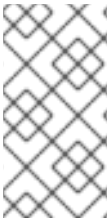
Respond with **no** to receive a prompt for each individual host configuration file.

6. Upon completion, you should see the following message in the server console.

```
Migration Result: SUCCESS
```

3.2. RUN THE JBOSS SERVER MIGRATION TOOL IN NON-INTERACTIVE MODE

You can run the JBoss Server Migration Tool in non-interactive mode. This mode allows it to run without prompts.



NOTE

The JBoss Server Migration Tool automatically migrates all subsystem configurations for all server configuration files. For information on how to configure the tool at the subsystem or task level, see Configure the Migration Tasks Performed by the JBoss Server Migration Tool.

1. To run the tool in non-interactive mode, navigate to the target server installation directory and run the following command, providing the **source** argument as the path to the source server installation and setting the **--interactive** or **-i** argument to **false**.

```
$ EAP_HOME/bin/jboss-server-migration.sh --source EAP_PREVIOUS_HOME
--interactive false
```

2. By default, the tool automatically migrates all of the source server's standalone and managed domain configuration files. However, you can configure the tool's properties to skip migration of specific configurations. Upon completion, you should see the following message in the server console.

```
Migration Result: SUCCESS
```

CHAPTER 4. REVIEWING THE RESULTS OF JBOSS SERVER MIGRATION TOOL EXECUTION

4.1. REVIEW THE MIGRATED CONFIGURATION FILES

When the migration is complete, review the migrated server configuration files in the ***EAP_HOME*/standalone/configuration/** and ***EAP_HOME*/domain/configuration/** directories.

- Note that any original ***EAP_HOME*** target server configuration file names selected for migration are backed up and are now appended with **.beforeMigration**.
- The ***EAP_HOME*** target server configuration file names not appended with **.beforeMigration** are now updated with the content migrated from the ***EAP_PREVIOUS_HOME*** source server configuration.
- The original configuration files located in the ***EAP_PREVIOUS_HOME*** source server configuration directories remain untouched.
- The **logging.properties** and **standalone-load-balancer.xml** files in the target configuration directories remain untouched.

If you choose to migrate all of the available configurations, you should see the following configuration files in the target server directories.

Example: List of Configuration Files on the Target Server

```
$ ls EAP_HOME/standalone/configuration/
application-roles.properties
application-roles.properties.beforeMigration
application-users.properties
application-users.properties.beforeMigration
logging.properties
mgmt-groups.properties
mgmt-groups.properties.beforeMigration
mgmt-users.properties
mgmt-users.properties.beforeMigration
standalone-full-ha.xml
standalone-full-ha.xml.beforeMigration
standalone-full.xml
standalone-full.xml.beforeMigration
standalone-ha.xml
standalone-ha.xml.beforeMigration
standalone-load-balancer.xml
standalone-osgi.xml
standalone-osgi.xml.beforeMigration
standalone.xml
standalone.xml.beforeMigration

$ ls EAP_HOME/domain/configuration/
application-roles.properties
application-roles.properties.beforeMigration
application-users.properties
application-users.properties.beforeMigration
domain.xml
```

```
domain.xml.beforeMigration
host-master.xml
host-master.xml.beforeMigration
host-slave.xml
host-slave.xml.beforeMigration
host.xml
host.xml.beforeMigration
logging.properties
mgmt-groups.properties
mgmt-groups.properties.beforeMigration
mgmt-users.properties
mgmt-users.properties.beforeMigration
```

4.2. TRACKING MIGRATION TASK EXECUTION

The JBoss Server Migration Tool begins each target server migration by executing a root task, which can then execute subtasks. Those subtasks can then also execute additional tasks and subtasks. As it executes, the tool tracks each migration task, along with any substasks, and saves the results in a tree structure that is later used to build the reports.

Each migration task is given a name, which consists of a task name concatenated with optional attributes using the following syntax.

```
TASK_NAME(ATTRIBUTE_1_NAME=ATTRIBUTE_1_VALUE,ATTRIBUTE_2_NAME=ATTRIBUTE_2_
VALUE, ...)
```

The name defines the task subject or type, and the attributes are used to distinguish between subtasks and sibling tasks. For example, all of the following are names to distinguish **`ejb3`** subsystem update tasks.

```
subsystem.ejb3.update
subsystem.ejb3.update.activate-ejb3-remoting-http-
connector(resource=/subsystem=ejb3)
subsystem.ejb3.update.setup-default-sfsb-passivation-disabled-
cache(resource=/subsystem=ejb3)
subsystem.ejb3.update.add-infinispan-passivation-store-and-distributable-
cache(resource=/subsystem=ejb3)
```

Since a migration task can be executed multiple times under different parent tasks, each task is stored in the tree using each of its parent task names, starting with root, separated by a **>** character. The task execution tree is used to build the migration reports.

A task execution can result in one of the following statuses.

Status	Description
Success	The task executed successfully.
Skipped	The task skipped the execution, either because it was not needed or because it was configured to be skipped.
Fail	The task execution failed.

4.3. REVIEW THE TASK SUMMARY LOG

The Task Summary is generated and printed to the migration console and to the JBoss Server Migration Tool log file. It provides a high-level overview of the migration results, by component and subtask, as a hierarchical list.

An example Task Summary report is located in the appendix of this guide.

See Configuring the Task Summary for options to configure the resulting output.

4.4. REVIEW THE JBOSS SERVER MIGRATION TOOL REPORTS

The JBoss Server Migration Tool generates nicely formatted HTML and XML reports in the ***EAP_HOME*/migration/reports/** directory. These reports provide a detailed analysis of the migration process and how the target server was configured during the migration. The default names for these reports are **migration-report.html** and **migration-report.xml**. Each of these names is configurable. For information about how to configure the reports, see Configuring Reporting for JBoss Server Migration Tool.

This section provides a brief overview of the content of these reports.

- The JBoss Server Migration Report HTML file.
- The Server Migration Report XML file.

4.4.1. JBoss Server Migration Tool HTML Report

The HTML report consists of three sections.

Summary

This section provides the execution start time, information about the source and target servers, and the result of the migration.

Environment

This section lists the environment properties that were used for the migration.

Tasks

This section, which includes collapsible subsections, provides statistics and a map of the executed migration tasks. Each task is listed by its name and is color-coded according to the status of the completion of the task:

- Green if it was successful.
- Red if it failed.
- Gray if it was skipped.

An example HTML report is located in the appendix of this guide.

See Configuring the HTML Report for options to configure the resulting output.

4.4.2. JBoss Server Migration Tool XML Report

The XML Report is a low level report that provides all of the migration data gathered by the tool. It is formatted in a way that it can be imported into and manipulated by third-party spreadsheet or other data manipulation tools.

An example XML report is located in the appendix of this guide.

See Configuring the XML Report for options to configure the resulting output.

CHAPTER 5. CONFIGURING THE JBOSS SERVER MIGRATION TOOL

5.1. CONFIGURING THE JBOSS SERVER MIGRATION TOOL PROPERTIES

You use properties to configure the JBoss Server Migration Tool logging and reporting output and to determine which components and configurations you want to migrate. You can configure these properties using a combination of the following methods.

- You can configure the properties file defined within the tool.
- You can pass user properties on the command line.
- You can configure system environment variables.

5.1.1. Configure the JBoss Server Migration Tool Using the Tool Properties

You can configure the JBoss Server Migration Tool using the **`environment.properties`** file located in the ***`EAP_HOME`*`/migration/configuration/`** directory. This standard Java properties file provides the default values for all of the valid properties that can be configured when migrating to the target server. To change a default value, remove the # comment character that precedes the property and set it to the value you need.

5.1.2. Configure the JBoss Server Migration Tool Using User Properties

If you prefer, you can create a standard Java properties file that defines JBoss Server Migration Tool configuration properties and pass its path on the command line using the **`--environment`** or **`-e`** argument. This path can be an absolute path or a path relative to the current directory.

```
$ EAP_HOME/bin/jboss-server-migration.sh --source EAP_PREVIOUS_HOME --
environment path/to/my-server-migration.properties
```

Properties defined in files passed on the command line using the **`--environment`** or **`-e`** argument override the ones defined in the ***`EAP_HOME`*`/migration/configuration/environment.properties`** file.

5.1.3. Configure the JBoss Server Migration Tool Using System Properties

You can configure the JBoss Server Migration Tool by passing system properties on the command line using the following syntax.

```
$ EAP_HOME/bin/jboss-server-migration.sh --source EAP_PREVIOUS_HOME -
Djboss.server.migration.PROPERTY_NAME=PROPERTY_VALUE
```

The system property name should be specified as **`jboss.server.migration.`** concatenated with the environment property name. The following example demonstrates how to specify the name of the XML report as **`migration-report.xml`** when starting the JBoss Server Migration Tool.

```
$ EAP_HOME/bin/jboss-server-migration.sh --source EAP_PREVIOUS_HOME  -
Djboss.server.migration.report.xml.fileName=migration-report.xml
```

Environment properties specified on the command line override both user configuration properties and tool configuration properties.

WARNING

Configuring the JBoss Server Migration Tool by passing system properties on the command line does not currently work for the following properties.

- **`report.html.fileName`**
- **`report.html.maxTaskPathSizeToDisplaySubtasks`**
- **`report.html.templateFileName`**
- **`report.summary.maxTaskPathSizeToDisplaySubtasks`**
- **`report.xml.fileName`**

This is a known issue that should be addressed in the next version of JBoss EAP. For more information about this issue, see JBEAP-12901.

5.2. CONFIGURING LOGGING FOR JBOSS SERVER MIGRATION TOOL

The JBoss Server Migration Tool uses the JBoss Logging framework to log the progress of the migration. Results are written to the console and also to a file named **`migration.log`**, which is located in the ***`EAP_HOME`*`/migration/logs/`** directory. This log file is created if it does not already exist, and its content is overwritten on each subsequent execution of the tool.

The logging configuration is provided by the ***`EAP_HOME`*`/migration/logging.properties`** file. You can modify this configuration file or you can specify an alternative logging configuration file by using the **`logging.configuration`** system property on the command line.

```
$ EAP_HOME/bin/jboss-server-migration.sh --source EAP_PREVIOUS_HOME -
Dlogging.configuration=file:EAP_PREVIOUS_HOME/migration/configuration/my-
alternate-logging.properties
```

5.3. CONFIGURING MODULES MIGRATION

The JBoss Server Migration Tool can migrate any module installed in the source server as long as that module is not already installed on the target server. Module migration can be done explicitly by request, or implicitly because another module or migrated server configuration depends on it.

5.3.1. Modules Environment Properties

You can control whether a module should be migrated or not by using the **`modules.includes`** and **`modules.excludes`** environment properties. The syntax for a module ID is **`name:slot`**. The **`:slot`** is optional and if it is not specified defaults to **`main`**.

- A module whose ID is referenced by the **`modules.excludes`** environment property is never migrated.
- A module whose ID is referenced by the **`modules.includes`** environment property is always migrated, unless it is referenced by the **`modules.excludes`** environment property.

5.3.2. Configuring Modules Properties

The environment properties used to migrate modules can be configured in any of the following ways:

- You can configure the properties in the tool's ***`EAP_HOME`*`/migration/configuration/environment.properties`** file.

```
modules.includes=com.example.moduleA,com.example.moduleB
modules.excludes=com.example.moduleC
```

- You can include the above properties in your own custom properties file, and then pass the properties file name on the command line using the **`--environment`** argument.

```
$ EAP_HOME/bin/jboss-server-migration.sh --source EAP_PREVIOUS_HOME
--environment PATH_TO_MY_PROPERTIES_FILE
```

- You can pass the information on the command line using a system property. The environment property names must be prefixed with **`jboss.server.migration.`**, for example:

```
$ EAP_HOME/bin/jboss-server-migration.sh --source EAP_PREVIOUS_HOME
-Djboss.server.migration.modules.includes="com.example.moduleA" -
Djboss.server.migration.modules.excludes="com.example.moduleC,com.ex
ample.moduleD"
```

WARNING

The JBoss Server Migration Tool does not verify that the source module is compatible with the target server. An incompatible migrated module can cause the target server to malfunction or not work at all. A module can be incompatible due to a dependency on a module that is installed on both the source and target servers, but includes or exposes different resources on each one.

5.4. CONFIGURING REPORTING FOR JBOSS SERVER MIGRATION TOOL

5.4.1. Configuring the Task Summary Log

You can customize the generation of the Task Summary using the following environment property.

Property Name	Type	Property Description and Default Value
report.summary.maxTaskPathSizeToDisplaySubtasks	Integer	Include migrated subtasks in the summary where the level is less than or equal to the specified integer. Defaults to **5**.

5.4.2. Configuring the HTML Report

You can customize the HTML report using the following environment properties.

Property Name	Type	Property Description and Default Value
report.html.fileName	String	The name of the HTML report file. If not set, the report is not generated. Defaults to ***EAP_HOME*/migration/reports/migration-report.html**.
report.html.maxTaskPathSizeToDisplaySubtasks	Integer	Include migrated subtasks in the summary where the level is less than or equal to the specified integer. Defaults to **4**.
report.html.templateFileName	String	The HTML report template file name. Defaults to **migration-report-template.html**.

5.4.3. Configuring the XML Report

You can customize the XML report using the following environment properties.

Property Name	Type	Property Description and Default Value
report.xml.fileName	String	The name of the XML report file. If not set, the report is not generated. Defaults to ***EAP_HOME*/migration/reports/migration-report.xml**.

5.5. CONFIGURING THE MIGRATION OF THE STANDALONE SERVER CONFIGURATION

You can configure the JBoss Server Migration Tool to skip the migration of a standalone server entirely, to provide the configuration file names that you want to migrate, or to provide alternate paths for the source or target server's base and configuration directories.

You can customize the migration of the standalone server configuration using the following environment properties.

Table 5.1. Standalone Server Migration Environment Properties

Property Name	Property Description
standalone.skip	If set to **`true`**, the tool skips the entire standalone server migration.
server.source.standalone.serverDir	Defines an alternative path for the source server's standalone directory, which defaults to the source server's ***`EAP_HOME`*`/standalone/`** directory.
server.source.standalone.configDir	Defines an alternative path for the source server's standalone configuration directory, which defaults to the source server's ***`EAP_HOME`*`/standalone/configuration/`** directory.
server.source.standalone.configFiles	A comma-delimited list of the source server's standalone configurations to be migrated.
server.target.standalone.serverDir	Defines an alternative path for the target server's standalone directory, which defaults to the target server's ***`EAP_HOME`*`/standalone/`** directory.
server.target.standalone.configDir	Defines an alternative path for the target server's standalone configuration directory, which defaults to the target server's ***`EAP_HOME`*`/standalone/configuration/`** directory.

For information about how to configure the JBoss Server Migration Tool using these properties, see Configuring the JBoss Server Migration Tool.

5.6. CONFIGURING THE MIGRATION OF A MANAGED DOMAIN CONFIGURATION

You can configure the JBoss Server Migration Tool to skip the migration of a managed domain entirely, to provide the configuration file names that you want to migrate, or to provide alternate paths for the source or target server's base and configuration directories.

You can customize the migration of the managed domain configuration using the following environment properties.

Table 5.2. Managed Domain Migration Environment Properties

Property Name	Property Description
domain.skip	If set to **`true`**, the tool skips the entire managed domain migration.
server.source.domain.domainDir	Defines an alternative path for the source server's managed domain directory, which defaults to the source server's ***`EAP_HOME`*`/domain/`** directory.
server.source.domain.configDir	Defines an alternative path for the source server's managed domain configuration directory, which defaults to the source server's ***`EAP_HOME`*`/domain/configuration/`** directory.
server.source.domain.domainConfigFiles	A comma-delimited list of the source server's managed domain configuration files that are to be migrated.
server.source.domain.hostConfigFiles	A comma-delimited list of the source server's host configuration files that are to be migrated.
server.target.domain.domainDir	Defines an alternative path for the target server's managed domain directory, which defaults to the target server's ***`EAP_HOME`*`/domain/`** directory.
server.target.domain.configDir	Defines an alternative path for the target server's managed domain configuration directory, which defaults to the target server's ***`EAP_HOME`*`/domain/configuration/`** directory.

For information about how to configure the JBoss Server Migration Tool using these properties, see Configuring the JBoss Server Migration Tool.

5.7. CONFIGURE THE MIGRATION TASKS PERFORMED BY THE JBOSS SERVER MIGRATION TOOL

By default, the JBoss Server Migration Tool automatically migrates all components and subsystems for each standalone server, managed domain, and host configuration you choose to migrate. You can customize the execution of specific tasks and subtasks performed by the tool using environment properties. For example, you can configure the tool to skip the removal of unsupported subsystems or to skip the migration of deployments. The tasks performed by the tool are dependent upon the type of server configuration and the version of the source server from which you are migrating.

Information about how to configure environment properties to customize the tasks performed by the JBoss Server Migration Tool can be found in the following sections.

CHAPTER 6. MIGRATING JBOSS EAP 6.4 CONFIGURATIONS TO JBOSS EAP 7.1

6.1. MIGRATING A JBOSS EAP 6.4 STANDALONE SERVER TO JBOSS EAP 7.1

By default, the JBoss Server Migration Tool performs the following tasks when migrating a standalone server configuration from JBoss EAP 6.4 to JBoss EAP 7.1.

- Remove any unsupported subsystems.
- Migrate any referenced modules.
- Migrate any referenced paths.
- Update the **`transactions`** subsystem.
- Migrate the **`jacorb`** subsystem.
- Migrate the **`web`** subsystem.
- Migrate the **`messaging`** subsystem.
- Update the **`infinispan`** subsystem.
- Update the **`ee`** subsystem.
- Update the **`ejb3`** subsystem.
- Update the **`jgroups`** subsystem.
- Update the **`remoting`** subsystem.
- Update the **`undertow`** subsystem.
- Update the **`messaging-activemq`** subsystem.
- Add the **`batch-jberet`** subsystem.
- Add the **`core-management`** subsystem.
- Add the **`elytron`** subsystem.
- Add the **`request-controller`** subsystem.
- Add the **`security-manager`** subsystem.
- Add the **`singleton`** subsystem.
- Set up HTTP Upgrade management.
- Set up the private interface.
- Add socket binding port expressions.

- Migrate compatible security realms.
- Add the default SSL server identity to the **`ApplicationRealm`**.
- Migrate deployments.

6.1.1. Remove Unsupported Subsystems

The following JBoss EAP 6.4 subsystems are not supported by JBoss EAP 7:

Subsystem Name	Configuration Namespace	Extension Module
cmp	urn:jboss:domain:cmp:*	org.jboss.as.cmp
configadmin	urn:jboss:domain:configadmin:*	org.jboss.as.configadmin
jaxr	urn:jboss:domain:jaxr:*	org.jboss.as.jaxr
osgi	urn:jboss:domain:osgi:*	org.jboss.as.osgi
threads	urn:jboss:domain:threads:*	org.jboss.as.threads

The JBoss Server Migration Tool removes all unsupported subsystem configurations and extensions from migrated server configurations. The tool logs each subsystem and extension to its log file and to the console as it is removed.

To skip removal of the unsupported subsystems, set the **`subsystems.remove-unsupported-subsystems.skip`** environment property to **`true`**.

6.1.2. Migrate Referenced Modules

A configuration that is migrated from a source server to a target server might reference or depend on a module that is not installed on the target server. The JBoss Server Migration Tool detects this and automatically migrates the referenced modules, plus their dependent modules, from the source server to the target server.

A module referenced by a standalone server configuration is migrated using the following process.

- A module referenced by a security realm configuration is migrated as a plug-in module.
- A module referenced by the **`datasource`** subsystem configuration is migrated as a datasource driver module.
- A module referenced by the **`ee`** subsystem configuration is migrated as a global module.
- A module referenced by the **`naming`** subsystem configuration is migrated as an object factory module.
- A module referenced by the **`messaging`** subsystem configuration is migrated as a JMS bridge module.
- A module referenced by a vault configuration is migrated to the new configuration.

The console logs a message noting the module ID for any module that is migrated. It is possible to exclude the migration of specific modules by specifying the module ID in the **`modules.excludes`** environment property. See Configuring the Migration of Modules for more information.

6.1.3. Migrate Referenced Paths

A configuration that is migrated from a source server to a target server might reference or depend on file paths and directories that must also be migrated to the target server. The JBoss Server Migration Tool does not migrate absolute path references. It only migrates files or directories that are configured as relative to the source configuration. The console logs a message noting each path that is migrated.

The JBoss Server Migration Tool automatically migrates the following path references:

- Vault keystore and encrypted file's directory.

To skip the migration of referenced paths, set the **`paths.migrate-paths-requested-by-configuration.vault.skip`** environment property to **`true`**.

6.1.4. Update the Transactions Subsystem

The JBoss Server Migration Tool updates the **`transactions`** subsystem with the configuration changes required by the JBoss EAP 7.1 server.

The JBoss Server Migration Tool removes the **`path`** and **`relative-to`** attributes from the **`transactions`** subsystem and replaces them with the equivalent **`object-store-path`** and **`object-store-relative-to`** attributes.

To skip the automatic update of the **`transactions`** subsystem configuration, set the **`subsystem.transactions.update-xml-object-store-paths.skip`** environment property to **`true`**.

6.1.5. Migrate the Jacorb Subsystem

The **`jacorb`** subsystem is deprecated in JBoss EAP 7 and is replaced by the **`iiop-openjdk`** subsystem. By default, the JBoss Server Migration Tool automatically migrates the **`jacorb`** subsystem configuration to its replacement **`iiop-openjdk`** subsystem configuration and logs the results to its log file and to the console.

To skip the automatic migration to the **`iiop-openjdk`** subsystem configuration, set the **`subsystem.jacorb.migrate.skip`** environment property value to **`true`**.

6.1.6. Migrate the Web Subsystem

The **`web`** subsystem is deprecated in JBoss EAP 7 and is replaced by the **`undertow`** subsystem. By default, the JBoss Server Migration Tool automatically migrates the **`web`** subsystem configuration to its replacement **`undertow`** subsystem configuration and logs the results to its log file and to the console.

To skip automatic migration of the **`web`** subsystem, set the **`subsystem.web.migrate.skip`** environment property value to **`true`**.

6.1.7. Migrate the Messaging Subsystem

The **`messaging`** subsystem is deprecated in JBoss EAP 7 and is replaced by the **`messaging-`**

activemq subsystem. The JBoss Server Migration Tool automatically migrates the **messaging** subsystem configuration to its replacement **messaging-activemq** subsystem configuration and logs the results to its log file and to the console.

To skip automatic migration of the **messaging** subsystem, set the **subsystem.messaging.migrate.skip** environment property value to **true**.

6.1.8. Update the Infinispan Subsystem

The JBoss Server Migration Tool updates the **infinispan** subsystem configuration to better align with the new default JBoss EAP 7.1 configurations.

- It adds the EJB cache container, which is present in the JBoss EAP 7.1 default configuration, to configurations where it is not already included.
- It adds the server cache container, which is present in the JBoss EAP 7.1 default configuration.
- It updates the module name in the Hibernate cache container configuration.

The JBoss Server Migration Tool automatically updates the **infinispan** subsystem configuration and logs the results to its log file and to the console.

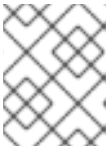
You can customize the update of the **infinispan** system by setting the following environment properties.

Property Name	Property Description
subsystem.infinispan.update.skip	If set to **true**, skip the update of the **infinispan** subsystem.
subsystem.infinispan.update.add-infinispan-ejb-cache.skip	If set to **true**, do not add the EJB cache container.
subsystem.infinispan.update.add-infinispan-server-cache.skip	If set to **true**, do not add the server cache container.
subsystem.infinispan.update.fix-hibernate-cache-module-name.skip	If set to **true**, do not update the module name in the Hibernate cache container configuration.

6.1.9. Update the EE Subsystem

The JBoss Server Migration Tool updates the **ee** subsystem to configure the new Java EE 7 features supported in JBoss EAP 7.1.

- It configures instances of Java EE 7 concurrency utilities, such as container-managed executors, that are present in the JBoss EAP 7.1 default configuration and logs the results to its log file and to the console.
- It defines the Java EE 7 default resources, such as the default datasource, that are present in default JBoss EAP 6.4 configuration. If the resources are not found, the tool lists all available resources in the configuration, and then provides a prompt to select a resource from the list or to provide the JNDI address of the resource that should be set as the default.



NOTE

JNDI names that are specified are assumed to be valid. JNDI names are not validated by the tool.

The JBoss Server Migration Tool automatically updates the **ee** subsystem configuration and logs the results to its log file and to the console.

You can customize the update of the **ee** system by setting the following environment properties.

Property Name	Property Description
subsystem.ee.update.skip	If set to **true**, skip the update of the **ee** subsystem.
subsystem.ee.update.setup-ee-concurrency-utilities.skip	If set to **true**, do not add the default instances of concurrency utilities.
subsystem.ee.update.setup-javaee7-default-bindings.skip	If set to **true**, do not set up Java EE 7 default resources.
subsystem.ee.update.setup-javaee7-default-bindings.defaultDataSourceName	Specifies the name of the default datasource to look for in the source configuration.
subsystem.ee.update.setup-javaee7-default-bindings.defaultDataSourceJndiName	Specifies the JNDI name for the default datasource.
subsystem.ee.update.setup-javaee7-default-bindings.defaultJmsConnectionFactoryName	Specifies the name of the default JMS connection factory.
subsystem.ee.update.setup-javaee7-default-bindings.defaultJmsConnectionFactoryJndiName	Specifies the JNDI name for the default JMS connection factory.

Configuring Concurrency Utilities in the EE Subsystem
If you choose to configure the Java EE 7 concurrency utilities, then the tool automatically configures the instances that are present in the default JBoss EAP 7.1 configurations and logs the results to its log file and to the console.

```
INFO  [ServerMigrationTask#49] Default ContextService added to EE
subsystem configuration.
INFO  [ServerMigrationTask#49] Default ManagedThreadFactory added to EE
subsystem configuration.
INFO  [ServerMigrationTask#49] Default ManagedExecutorService added to EE
subsystem configuration.
INFO  [ServerMigrationTask#49] Default ManagedScheduledExecutorService
added to EE subsystem configuration.
```

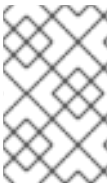
Configuring Default Resources in the EE Subsystem
When defining the Java EE 7 default resources the tool automatically selects those that are present in the default JBoss EAP 7.1 configuration.

```
INFO  [ServerMigrationTask#50] Java EE Default Datasource configured with
JNDI name java:jboss/datasources/ExampleDS.
```

If no default resource is found, the tool lists all resources that are available in the configuration, and then provides a prompt to select the default resource or to provide the JNDI address of the resource that should be set as the default.

The following is an example of the interaction that occurs when migrating a configuration file with an **`ExampleDS`** datasource.

```
INFO  [ServerMigrationTask#22] Default datasource not found.
0. ExampleDS
1. Unconfigured data source, I want to enter the JNDI name...
Please select Java EE's Default Datasource: (0): 0
INFO  [ServerMigrationTask#22] Java EE Default Datasource configured with
JNDI name java:jboss/datasources/ExampleDS.

Save this Java EE Default Datasource JNDI name and use it when migrating
other config files?
yes/no? y
```



NOTE

If you run the JBoss Server Migration Tool in non-interactive mode and the expected JBoss EAP 6.4 default resources, such as the default JMS connection factory, are not available, the tool does not configure those resources.

6.1.10. Update the EJB 3 Subsystem

The JBoss Server Migration Tool makes the following updates to the **`ejb3`** subsystem to better align with the new default JBoss EAP 7.1 configurations.

- It updates the remote service configuration to reference the new HTTP connector.
- It configures the **`default-sfsb-passivation-disabled-cache`** attribute to use the **`default-sfsb-cache`**.
- It replaces the legacy passivation store and cache configurations with the JBoss EAP 7.1 default values.

The JBoss Server Migration Tool automatically updates the **`ejb3`** subsystem configuration and logs the results to its log file and to the console.

You can customize the update of the **`ejb3`** system by setting the following environment properties.

Property Name	Property Description
subsystem.ejb3.update.skip	If set to **`true`**, skip the update of the **`ejb3`** subsystem.
subsystem.ejb3.update.add-infinispan-passivation-store-and-distributable-cache.skip	If set to **`true`**, do not replace the **`passivation-store`** and **`cache`** configurations.

Property Name	Property Description
subsystem.ejb3.update.setup-default-sfsb-passivation-disabled-cache.skip	If set to **true**, do not update the **default-sfsb-passivation-disabled-cache** configuration.
subsystem.ejb3.update.activate-ejb3-remoting-http-connector.skip	If set to **true**, do not update the EJB remoting configuration.

6.1.11. Update the JGroups Subsystem

The JBoss Server Migration Tool updates the **jgroups** subsystem to align with the new JBoss EAP 7.1 configurations.

- It replaces the **MERGE2** protocol with **MERGE3**.
- It replaces the **FD** protocol with **FD_ALL**.
- It replaces the **pbcast.NAKACK** protocol with **pbcast.NAKACK2**.
- It replaces the **UNICAST2** protocol with **UNICAST3**.
- It removes the **RSVP** protocol.

To skip the automatic migration of the **jgroups** subsystem, set the **subsystem.jgroups.update.skip** environment property to **true**.

6.1.12. Update the Remoting Subsystem

JBoss EAP 7 introduces a new HTTP connector that replaces all legacy remoting protocols and ports using a single port. The JBoss Server Migration Tool automatically updates the **remoting** subsystem to use the new HTTP connector.

To skip the automatic update of the **remoting** subsystem configuration, set the **subsystem.remoting.update.skip** environment property to **true**.

6.1.13. Update the Undertow Subsystem

In addition to migrating the **web** subsystem for JBoss EAP 7.1, the JBoss Server Migration Tool updates its replacement **undertow** subsystem to add the new features it supports.

- It sets the default HTTP listener redirect socket.
- It adds support for Java EE 7 WebSockets.
- It sets the default HTTPS listener.
- It adds support for HTTP2.
- It sets the default **Server** response header.
- It sets the default **X-Powered-By** response header.
- It sets the default HTTP **Invoker**.

The JBoss Server Migration Tool automatically updates the **undertow** subsystem configuration and logs the results to its log file and to the console.

You can customize the update of the **undertow** system by setting the following environment properties.

Property Name	Property Description
subsystem.undertow.update.skip	If set to **true**, skip the update of the **undertow** subsystem.
subsystem.undertow.update.set-default-http-listener-redirect-socket.skip	If set to **true**, do not set the default HTTP listener redirect socket.
subsystem.undertow.update.add-undertow-websockets.skip	If set to **true**, do not add support for WebSockets.
subsystem.undertow.update.add-undertow-https-listener.skip	If set to **true**, do not set the default HTTPS listener.
subsystem.undertow.update.enable-http2.skip	If set to **true**, do not add support for HTTP2.
subsystem.undertow.update.add-response-header.server-header.skip	If set to **true**, do not set the default **Server** response header.
subsystem.undertow.update.add-response-header.x-powered-by-header.skip	If set to **true**, do not set the default **X-Powered-By** response header.
subsystem.undertow.update.add-http-invoker.skip	If set to **true**, do not set the default HTTP **Invoker**.

6.1.14. Update the Messaging-ActiveMQ Subsystem

In addition to migrating the **messaging** subsystem for JBoss EAP 7.1, the JBoss Server Migration Tool updates its replacement **messaging-activemq** subsystem to add the new features it supports.

- It adds the default HTTP connector and acceptor to enable the new HTTP-based remote messaging clients.

The JBoss Server Migration Tool automatically updates the **messaging-activemq** subsystem configuration and logs the results to its log file and to the console.

To skip the automatic update of the **messaging-activemq** subsystem, set the **subsystem.messaging-activemq.update.skip** environment property to **true**.

6.1.15. Add the Batch JBeret Subsystem

The JBoss EAP 7.1 **batch-jberet** subsystem provides support for JSR 352: Batch Applications for the Java Platform. The JBoss Server Migration Tool automatically adds the default **batch-jberet** subsystem configuration to the migrated configuration file.

To skip the addition of the **batch-jberet** subsystem configuration, set the **subsystem.batch-jberet.add.skip** environment property to **true**.

6.1.16. Add the Core Management Subsystem

The JBoss EAP 7.1 **core-management** subsystem provides management-related resources, which were previously configured in the management core service. Examples of these resources include the ability to view a history of configuration changes made to the server and the ability to monitor for server lifecycle events. The JBoss Server Migration Tool automatically adds the default **core-management** subsystem configuration to the migrated configuration file.

To skip the addition of the **core-management** subsystem configuration, set the **subsystem.core-management.add.skip** environment property to **true**.

6.1.17. Add the Elytron Subsystem

The JBoss EAP 7.1 **elytron** subsystem provides a single unified security framework that can manage and configure access for both standalone servers and managed domains. It can also be used to configure security access for applications deployed to JBoss EAP servers. The JBoss Server Migration Tool automatically adds the default **elytron** subsystem configuration to the migrated configuration file.

To skip the addition of the **elytron** subsystem configuration, set the **subsystem.elytron.add.skip** environment property to **true**.

6.1.18. Add the Request Controller Subsystem

The JBoss EAP 7.1 **request-controller** subsystem provides congestion control and graceful shutdown functionality. The JBoss Server Migration Tool automatically adds the default **request-controller** subsystem configuration to the migrated configuration file.

To skip the addition of the **request-controller** subsystem configuration, set the **subsystem.request-controller.add.skip** environment property to **true**.

6.1.19. Add the Security Manager Subsystem

The JBoss EAP 7.1 **security-manager** subsystem provides support for Java EE 7 security permissions. The JBoss Server Migration Tool automatically adds the default **security-manager** subsystem configuration to the migrated configuration file.

To skip the addition of the **security-manager** subsystem configuration, set the **subsystem.security-manager.add.skip** environment property to **true**.

6.1.20. Add the Singleton Subsystem

The JBoss EAP 7.1 **singleton** subsystem provides singleton functionality. The JBoss Server Migration Tool automatically adds the default **singleton** subsystem configuration to the migrated configuration file.

To skip the addition of the **singleton** subsystem configuration, set the **subsystem.singleton.add.skip** environment property to **true**.

6.1.21. Set Up HTTP Upgrade Management

The addition of Undertow in JBoss EAP 7 added HTTP Upgrade, which allows for multiple protocols to be multiplexed over a single port. This means a management client can make an initial connection over HTTP, but then send a request to upgrade that connection to another protocol. The JBoss Server Migration Tool automatically updates the configuration to support HTTP Upgrade management.

To skip configuration of HTTP Upgrade management, set the **`management.setup-http-upgrade.skip`** environment property to **`true`**.

6.1.22. Set Up the Private Interface

The JBoss EAP 7 default configuration uses a new private interface on all **`jgroups`** socket bindings. The JBoss Server Migration Tool automatically updates the migrated **`jgroups`** socket bindings to use same configuration.

To skip the configuration of the private interface, set the **`interface.private.setup.skip environment`** property to **`true`**.

6.1.23. Add Socket Binding Port Expressions

The JBoss EAP 7.1 default configurations use value expressions for the port attribute of the following socket bindings:

- **`ajp`**
- **`http`**
- **`https`**

The JBoss Server Migration Tool automatically adds these value expressions to the migrated server configurations.

To skip update of the socket binding port expressions, set the **`socket-bindings.add-port-expressions.skip`** environment property to **`true`**.

6.1.24. Migrate Compatible Security Realms

Because the JBoss EAP 7.1 security realm configurations are fully compatible with the JBoss EAP 6.4 security realm configurations, they require no update by the JBoss Server Migration Tool. However, if the **`application-users.properties`**, **`application-roles.properties`**, **`mgmt-users.properties`**, and **`mgmt-groups.properties`** files are not referenced using an absolute path, the tool copies them to the path expected by the migrated configuration file.

To skip the security realms migration, set the **`security-realms.migrate-properties.skip environment`** property to **`true`**.

6.1.25. Add the Default SSL Server Identity to the ApplicationRealm

The JBoss EAP 7.1 default configuration includes an SSL server identity for the default **`ApplicationRealm`** security realm. The JBoss Server Migration Tool automatically adds this identity to the migrated configuration files.

To skip the addition of this identity, set the **`security-realm.ApplicationRealm.add-ssl-server-identity.skip`** environment property to **`true`**.

6.1.26. Migrate Deployments

The JBoss Server Migration Tool can migrate the following types of standalone server deployment configurations.

- Deployments it references, also known as persistent deployments.
- Deployments found in directories monitored by its deployment scanners.
- Deployment overlays it references.

The migration of a deployment consists of installing related file resources on the target server, and possibly updating the migrated configuration.

The JBoss Server Migration Tool is preconfigured to skip deployments by default when running in non-interactive mode. To enable migration of deployments, set the **`deployments.migrate-deployments.skip`** environment property to **`false`**.

IMPORTANT

Be aware that when you run the JBoss Server Migration Tool in interactive mode and enter invalid input, the resulting behavior depends on the value of the **`deployments.migrate-deployments`** environment property.

- If **`deployments.migrate-deployments.skip`** is set to **`false`** and you enter invalid input, the tool will try to migrate the deployments.
- If **`deployments.migrate-deployments.skip`** is set to **`true`** and you enter invalid input, the tool will skip the deployments migration.

To enable the migration of specific types of deployments, see the following sections.

WARNING

The JBoss Server Migration Tool does not determine whether deployed resources are compatible with the target server. This means that applications or resources might not deploy, might not work as expected, or might not work at all. Also be aware that artifacts such as JBoss EAP 6.4 **`*-jms.xml`** configuration files are copied without modification and can cause the JBoss EAP server to boot with errors.

Red Hat recommends that you use the Red Hat Application Migration Toolkit to analyze deployments to determine compatibility among different JBoss EAP servers. For more information, see the Red Hat Application Migration Toolkit product documentation.

6.1.26.1. Migrate Persistent Deployments

To enable migration of persistent deployments when running in non-interactive mode, set the **`deployments.migrate-persistent-deployments.skip`** environment property to **`false`**.

The JBoss Server Migration Tool searches for any persistent deployment references and lists them to the console.

```
INFO  [ServerMigrationTask#67] Persistent deployments found: [cmtool-
helloworld3.war, cmtool-helloworld4.war, cmtool-helloworld2.war, cmtool-
helloworld1.war]
```

The processing workflow then depends on whether you are running the tool in interactive mode or in non-interactive mode, as described below.

Migrating Persistent Deployments in Non-interactive Mode

If you run the tool in non-interactive mode, the tool uses the preconfigured properties to determine whether to migrate the persistent deployments. Persistent deployments are migrated only if both the **`deployments.migrate-deployments.skip`** and **`deployments.migrate-persistent-deployments.skip`** properties are set to **`false`**.

Migrating Persistent Deployments in Interactive Mode

If you run the tool in interactive mode, the JBoss Server Migration Tool prompts you for each deployment using the following workflow.

1. After printing the persistent deployments it finds to the console, you see the following prompt.

   ```
   This tool is not able to assert if persistent deployments found are
   compatible with the target server, skip persistent deployments
   migration?
   yes/no?
   ```

   - Respond with **yes** to skip migration of persistent deployments. All deployment references are removed from the migrated configuration and you end this part of the migration process.
   - Respond with **no** to continue with the migration.

2. If you choose to continue, you see the following prompt.

   ```
   Migrate all persistent deployments found?
   yes/no?
   ```

   - Respond with **yes** to automatically migrate all deployments and end this part of the migration process.
   - Respond with **no** to continue with the migration.

3. If you choose to continue, you receive a prompt asking to confirm the migration for each referenced deployment.

   ```
   Migrate persistent deployment 'helloworld01.war'?
   yes/no?
   ```

   - Respond with **yes** to migrate the deployment.
   - Respond with **no** to remove the deployment from the migrated configuration.

     ```
     INFO  [ServerMigrationTask#68] Removed persistent deployment from
     configuration /deployment=helloworld01.war
     ```

6.1.26.2. Migrate Deployment Scanner Deployments

Deployment scanners, which are only used in standalone server configurations, monitor a directory for new files and manage their deployment automatically or through special deployment marker files.

To enable migration of deployments that are located in directories watched by a deployment scanner when running in non-interactive mode, set the **`deployments.migrate-deployment-scanner-deployments.skip`** environment property to **`false`**.

When migrating a standalone server configuration, the JBoss Server Migration Tool first searches for any configured deployment scanners. For each scanner found, it searches its monitored directories for deployments marked as deployed and prints the results to the console.

The processing workflow then depends on whether you are running the tool in interactive mode or in non-interactive mode, as described below.

Migrating Deployment Scanner Deployments in Non-interactive Mode

If you run the tool in non-interactive mode, the tool uses the preconfigured properties to determine whether to migrate the deployment scanner deployments. Deployment scanner deployments are migrated only if both the **`deployments.migrate-deployments.skip`** and **`deployments.migrate-deployment-scanner-deployments.skip`** properties are set to **`false`**.

Migrating Deployment Scanner Deployments in Interactive Mode

If you run the tool in interactive mode, the JBoss Server Migration Tool prompts you for each deployment using the following workflow.

1. After printing the deployment scanner deployments it finds to the console, you see the following prompt.

```
This tool is not able to assert if the scanner's deployments found
are compatible with the target server, skip scanner's deployments
migration?
yes/no?
```

   - Respond with **yes** to skip migration of deployment scanner deployments. All deployment references are removed from the migrated configuration and you end this part of the migration process.
   - Respond with **no** to continue with the migration.

2. If you choose to continue, you see the following prompt.

```
Migrate all scanner's deployments found?
yes/no?
```

   - Respond with **yes** to automatically migrate all deployments and end this part of the migration process.
   - Respond with **no** to continue with the migration.

3. If you choose to continue, you receive a prompt asking to confirm the migration for each referenced deployment.

```
Migrate scanner's deployment 'helloworld02.war'?
yes/no?
```

- Respond with **yes** to migrate the deployment.
- Respond with **no** to remove the deployment from the migrated configuration.

```
INFO  [ServerMigrationTask#69] Resource with path
EAP_HOME/standalone/deployments/helloworld02.war migrated.
```

6.1.26.3. Migrate Deployment Overlays

The migration of deployment overlays is a fully automated process. If you have enabled migration of deployments by setting the **`deployments.migrate-deployments.skip`** environment property to **`false`**, the JBoss Server Migration Tool searches for deployment overlays referenced in the standalone server configuration that are linked to migrated deployments. It automatically migrates those that are found, removes those that are not referenced, and logs the results to its log file and to the console.

6.2. MIGRATING A JBOSS EAP 6.4 MANAGED DOMAIN TO JBOSS EAP 7.1

WARNING

When you use the JBoss Server Migration Tool to migrate a domain controller to a new release of JBoss EAP, it can cause the hosts that connect to it, but still run on a previous release of the server, to fail to boot. For this reason, you should migrate your hosts before you migrate your domain controller. If you do not want to migrate your hosts, be sure to read the following information before you begin.

Review Configure a JBoss EAP 7.x Domain Controller to Administer JBoss EAP 6 Instances in the *Configuration Guide* for JBoss EAP. Pay particular attention to the section entitled Prevent the JBoss EAP 6 Instances from Receiving JBoss EAP 7 Updates.

For additional information, see Managing Multiple JBoss EAP Versions in the *Configuration Guide* for JBoss EAP.

By default, the JBoss Server Migration Tool performs the following tasks when migrating a managed domain configuration from JBoss EAP 6.4 to JBoss EAP 7.1.

- Remove any unsupported subsystems.
- Migrate any referenced modules.
- Migrate any referenced paths.
- Update the **`transactions`** subsystem.
- Migrate the **`jacorb`** subsystem.

- Migrate the **`web`** subsystem.
- Migrate the **`messaging`** subsystem.
- Update the **`infinispan`** subsystem.
- Update the **`ee`** subsystem.
- Update the **`ejb3`** subsystem.
- Update the **`jgroups`** subsystem.
- Update the **`remoting`** subsystem.
- Update the **`undertow`** subsystem.
- Update the **`messaging-activemq`** subsystem.
- Add the **`batch-jberet`** subsystem.
- Add the **`core-management`** subsystem.
- Add the **`elytron`** subsystem.
- Add the **`request-controller`** subsystem.
- Add the **`security-manager`** subsystem.
- Add the **`singleton`** subsystem.
- Update the **`unsecure`** interface.
- Set up the private interface.
- Add socket binding port expressions.
- Add socket binding multicast address expressions.
- Add the load balancer profile.
- Add the host excludes configuration.
- Remove the **`PermGen`** attributes from the JVM configurations.
- Migrate deployments.

6.2.1. Remove Unsupported Subsystems

The following JBoss EAP 6.4 subsystems are not supported by JBoss EAP 7:

Subsystem Name	Configuration Namespace	Extension Module
cmp	urn:jboss:domain:cmp:*	org.jboss.as.cmp

Subsystem Name	Configuration Namespace	Extension Module
configadmin	urn:jboss:domain:configadmin:*	org.jboss.as.configadmin
jaxr	urn:jboss:domain:jaxr:*	org.jboss.as.jaxr
osgi	urn:jboss:domain:osgi:*	org.jboss.as.osgi
threads	urn:jboss:domain:threads:*	org.jboss.as.threads

The JBoss Server Migration Tool removes all unsupported subsystem configurations and extensions from migrated server configurations. The tool logs each subsystem and extension to its log file and to the console as it is removed.

To skip removal of the unsupported subsystems, set the **`subsystems.remove-unsupported-subsystems.skip`** environment property to **`true`**.

6.2.2. Migrate Referenced Modules

A configuration that is migrated from a source server to a target server might reference or depend on a module that is not installed on the target server. The JBoss Server Migration Tool detects this and automatically migrates the referenced modules, plus their dependent modules, from the source server to the target server.

A module referenced by a managed domain configuration is migrated using the following process.

- A module referenced by a security realm configuration is migrated as a plug-in module.
- A module referenced by the **`datasource`** subsystem configuration is migrated as a datasource driver module.
- A module referenced by the **`ee`** subsystem configuration is migrated as a global module.
- A module referenced by the **`naming`** subsystem configuration is migrated as an object factory module.
- A module referenced by the **`messaging`** subsystem configuration is migrated as a JMS bridge module.
- A module referenced by a vault configuration is migrated to the new configuration.

The console logs a message noting the module ID for any module that is migrated. It is possible to exclude the migration of specific modules by specifying the module ID in the **`modules.excludes`** environment property. See Configuring the Migration of Modules for more information.

6.2.3. Migrate Referenced Paths

A configuration that is migrated from a source server to a target server might reference or depend on file paths and directories that must also be migrated to the target server. The JBoss Server Migration Tool does not migrate absolute path references. It only migrates files or directories that are configured as relative to the source configuration. The console logs a message noting each path that is migrated.

The JBoss Server Migration Tool automatically migrates the following path references:

- Vault keystore and encrypted file's directory.

To skip the migration of referenced paths, set the **`paths.migrate-paths-requested-by-configuration.vault.skip`** environment property to **`true`**.

6.2.4. Update the Transactions Subsystem

The JBoss Server Migration Tool updates the **`transactions`** subsystem with the configuration changes required by the JBoss EAP 7.1 server.

The JBoss Server Migration Tool removes the **`path`** and **`relative-to`** attributes from the **`transactions`** subsystem and replaces them with the equivalent **`object-store-path`** and **`object-store-relative-to`** attributes.

To skip the automatic update of the **`transactions`** subsystem configuration, set the **`subsystem.transactions.update-xml-object-store-paths.skip`** environment property to **`true`**.

6.2.5. Migrate the Jacorb Subsystem

The **`jacorb`** subsystem is deprecated in JBoss EAP 7 and is replaced by the **`iiop-openjdk`** subsystem. By default, the JBoss Server Migration Tool automatically migrates the **`jacorb`** subsystem configuration to its replacement **`iiop-openjdk`** subsystem configuration and logs the results to its log file and to the console.

To skip the automatic migration to the **`iiop-openjdk`** subsystem configuration, set the **`subsystem.jacorb.migrate.skip`** environment property value to **`true`**.

6.2.6. Migrate the Web Subsystem

The **`web`** subsystem is deprecated in JBoss EAP 7 and is replaced by the **`undertow`** subsystem. By default, the JBoss Server Migration Tool automatically migrates the **`web`** subsystem configuration to its replacement **`undertow`** subsystem configuration and logs the results to its log file and to the console.

To skip automatic migration of the **`web`** subsystem, set the **`subsystem.web.migrate.skip`** environment property value to **`true`**.

6.2.7. Migrate the Messaging Subsystem

The **`messaging`** subsystem is deprecated in JBoss EAP 7 and is replaced by the **`messaging-activemq`** subsystem. The JBoss Server Migration Tool automatically migrates the **`messaging`** subsystem configuration to its replacement **`messaging-activemq`** subsystem configuration and logs the results to its log file and to the console.

To skip automatic migration of the **`messaging`** subsystem, set the **`subsystem.messaging.migrate.skip`** environment property value to **`true`**.

6.2.8. Update the Infinispan Subsystem

The JBoss Server Migration Tool updates the **`infinispan`** subsystem configuration to better align with the new default JBoss EAP 7.1 configurations.

- It adds the EJB cache container, which is present in the JBoss EAP 7.1 default configuration, to configurations where it is not already included.
- It adds the server cache container, which is present in the JBoss EAP 7.1 default configuration.
- It updates the module name in the Hibernate cache container configuration.

The JBoss Server Migration Tool automatically updates the **`infinispan`** subsystem configuration and logs the results to its log file and to the console.

You can customize the update of the **`infinispan`** system by setting the following environment properties.

Property Name	Property Description
subsystem.infinispan.update.skip	If set to **`true`**, skip the update of the **`infinispan`** subsystem.
subsystem.infinispan.update.add-infinispan-ejb-cache.skip	If set to **`true`**, do not add the EJB cache container.
subsystem.infinispan.update.add-infinispan-server-cache.skip	If set to **`true`**, do not add the server cache container.
subsystem.infinispan.update.fix-hibernate-cache-module-name.skip	If set to **`true`**, do not update the module name in the Hibernate cache container configuration.

6.2.9. Update the EE Subsystem

The JBoss Server Migration Tool updates the **ee** subsystem to configure the new Java EE 7 features supported in JBoss EAP 7.1.

- It configures instances of Java EE 7 concurrency utilities, such as container-managed executors, that are present in the JBoss EAP 7.1 default configuration and logs the results to its log file and to the console.
- It defines the Java EE 7 default resources, such as the default datasource, that are present in default JBoss EAP 6.4 configuration. If the resources are not found, the tool lists all available resources in the configuration, and then provides a prompt to select a resource from the list or to provide the JNDI address of the resource that should be set as the default.

NOTE

JNDI names that are specified are assumed to be valid. JNDI names are not validated by the tool.

The JBoss Server Migration Tool automatically updates the **ee** subsystem configuration and logs the results to its log file and to the console.

You can customize the update of the **ee** system by setting the following environment properties.

Property Name	Property Description
subsystem.ee.update.skip	If set to **true**, skip the update of the **ee** subsystem.
subsystem.ee.update.setup-ee-concurrency-utilities.skip	If set to **true**, do not add the default instances of concurrency utilities.
subsystem.ee.update.setup-javaee7-default-bindings.skip	If set to **true**, do not set up Java EE 7 default resources.
subsystem.ee.update.setup-javaee7-default-bindings.defaultDataSourceName	Specifies the name of the default datasource to look for in the source configuration.
subsystem.ee.update.setup-javaee7-default-bindings.defaultDataSourceJndiName	Specifies the JNDI name for the default datasource.
subsystem.ee.update.setup-javaee7-default-bindings.defaultJmsConnectionFactoryName	Specifies the name of the default JMS connection factory.
subsystem.ee.update.setup-javaee7-default-bindings.defaultJmsConnectionFactoryJndiName	Specifies the JNDI name for the default JMS connection factory.

Configuring Concurrency Utilities in the EE Subsystem
If you choose to configure the Java EE 7 concurrency utilities, then the tool automatically configures the instances that are present in the default JBoss EAP 7.1 configurations and logs the results to its log file and to the console.

```
INFO  [ServerMigrationTask#49] Default ContextService added to EE
subsystem configuration.
INFO  [ServerMigrationTask#49] Default ManagedThreadFactory added to EE
subsystem configuration.
INFO  [ServerMigrationTask#49] Default ManagedExecutorService added to EE
subsystem configuration.
INFO  [ServerMigrationTask#49] Default ManagedScheduledExecutorService
added to EE subsystem configuration.
```

Configuring Default Resources in the EE Subsystem
When defining the Java EE 7 default resources the tool automatically selects those that are present in the default JBoss EAP 7.1 configuration.

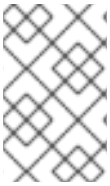
```
INFO  [ServerMigrationTask#50] Java EE Default Datasource configured with
JNDI name java:jboss/datasources/ExampleDS.
```

If no default resource is found, the tool lists all resources that are available in the configuration, and then provides a prompt to select the default resource or to provide the JNDI address of the resource that should be set as the default.

The following is an example of the interaction that occurs when migrating a configuration file with an **ExampleDS** datasource.

```
INFO  [ServerMigrationTask#22] Default datasource not found.
```

```
0. ExampleDS
1. Unconfigured data source, I want to enter the JNDI name...
Please select Java EE's Default Datasource: (0): 0
INFO  [ServerMigrationTask#22] Java EE Default Datasource configured with
JNDI name java:jboss/datasources/ExampleDS.

Save this Java EE Default Datasource JNDI name and use it when migrating
other config files?
yes/no? y
```



NOTE

If you run the JBoss Server Migration Tool in non-interactive mode and the expected JBoss EAP 6.4 default resources, such as the default JMS connection factory, are not available, the tool does not configure those resources.

6.2.10. Update the EJB 3 Subsystem

The JBoss Server Migration Tool makes the following updates to the **`ejb3`** subsystem to better align with the new default JBoss EAP 7.1 configurations.

- It updates the remote service configuration to reference the new HTTP connector.
- It configures the **`default-sfsb-passivation-disabled-cache`** attribute to use the **`default-sfsb-cache`**.
- It replaces the legacy passivation store and cache configurations with the JBoss EAP 7.1 default values.

The JBoss Server Migration Tool automatically updates the **`ejb3`** subsystem configuration and logs the results to its log file and to the console.

You can customize the update of the **`ejb3`** system by setting the following environment properties.

Property Name	Property Description
subsystem.ejb3.update.skip	If set to **`true`**, skip the update of the **`ejb3`** subsystem.
subsystem.ejb3.update.add-infinispan-passivation-store-and-distributable-cache.skip	If set to **`true`**, do not replace the **`passivation-store`** and **`cache`** configurations.
subsystem.ejb3.update.setup-default-sfsb-passivation-disabled-cache.skip	If set to **`true`**, do not update the **`default-sfsb-passivation-disabled-cache`** configuration.
subsystem.ejb3.update.activate-ejb3-remoting-http-connector.skip	If set to **`true`**, do not update the EJB remoting configuration.

6.2.11. Update the JGroups Subsystem

The JBoss Server Migration Tool updates the **`jgroups`** subsystem to align with the new JBoss EAP 7.1 configurations.

- It replaces the **MERGE2** protocol with **MERGE3**.
- It replaces the **FD** protocol with **FD_ALL**.
- It replaces the **pbcast.NAKACK** protocol with **pbcast.NAKACK2**.
- It replaces the **UNICAST2** protocol with **UNICAST3**.
- It removes the **RSVP** protocol.

To skip the automatic migration of the **jgroups** subsystem, set the **subsystem.jgroups.update.skip** environment property to **true**.

6.2.12. Update the Remoting Subsystem

JBoss EAP 7 introduces a new HTTP connector that replaces all legacy remoting protocols and ports using a single port. The JBoss Server Migration Tool automatically updates the **remoting** subsystem to use the new HTTP connector.

To skip the automatic update of the **remoting** subsystem configuration, set the **subsystem.remoting.update.skip** environment property to **true**.

6.2.13. Update the Undertow Subsystem

In addition to migrating the **web** subsystem for JBoss EAP 7.1, the JBoss Server Migration Tool updates its replacement **undertow** subsystem to add the new features it supports.

- It sets the default HTTP listener redirect socket.
- It adds support for Java EE 7 WebSockets.
- It sets the default HTTPS listener.
- It adds support for HTTP2.
- It sets the default **Server** response header.
- It sets the default **X-Powered-By** response header.
- It sets the default HTTP **Invoker**.

The JBoss Server Migration Tool automatically updates the **undertow** subsystem configuration and logs the results to its log file and to the console.

You can customize the update of the **undertow** system by setting the following environment properties.

Property Name	Property Description
subsystem.undertow.update.skip	If set to **true**, skip the update of the **undertow** subsystem.
subsystem.undertow.update.set-default-http-listener-redirect-socket.skip	If set to **true**, do not set the default HTTP listener redirect socket.

Property Name	Property Description
subsystem.undertow.update.add-undertow-websockets.skip	If set to **`true`**, do not add support for WebSockets.
subsystem.undertow.update.add-undertow-https-listener.skip	If set to **`true`**, do not set the default HTTPS listener.
subsystem.undertow.update.enable-http2.skip	If set to **`true`**, do not add support for HTTP2.
subsystem.undertow.update.add-response-header.server-header.skip	If set to **`true`**, do not set the default **`Server`** response header.
subsystem.undertow.update.add-response-header.x-powered-by-header.skip	If set to **`true`**, do not set the default **`X-Powered-By`** response header.
subsystem.undertow.update.add-http-invoker.skip	If set to **`true`**, do not set the default HTTP **`Invoker`**.

6.2.14. Update the Messaging-ActiveMQ Subsystem

In addition to migrating the **`messaging`** subsystem for JBoss EAP 7.1, the JBoss Server Migration Tool updates its replacement **`messaging-activemq`** subsystem to add the new features it supports.

- It adds the default HTTP connector and acceptor to enable the new HTTP-based remote messaging clients.

The JBoss Server Migration Tool automatically updates the **`messaging-activemq`** subsystem configuration and logs the results to its log file and to the console.

To skip the automatic update of the **`messaging-activemq`** subsystem, set the **`subsystem.messaging-activemq.update.skip`** environment property to **`true`**.

6.2.15. Add the Batch JBeret Subsystem

The JBoss EAP 7.1 **`batch-jberet`** subsystem provides support for JSR 352: Batch Applications for the Java Platform. The JBoss Server Migration Tool automatically adds the default **`batch-jberet`** subsystem configuration to the migrated configuration file.

To skip the addition of the **`batch-jberet`** subsystem configuration, set the **`subsystem.batch-jberet.add.skip`** environment property to **`true`**.

6.2.16. Add the Core Management Subsystem

The JBoss EAP 7.1 **`core-management`** subsystem provides management-related resources, which were previously configured in the management core service. Examples of these resources include the ability to view a history of configuration changes made to the server and the ability to monitor for server lifecycle events. The JBoss Server Migration Tool automatically adds the default **`core-management`** subsystem configuration to the migrated configuration file.

To skip the addition of the **core-management** subsystem configuration, set the **subsystem.core-management.add.skip** environment property to **true**.

6.2.17. Add the Elytron Subsystem

The JBoss EAP 7.1 **elytron** subsystem provides a single unified security framework that can manage and configure access for both standalone servers and managed domains. It can also be used to configure security access for applications deployed to JBoss EAP servers. The JBoss Server Migration Tool automatically adds the default **elytron** subsystem configuration to the migrated configuration file.

To skip the addition of the **elytron** subsystem configuration, set the **subsystem.elytron.add.skip** environment property to **true**.

6.2.18. Add the Request Controller Subsystem

The JBoss EAP 7.1 **request-controller** subsystem provides congestion control and graceful shutdown functionality. The JBoss Server Migration Tool automatically adds the default **request-controller** subsystem configuration to the migrated configuration file.

To skip the addition of the **request-controller** subsystem configuration, set the **subsystem.request-controller.add.skip** environment property to **true**.

6.2.19. Add the Security Manager Subsystem

The JBoss EAP 7.1 **security-manager** subsystem provides support for Java EE 7 security permissions. The JBoss Server Migration Tool automatically adds the default **security-manager** subsystem configuration to the migrated configuration file.

To skip the addition of the **security-manager** subsystem configuration, set the **subsystem.security-manager.add.skip** environment property to **true**.

6.2.20. Add the Singleton Subsystem

The JBoss EAP 7.1 **singleton** subsystem provides singleton functionality. The JBoss Server Migration Tool automatically adds the default **singleton** subsystem configuration to the migrated configuration file.

To skip the addition of the **singleton** subsystem configuration, set the **subsystem.singleton.add.skip** environment property to **true**.

6.2.21. Update the Unsecure Interface

The JBoss Server Migration Tool automatically updates the **unsecure** interface configuration to align with the JBoss EAP 7.1 default configuration.

To skip configuration of the **unsecure** interface, set the **interface.unsecure.update.skip** environment property to **true**.

6.2.22. Set Up the Private Interface

The JBoss EAP 7 default configuration uses a new private interface on all **jgroups** socket bindings. The JBoss Server Migration Tool automatically updates the migrated **jgroups** socket bindings to use same configuration.

To skip the configuration of the private interface, set the **`interface.private.setup.skip environment`** property to **`true`**.

6.2.23. Add Socket Binding Port Expressions

The JBoss EAP 7.1 default configurations use value expressions for the port attribute of the following socket bindings:

- **`ajp`**
- **`http`**
- **`https`**

The JBoss Server Migration Tool automatically adds these value expressions to the migrated server configurations.

To skip update of the socket binding port expressions, set the **`socket-bindings.add-port-expressions.skip`** environment property to **`true`**.

6.2.24. Add Socket Binding Multicast Address Expressions

The JBoss EAP 7.1 default configuration uses value expressions in the **`multicast-address`** attribute of mod_cluster socket bindings. The JBoss Server Migration Tool automatically adds these value expressions to the migrated configuration files.

To skip the addition of these expressions, set the **`socket-bindings.multicast-address.add-expressions.skip`** environment property to **`true`**.

6.2.25. Add the Load Balancer Profile

The JBoss EAP 7.1 includes a new default profile specifically tailored for hosts that serve as load balancers. The JBoss Server Migration Tool automatically adds and configures this profile to all migrated managed domain configurations.

To skip the addition of this profile, set the **`profile.load-balancer.add.skip`** environment property to **`true`**.

6.2.26. Add Host Excludes

The JBoss EAP 7.1 domain controller can potentially include functionality that is not supported by hosts running on older versions of the server. The **`host-exclude`** configuration specifies the resources that should be hidden from those older versions.

When migrating a domain controller configuration, the JBoss Server Migration Tool adds to or replaces the source server's **`host-exclude`** configuration with the configuration of the target JBoss EAP 7.1 server.

The JBoss Server Migration Tool automatically updates the **`host-exclude`** configuration and logs the results to its log file and to the console.

```
INFO  Host-excludes configuration added.
```

6.2.27. Remove the PermGen Attributes from the JVM Configurations

The usage of **`PermGen`** attributes in JVM configurations is deprecated in JBoss EAP 7. The JBoss Server Migration Tool automatically removes them from all JVM configurations for all server groups.

To skip removal of the **`PermGen`** attributes, set the **`jvms.remove-permgen-attributes.skip`** environment property value to **`true`**.

6.2.28. Migrate Deployments

The JBoss Server Migration Tool can migrate the following types of managed domain deployment configurations.

- Deployments it references, also known as persistent deployments.
- Deployment overlays it references.

The migration of a deployment consists of installing related file resources on the target server, and possibly updating the migrated configuration.

The JBoss Server Migration Tool is preconfigured to skip deployments by default when running in non-interactive mode. To enable migration of deployments, set the **`deployments.migrate-deployments.skip`** environment property to **`false`**.

IMPORTANT

Be aware that when you run the JBoss Server Migration Tool in interactive mode and enter invalid input, the resulting behavior depends on the value of the **`deployments.migrate-deployments`** environment property.

- If **`deployments.migrate-deployments.skip`** is set to **`false`** and you enter invalid input, the tool will try to migrate the deployments.
- If **`deployments.migrate-deployments.skip`** is set to **`true`** and you enter invalid input, the tool will skip the deployments migration.

To enable the migration of specific types of deployments, see the following sections.

WARNING

The JBoss Server Migration Tool does not determine whether deployed resources are compatible with the target server. This means that applications or resources might not deploy, might not work as expected, or might not work at all. Also be aware that artifacts such as JBoss EAP 6.4 **`*-jms.xml`** configuration files are copied without modification and can cause the JBoss EAP server to boot with errors.

Red Hat recommends that you use the Red Hat Application Migration Toolkit to analyze deployments to determine compatibility among different JBoss EAP servers. For more information, see the Red Hat Application Migration Toolkit product documentation.

6.2.28.1. Migrate Persistent Deployments

To enable migration of persistent deployments when running in non-interactive mode, set the **`deployments.migrate-persistent-deployments.skip`** environment property to **`false`**.

The JBoss Server Migration Tool searches for any persistent deployment references and lists them to the console.

```
INFO  [ServerMigrationTask#67] Persistent deployments found: [cmtool-
helloworld3.war, cmtool-helloworld4.war, cmtool-helloworld2.war, cmtool-
helloworld1.war]
```

The processing workflow then depends on whether you are running the tool in interactive mode or in non-interactive mode, as described below.

Migrating Persistent Deployments in Non-interactive Mode

If you run the tool in non-interactive mode, the tool uses the preconfigured properties to determine whether to migrate the persistent deployments. Persistent deployments are migrated only if both the **`deployments.migrate-deployments.skip`** and **`deployments.migrate-persistent-deployments.skip`** properties are set to **`false`**.

Migrating Persistent Deployments in Interactive Mode

If you run the tool in interactive mode, the JBoss Server Migration Tool prompts you for each deployment using the following workflow.

1. After printing the persistent deployments it finds to the console, you see the following prompt.

   ```
   This tool is not able to assert if persistent deployments found are
   compatible with the target server, skip persistent deployments
   migration?
   yes/no?
   ```

   - Respond with **yes** to skip migration of persistent deployments. All deployment references are removed from the migrated configuration and you end this part of the migration process.
   - Respond with **no** to continue with the migration.

2. If you choose to continue, you see the following prompt.

   ```
   Migrate all persistent deployments found?
   yes/no?
   ```

   - Respond with **yes** to automatically migrate all deployments and end this part of the migration process.
   - Respond with **no** to continue with the migration.

3. If you choose to continue, you receive a prompt asking to confirm the migration for each referenced deployment.

   ```
   Migrate persistent deployment 'helloworld01.war'?
   yes/no?
   ```

   - Respond with **yes** to migrate the deployment.

- Respond with **no** to remove the deployment from the migrated configuration.

```
INFO  [ServerMigrationTask#68] Removed persistent deployment from
configuration /deployment=helloworld01.war
```

6.2.28.2. Migrate Deployment Overlays

The migration of deployment overlays is a fully automated process. If you have enabled migration of deployments by setting the **`deployments.migrate-deployments.skip`** environment property to **`false`**, the JBoss Server Migration Tool searches for deployment overlays referenced in the standalone server configuration that are linked to migrated deployments. It automatically migrates those that are found, removes those that are not referenced, and logs the results to its log file and to the console.

6.3. MIGRATING A JBOSS EAP 6.4 HOST CONFIGURATION TO JBOSS EAP 7.1

By default, the JBoss Server Migration Tool performs the following tasks when migrating a host server configuration from JBoss EAP 6.4 to JBoss EAP 7.1.

- Migrate any referenced modules.
- Migrate any referenced paths.
- Add the **`core-management`** subsystem.
- Add the **`elytron`** subsystem.
- Add the **`jmx`** subsystem.
- Remove the **`unsecure`** interface.
- Set up HTTP Upgrade management.
- Remove the **`PermGen`** attributes from the JVM configurations.
- Migrate compatible security realms.
- Add the default SSL server identity to the **`ApplicationRealm`**.

6.3.1. Migrate Referenced Modules

A configuration that is migrated from a source server to a target server might reference or depend on a module that is not installed on the target server. The JBoss Server Migration Tool detects this and automatically migrates the referenced modules, plus their dependent modules, from the source server to the target server.

A module referenced by a host server configuration is migrated using the following process.

- A module referenced by a security realm configuration is migrated as a plug-in module.
- A module referenced by a vault configuration is migrated to the new configuration.

The console logs a message noting the module ID for any module that is migrated. It is possible to exclude the migration of specific modules by specifying the module ID in the **`modules.excludes`** environment property. See Configuring the Migration of Modules for more information.

6.3.2. Migrate Referenced Paths

A configuration that is migrated from a source server to a target server might reference or depend on file paths and directories that must also be migrated to the target server. The JBoss Server Migration Tool does not migrate absolute path references. It only migrates files or directories that are configured as relative to the source configuration. The console logs a message noting each path that is migrated.

The JBoss Server Migration Tool automatically migrates the following path references:

- Vault keystore and encrypted file's directory.

To skip the migration of referenced paths, set the **`paths.migrate-paths-requested-by-configuration.vault.skip`** environment property to **`true`**.

6.3.3. Add the Core Management Subsystem

The JBoss EAP 7.1 **`core-management`** subsystem provides management-related resources, which were previously configured in the management core service. Examples of these resources include the ability to view a history of configuration changes made to the server and the ability to monitor for server lifecycle events. The JBoss Server Migration Tool automatically adds the default **`core-management`** subsystem configuration to the migrated configuration file.

To skip the addition of the **`core-management`** subsystem configuration, set the **`subsystem.core-management.add.skip`** environment property to **`true`**.

6.3.4. Add the Elytron Subsystem

The JBoss EAP 7.1 **`elytron`** subsystem provides a single unified security framework that can manage and configure access for both standalone servers and managed domains. It can also be used to configure security access for applications deployed to JBoss EAP servers. The JBoss Server Migration Tool automatically adds the default **`elytron`** subsystem configuration to the migrated configuration file.

To skip the addition of the **`elytron`** subsystem configuration, set the **`subsystem.elytron.add.skip`** environment property to **`true`**.

6.3.5. Add the JMX Subsystem to the Host Configuration

The JBoss EAP 7 **`jmx`** subsystem provides the ability to manage and monitor systems. The JBoss Server Migration Tool automatically adds this subsystem to the migrated configuration file.

To skip the addition of the **`jmx`** subsystem configuration, set the **`subsystem.jmx.add.skip`** environment property to **`true`**.

6.3.6. Remove the unsecure Interface

The JBoss Server Migration Tool automatically removes the **`unsecure`** interface configuration to align with the JBoss EAP 7.1 default configuration.

To skip removal of the **`unsecure`** interface, set the **`interface.unsecure.remove.skip`** environment property to **`true`**.

6.3.7. Set Up HTTP Upgrade Management

The addition of Undertow in JBoss EAP 7 added HTTP Upgrade, which allows for multiple protocols to be multiplexed over a single port. This means a management client can make an initial connection over HTTP, but then send a request to upgrade that connection to another protocol. The JBoss Server Migration Tool automatically updates the configuration to support HTTP Upgrade management.

To skip configuration of HTTP Upgrade management, set the **`management.setup-http-upgrade.skip`** environment property to **`true`**.

6.3.8. Remove the PermGen Attributes from the JVM Configurations

The usage of **`PermGen`** attributes in JVM configurations is deprecated in JBoss EAP 7. The JBoss Server Migration Tool automatically removes them from all JVM configurations for all server groups.

To skip removal of the **`PermGen`** attributes, set the **`jvms.remove-permgen-attributes.skip`** environment property value to **`true`**.

6.3.9. Migrate Compatible Security Realms

Because the JBoss EAP 7.1 security realm configurations are fully compatible with the JBoss EAP 6.4 security realm configurations, they require no update by the JBoss Server Migration Tool. However, if the **`application-users.properties`**, **`application-roles.properties`**, **`mgmt-users.properties`**, and **`mgmt-groups.properties`** files are not referenced using an absolute path, the tool copies them to the path expected by the migrated configuration file.

To skip the security realms migration, set the **`security-realms.migrate-properties.skip environment`** property to **`true`**.

6.3.10. Add the Default SSL Server Identity to the ApplicationRealm

The JBoss EAP 7.1 default configuration includes an SSL server identity for the default **`ApplicationRealm`** security realm. The JBoss Server Migration Tool automatically adds this identity to the migrated configuration files.

To skip the addition of this identity, set the **`security-realm.ApplicationRealm.add-ssl-server-identity.skip`** environment property to **`true`**.

CHAPTER 7. MIGRATING JBOSS EAP 7.0 CONFIGURATIONS TO JBOSS EAP 7.1

7.1. MIGRATING A JBOSS EAP 7.0 STANDALONE SERVER TO JBOSS EAP 7.1

By default, the JBoss Server Migration Tool performs the following tasks when migrating a standalone server configuration from JBoss EAP 7.0 to JBoss EAP 7.1.

- Migrate any referenced modules.
- Migrate any referenced paths.
- Update the **`infinispan`** subsystem.
- Update the **`undertow`** subsystem.
- Add the **`core-management`** subsystem.
- Add the **`elytron`** subsystem.
- Add socket binding multicast address expressions.
- Migrate compatible security realms.
- Add the default SSL server identity to the **`ApplicationRealm`**.
- Migrate deployments.

7.1.1. Remove Unsupported Subsystems

The JBoss Server Migration Tool removes all unsupported subsystem configurations and extensions from migrated server configurations. The tool logs each subsystem and extension to its log file and to the console as it is removed.

NOTE

Any subsystem that was not supported in JBoss EAP 7.0, but was added by an administrator to that server, is also not supported in JBoss EAP 7.1 and will be removed.

To skip removal of the unsupported subsystems, set the **`subsystems.remove-unsupported-subsystems.skip`** environment property to **`true`**.

7.1.2. Migrate Referenced Modules

A configuration that is migrated from a source server to a target server might reference or depend on a module that is not installed on the target server. The JBoss Server Migration Tool detects this and automatically migrates the referenced modules, plus their dependent modules, from the source server to the target server.

A module referenced by a standalone server configuration is migrated using the following process.

- A module referenced by a security realm configuration is migrated as a plug-in module.

- A module referenced by the **`datasource`** subsystem configuration is migrated as a datasource driver module.
- A module referenced by the **`ee`** subsystem configuration is migrated as a global module.
- A module referenced by the **`naming`** subsystem configuration is migrated as an object factory module.
- A module referenced by the **`messaging`** subsystem configuration is migrated as a JMS bridge module.
- A module referenced by a vault configuration is migrated to the new configuration.

The console logs a message noting the module ID for any module that is migrated. It is possible to exclude the migration of specific modules by specifying the module ID in the **`modules.excludes`** environment property. See Configuring the Migration of Modules for more information.

7.1.3. Migrate Referenced Paths

A configuration that is migrated from a source server to a target server might reference or depend on file paths and directories that must also be migrated to the target server. The JBoss Server Migration Tool does not migrate absolute path references. It only migrates files or directories that are configured as relative to the source configuration. The console logs a message noting each path that is migrated.

The JBoss Server Migration Tool automatically migrates the following path references:

- Vault keystore and encrypted file's directory.

To skip the migration of referenced paths, set the **`paths.migrate-paths-requested-by-configuration.vault.skip`** environment property to **`true`**.

7.1.4. Update the Infinispan Subsystem

The JBoss Server Migration Tool updates the **`infinispan`** subsystem configuration to better align with the new default JBoss EAP 7.1 configurations.

- It adds the EJB cache container, which is present in the JBoss EAP 7.1 default configuration, to configurations where it is not already included.
- It updates the module name in the Hibernate cache container configuration.

The JBoss Server Migration Tool automatically updates the **`infinispan`** subsystem configuration and logs the results to its log file and to the console.

You can customize the update of the **`infinispan`** system by setting the following environment properties.

Property Name	Property Description
subsystem.infinispan.update.skip	If set to **`true`**, skip the update of the **`infinispan`** subsystem.
subsystem.infinispan.update.add-infinispan-ejb-cache.skip	If set to **`true`**, do not add the EJB cache container.

Property Name	Property Description
subsystem.infinispan.update.fix-hibernate-cache-module-name.skip	If set to **`true`**, do not update the module name in the Hibernate cache container configuration.

7.1.5. Update the Undertow Subsystem

In addition to migrating the **`web`** subsystem for JBoss EAP 7.1, the JBoss Server Migration Tool updates its replacement **`undertow`** subsystem to add the new features it supports.

- It sets the default HTTP listener redirect socket.
- It adds support for Java EE 7 WebSockets.
- It sets the default HTTPS listener.
- It adds support for HTTP2.
- It sets the default **`Server`** response header.
- It sets the default **`X-Powered-By`** response header.
- It sets the default HTTP **`Invoker`**.

The JBoss Server Migration Tool automatically updates the **`undertow`** subsystem configuration and logs the results to its log file and to the console.

You can customize the update of the **`undertow`** system by setting the following environment properties.

Property Name	Property Description
subsystem.undertow.update.skip	If set to **`true`**, skip the update of the **`undertow`** subsystem.
subsystem.undertow.update.set-default-http-listener-redirect-socket.skip	If set to **`true`**, do not set the default HTTP listener redirect socket.
subsystem.undertow.update.add-undertow-websockets.skip	If set to **`true`**, do not add support for WebSockets.
subsystem.undertow.update.add-undertow-https-listener.skip	If set to **`true`**, do not set the default HTTPS listener.
subsystem.undertow.update.enable-http2.skip	If set to **`true`**, do not add support for HTTP2.
subsystem.undertow.update.add-response-header.server-header.skip	If set to **`true`**, do not set the default **`Server`** response header.
subsystem.undertow.update.add-response-header.x-powered-by-header.skip	If set to **`true`**, do not set the default **`X-Powered-By`** response header.

Property Name	Property Description
subsystem.undertow.update.add-http-invoker.skip	If set to **true**, do not set the default HTTP **Invoker**.

7.1.6. Add the Core Management Subsystem

The JBoss EAP 7.1 **core-management** subsystem provides management-related resources, which were previously configured in the management core service. Examples of these resources include the ability to view a history of configuration changes made to the server and the ability to monitor for server lifecycle events. The JBoss Server Migration Tool automatically adds the default **core-management** subsystem configuration to the migrated configuration file.

To skip the addition of the **core-management** subsystem configuration, set the **subsystem.core-management.add.skip** environment property to **true**.

7.1.7. Add the Elytron Subsystem

The JBoss EAP 7.1 **elytron** subsystem provides a single unified security framework that can manage and configure access for both standalone servers and managed domains. It can also be used to configure security access for applications deployed to JBoss EAP servers. The JBoss Server Migration Tool automatically adds the default **elytron** subsystem configuration to the migrated configuration file.

To skip the addition of the **elytron** subsystem configuration, set the **subsystem.elytron.add.skip** environment property to **true**.

7.1.8. Add Socket Binding Multicast Address Expressions

The JBoss EAP 7.1 default configuration uses value expressions in the **multicast-address** attribute of mod_cluster socket bindings. The JBoss Server Migration Tool automatically adds these value expressions to the migrated configuration files.

To skip the addition of these expressions, set the **socket-bindings.multicast-address.add-expressions.skip** environment property to **true**.

7.1.9. Migrate Compatible Security Realms

Because the JBoss EAP 7.1 security realm configurations are fully compatible with the JBoss EAP 7.0 security realm configurations, they require no update by the JBoss Server Migration Tool. However, if the **application-users.properties**, **application-roles.properties**, **mgmt-users.properties**, and **mgmt-groups.properties** files are not referenced using an absolute path, the tool copies them to the path expected by the migrated configuration file.

To skip the security realms migration, set the **security-realms.migrate-properties.skip environment** property to **true**.

7.1.10. Add the Default SSL Server Identity to the ApplicationRealm

The JBoss EAP 7.1 default configuration includes an SSL server identity for the default **ApplicationRealm** security realm. The JBoss Server Migration Tool automatically adds this identity to the migrated configuration files.

To skip the addition of this identity, set the **`security-realm.ApplicationRealm.add-ssl-server-identity.skip`** environment property to **`true`**.

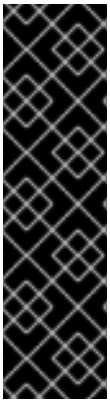
7.1.11. Migrate Deployments

The JBoss Server Migration Tool can migrate the following types of standalone server deployment configurations.

- Deployments it references, also known as persistent deployments.
- Deployments found in directories monitored by its deployment scanners.
- Deployment overlays it references.

The migration of a deployment consists of installing related file resources on the target server, and possibly updating the migrated configuration.

The JBoss Server Migration Tool is preconfigured to skip deployments by default when running in non-interactive mode. To enable migration of deployments, set the **`deployments.migrate-deployments.skip`** environment property to **`false`**.



IMPORTANT

Be aware that when you run the JBoss Server Migration Tool in interactive mode and enter invalid input, the resulting behavior depends on the value of the **`deployments.migrate-deployments`** environment property.

- If **`deployments.migrate-deployments.skip`** is set to **`false`** and you enter invalid input, the tool will try to migrate the deployments.
- If **`deployments.migrate-deployments.skip`** is set to **`true`** and you enter invalid input, the tool will skip the deployments migration.

To enable the migration of specific types of deployments, see the following sections.

WARNING

The JBoss Server Migration Tool does not determine whether deployed resources are compatible with the target server. This means that applications or resources might not deploy, might not work as expected, or might not work at all. Also be aware that artifacts such as JBoss EAP 6.4 **`*-jms.xml`** configuration files are copied without modification and can cause the JBoss EAP server to boot with errors.

Red Hat recommends that you use the Red Hat Application Migration Toolkit to analyze deployments to determine compatibility among different JBoss EAP servers. For more information, see the Red Hat Application Migration Toolkit product documentation.

7.1.11.1. Migrate Persistent Deployments

To enable migration of persistent deployments when running in non-interactive mode, set the **`deployments.migrate-persistent-deployments.skip`** environment property to **`false`**.

The JBoss Server Migration Tool searches for any persistent deployment references and lists them to the console.

```
INFO  [ServerMigrationTask#67] Persistent deployments found: [cmtool-
helloworld3.war, cmtool-helloworld4.war, cmtool-helloworld2.war, cmtool-
helloworld1.war]
```

The processing workflow then depends on whether you are running the tool in interactive mode or in non-interactive mode, as described below.

Migrating Persistent Deployments in Non-interactive Mode

If you run the tool in non-interactive mode, the tool uses the preconfigured properties to determine whether to migrate the persistent deployments. Persistent deployments are migrated only if both the **`deployments.migrate-deployments.skip`** and **`deployments.migrate-persistent-deployments.skip`** properties are set to **`false`**.

Migrating Persistent Deployments in Interactive Mode

If you run the tool in interactive mode, the JBoss Server Migration Tool prompts you for each deployment using the following workflow.

1. After printing the persistent deployments it finds to the console, you see the following prompt.

   ```
   This tool is not able to assert if persistent deployments found are
   compatible with the target server, skip persistent deployments
   migration?
   yes/no?
   ```

   - Respond with **yes** to skip migration of persistent deployments. All deployment references are removed from the migrated configuration and you end this part of the migration process.
   - Respond with **no** to continue with the migration.

2. If you choose to continue, you see the following prompt.

   ```
   Migrate all persistent deployments found?
   yes/no?
   ```

   - Respond with **yes** to automatically migrate all deployments and end this part of the migration process.
   - Respond with **no** to continue with the migration.

3. If you choose to continue, you receive a prompt asking to confirm the migration for each referenced deployment.

   ```
   Migrate persistent deployment 'helloworld01.war'?
   yes/no?
   ```

   - Respond with **yes** to migrate the deployment.
   - Respond with **no** to remove the deployment from the migrated configuration.

```
INFO  [ServerMigrationTask#68] Removed persistent deployment from
configuration /deployment=helloworld01.war
```

7.1.11.2. Migrate Deployment Scanner Deployments

Deployment scanners, which are only used in standalone server configurations, monitor a directory for new files and manage their deployment automatically or through special deployment marker files.

To enable migration of deployments that are located in directories watched by a deployment scanner when running in non-interactive mode, set the **`deployments.migrate-deployment-scanner-deployments.skip`** environment property to **`false`**.

When migrating a standalone server configuration, the JBoss Server Migration Tool first searches for any configured deployment scanners. For each scanner found, it searches its monitored directories for deployments marked as deployed and prints the results to the console.

The processing workflow then depends on whether you are running the tool in interactive mode or in non-interactive mode, as described below.

Migrating Deployment Scanner Deployments in Non-interactive Mode

If you run the tool in non-interactive mode, the tool uses the preconfigured properties to determine whether to migrate the deployment scanner deployments. Deployment scanner deployments are migrated only if both the **`deployments.migrate-deployments.skip`** and **`deployments.migrate-deployment-scanner-deployments.skip`** properties are set to **`false`**.

Migrating Deployment Scanner Deployments in Interactive Mode

If you run the tool in interactive mode, the JBoss Server Migration Tool prompts you for each deployment using the following workflow.

1. After printing the deployment scanner deployments it finds to the console, you see the following prompt.

```
This tool is not able to assert if the scanner's deployments found
are compatible with the target server, skip scanner's deployments
migration?
yes/no?
```

   - Respond with **yes** to skip migration of deployment scanner deployments. All deployment references are removed from the migrated configuration and you end this part of the migration process.
   - Respond with **no** to continue with the migration.

2. If you choose to continue, you see the following prompt.

```
Migrate all scanner's deployments found?
yes/no?
```

   - Respond with **yes** to automatically migrate all deployments and end this part of the migration process.
   - Respond with **no** to continue with the migration.

3. If you choose to continue, you receive a prompt asking to confirm the migration for each referenced deployment.

```
Migrate scanner's deployment 'helloworld02.war'?
yes/no?
```

- Respond with **yes** to migrate the deployment.
- Respond with **no** to remove the deployment from the migrated configuration.

```
INFO  [ServerMigrationTask#69] Resource with path
EAP_HOME/standalone/deployments/helloworld02.war migrated.
```

7.1.11.3. Migrate Deployment Overlays

The migration of deployment overlays is a fully automated process. If you have enabled migration of deployments by setting the **`deployments.migrate-deployments.skip`** environment property to **`false`**, the JBoss Server Migration Tool searches for deployment overlays referenced in the standalone server configuration that are linked to migrated deployments. It automatically migrates those that are found, removes those that are not referenced, and logs the results to its log file and to the console.

7.2. MIGRATING A JBOSS EAP 7.0 MANAGED DOMAIN TO JBOSS EAP 7.1

WARNING

When you use the JBoss Server Migration Tool to migrate a domain controller to a new release of JBoss EAP, it can cause the hosts that connect to it, but still run on a previous release of the server, to fail to boot. For this reason, you should migrate your hosts before you migrate your domain controller. If you do not want to migrate your hosts, be sure to read the following information before you begin.

Review Configure a JBoss EAP 7.1 Domain Controller to Administer JBoss EAP 7.0 Instances in the *Configuration Guide* for JBoss EAP. Pay particular attention to the section entitled Prevent the JBoss EAP 7.0 Instances from Receiving JBoss EAP 7.1 Updates.

For additional information, see Managing Multiple JBoss EAP Versions in the *Configuration Guide* for JBoss EAP.

By default, the JBoss Server Migration Tool performs the following tasks when migrating a managed domain configuration from JBoss EAP 7.0 to JBoss EAP 7.1.

- Migrate any referenced modules.
- Migrate any referenced paths.
- Update the **`infinispan`** subsystem.

- Update the **`undertow`** subsystem.
- Add the **`core-management`** subsystem.
- Add the **`elytron`** Subsystem.
- Add socket binding multicast address expressions.
- Add the load balancer profile.
- Add the host excludes configuration.
- Migrate deployments.

7.2.1. Remove Unsupported Subsystems

The JBoss Server Migration Tool removes all unsupported subsystem configurations and extensions from migrated server configurations. The tool logs each subsystem and extension to its log file and to the console as it is removed.

NOTE

Any subsystem that was not supported in JBoss EAP 7.0, but was added by an administrator to that server, is also not supported in JBoss EAP 7.1 and will be removed.

To skip removal of the unsupported subsystems, set the **`subsystems.remove-unsupported-subsystems.skip`** environment property to **`true`**.

7.2.2. Migrate Referenced Modules

A configuration that is migrated from a source server to a target server might reference or depend on a module that is not installed on the target server. The JBoss Server Migration Tool detects this and automatically migrates the referenced modules, plus their dependent modules, from the source server to the target server.

A module referenced by a managed domain configuration is migrated using the following process.

- A module referenced by a security realm configuration is migrated as a plug-in module.
- A module referenced by the **`datasource`** subsystem configuration is migrated as a datasource driver module.
- A module referenced by the **`ee`** subsystem configuration is migrated as a global module.
- A module referenced by the **`naming`** subsystem configuration is migrated as an object factory module.
- A module referenced by the **`messaging`** subsystem configuration is migrated as a JMS bridge module.
- A module referenced by a vault configuration is migrated to the new configuration.

The console logs a message noting the module ID for any module that is migrated. It is possible to exclude the migration of specific modules by specifying the module ID in the **`modules.excludes`** environment property. See Configuring the Migration of Modules for more information.

7.2.3. Migrate Referenced Paths

A configuration that is migrated from a source server to a target server might reference or depend on file paths and directories that must also be migrated to the target server. The JBoss Server Migration Tool does not migrate absolute path references. It only migrates files or directories that are configured as relative to the source configuration. The console logs a message noting each path that is migrated.

The JBoss Server Migration Tool automatically migrates the following path references:

- Vault keystore and encrypted file's directory.

To skip the migration of referenced paths, set the **`paths.migrate-paths-requested-by-configuration.vault.skip`** environment property to **`true`**.

7.2.4. Update the Infinispan Subsystem

The JBoss Server Migration Tool updates the **`infinispan`** subsystem configuration to better align with the new default JBoss EAP 7.1 configurations.

- It adds the EJB cache container, which is present in the JBoss EAP 7.1 default configuration, to configurations where it is not already included.
- It updates the module name in the Hibernate cache container configuration.

The JBoss Server Migration Tool automatically updates the **`infinispan`** subsystem configuration and logs the results to its log file and to the console.

You can customize the update of the **`infinispan`** system by setting the following environment properties.

Property Name	Property Description
subsystem.infinispan.update.skip	If set to **`true`**, skip the update of the **`infinispan`** subsystem.
subsystem.infinispan.update.add-infinispan-ejb-cache.skip	If set to **`true`**, do not add the EJB cache container.
subsystem.infinispan.update.fix-hibernate-cache-module-name.skip	If set to **`true`**, do not update the module name in the Hibernate cache container configuration.

7.2.5. Update the Undertow Subsystem

In addition to migrating the **`web`** subsystem for JBoss EAP 7.1, the JBoss Server Migration Tool updates its replacement **`undertow`** subsystem to add the new features it supports.

- It sets the default HTTP listener redirect socket.
- It adds support for Java EE 7 WebSockets.
- It sets the default HTTPS listener.
- It adds support for HTTP2.

- It sets the default **`Server`** response header.
- It sets the default **`X-Powered-By`** response header.
- It sets the default HTTP **`Invoker`**.

The JBoss Server Migration Tool automatically updates the **`undertow`** subsystem configuration and logs the results to its log file and to the console.

You can customize the update of the **`undertow`** system by setting the following environment properties.

Property Name	Property Description
subsystem.undertow.update.skip	If set to **`true`**, skip the update of the **`undertow`** subsystem.
subsystem.undertow.update.set-default-http-listener-redirect-socket.skip	If set to **`true`**, do not set the default HTTP listener redirect socket.
subsystem.undertow.update.add-undertow-websockets.skip	If set to **`true`**, do not add support for WebSockets.
subsystem.undertow.update.add-undertow-https-listener.skip	If set to **`true`**, do not set the default HTTPS listener.
subsystem.undertow.update.enable-http2.skip	If set to **`true`**, do not add support for HTTP2.
subsystem.undertow.update.add-response-header.server-header.skip	If set to **`true`**, do not set the default **`Server`** response header.
subsystem.undertow.update.add-response-header.x-powered-by-header.skip	If set to **`true`**, do not set the default **`X-Powered-By`** response header.
subsystem.undertow.update.add-http-invoker.skip	If set to **`true`**, do not set the default HTTP **`Invoker`**.

7.2.6. Add the Core Management Subsystem

The JBoss EAP 7.1 **`core-management`** subsystem provides management-related resources, which were previously configured in the management core service. Examples of these resources include the ability to view a history of configuration changes made to the server and the ability to monitor for server lifecycle events. The JBoss Server Migration Tool automatically adds the default **`core-management`** subsystem configuration to the migrated configuration file.

To skip the addition of the **`core-management`** subsystem configuration, set the **`subsystem.core-management.add.skip`** environment property to **`true`**.

7.2.7. Add the Elytron Subsystem

The JBoss EAP 7.1 **`elytron`** subsystem provides a single unified security framework that can manage and configure access for both standalone servers and managed domains. It can also be used to

configure security access for applications deployed to JBoss EAP servers. The JBoss Server Migration Tool automatically adds the default **`elytron`** subsystem configuration to the migrated configuration file.

To skip the addition of the **`elytron`** subsystem configuration, set the **`subsystem.elytron.add.skip`** environment property to **`true`**.

7.2.8. Add Socket Binding Multicast Address Expressions

The JBoss EAP 7.1 default configuration uses value expressions in the **`multicast-address`** attribute of mod_cluster socket bindings. The JBoss Server Migration Tool automatically adds these value expressions to the migrated configuration files.

To skip the addition of these expressions, set the **`socket-bindings.multicast-address.add-expressions.skip`** environment property to **`true`**.

7.2.9. Add the Load Balancer Profile

The JBoss EAP 7.1 includes a new default profile specifically tailored for hosts that serve as load balancers. The JBoss Server Migration Tool automatically adds and configures this profile to all migrated managed domain configurations.

To skip the addition of this profile, set the **`profile.load-balancer.add.skip`** environment property to **`true`**.

7.2.10. Add Host Excludes

The JBoss EAP 7.1 domain controller can potentially include functionality that is not supported by hosts running on older versions of the server. The **`host-exclude`** configuration specifies the resources that should be hidden from those older versions.

When migrating a domain controller configuration, the JBoss Server Migration Tool adds to or replaces the source server's **`host-exclude`** configuration with the configuration of the target JBoss EAP 7.1 server.

The JBoss Server Migration Tool automatically updates the **`host-exclude`** configuration and logs the results to its log file and to the console.

```
INFO  Host-excludes configuration added.
```

7.2.11. Migrate Deployments

The JBoss Server Migration Tool can migrate the following types of managed domain deployment configurations.

- Deployments it references, also known as persistent deployments.
- Deployment overlays it references.

The migration of a deployment consists of installing related file resources on the target server, and possibly updating the migrated configuration.

The JBoss Server Migration Tool is preconfigured to skip deployments by default when running in non-interactive mode. To enable migration of deployments, set the **`deployments.migrate-deployments.skip`** environment property to **`false`**.

IMPORTANT

Be aware that when you run the JBoss Server Migration Tool in interactive mode and enter invalid input, the resulting behavior depends on the value of the **`deployments.migrate-deployments`** environment property.

- If **`deployments.migrate-deployments.skip`** is set to **`false`** and you enter invalid input, the tool will try to migrate the deployments.
- If **`deployments.migrate-deployments.skip`** is set to **`true`** and you enter invalid input, the tool will skip the deployments migration.

To enable the migration of specific types of deployments, see the following sections.

WARNING

The JBoss Server Migration Tool does not determine whether deployed resources are compatible with the target server. This means that applications or resources might not deploy, might not work as expected, or might not work at all. Also be aware that artifacts such as JBoss EAP 6.4 **`*-jms.xml`** configuration files are copied without modification and can cause the JBoss EAP server to boot with errors.

Red Hat recommends that you use the Red Hat Application Migration Toolkit to analyze deployments to determine compatibility among different JBoss EAP servers. For more information, see the Red Hat Application Migration Toolkit product documentation.

7.2.11.1. Migrate Persistent Deployments

To enable migration of persistent deployments when running in non-interactive mode, set the **`deployments.migrate-persistent-deployments.skip`** environment property to **`false`**.

The JBoss Server Migration Tool searches for any persistent deployment references and lists them to the console.

```
INFO  [ServerMigrationTask#67] Persistent deployments found: [cmtool-
helloworld3.war, cmtool-helloworld4.war, cmtool-helloworld2.war, cmtool-
helloworld1.war]
```

The processing workflow then depends on whether you are running the tool in interactive mode or in non-interactive mode, as described below.

Migrating Persistent Deployments in Non-interactive Mode

If you run the tool in non-interactive mode, the tool uses the preconfigured properties to determine whether to migrate the persistent deployments. Persistent deployments are migrated only if both the **`deployments.migrate-deployments.skip`** and **`deployments.migrate-persistent-deployments.skip`** properties are set to **`false`**.

Migrating Persistent Deployments in Interactive Mode

If you run the tool in interactive mode, the JBoss Server Migration Tool prompts you for each deployment using the following workflow.

1. After printing the persistent deployments it finds to the console, you see the following prompt.

```
This tool is not able to assert if persistent deployments found are
compatible with the target server, skip persistent deployments
migration?
yes/no?
```

   - Respond with **yes** to skip migration of persistent deployments. All deployment references are removed from the migrated configuration and you end this part of the migration process.
   - Respond with **no** to continue with the migration.

2. If you choose to continue, you see the following prompt.

```
Migrate all persistent deployments found?
yes/no?
```

   - Respond with **yes** to automatically migrate all deployments and end this part of the migration process.
   - Respond with **no** to continue with the migration.

3. If you choose to continue, you receive a prompt asking to confirm the migration for each referenced deployment.

```
Migrate persistent deployment 'helloworld01.war'?
yes/no?
```

   - Respond with **yes** to migrate the deployment.
   - Respond with **no** to remove the deployment from the migrated configuration.

```
INFO  [ServerMigrationTask#68] Removed persistent deployment from
configuration /deployment=helloworld01.war
```

7.2.11.2. Migrate Deployment Overlays

The migration of deployment overlays is a fully automated process. If you have enabled migration of deployments by setting the **`deployments.migrate-deployments.skip`** environment property to **`false`**, the JBoss Server Migration Tool searches for deployment overlays referenced in the standalone server configuration that are linked to migrated deployments. It automatically migrates those that are found, removes those that are not referenced, and logs the results to its log file and to the console.

7.3. MIGRATING A JBOSS EAP 7.0 HOST CONFIGURATION TO JBOSS EAP 7.1

By default, the JBoss Server Migration Tool performs the following tasks when migrating a host server configuration from JBoss EAP 7.0 to JBoss EAP 7.1.

- Migrate any referenced modules.

- Migrate any referenced paths.
- Add the **`core-management`** subsystem.
- Add the **`elytron`** subsystem.
- Migrate compatible security realms.
- Add the default SSL server identity to the **`ApplicationRealm`**.

7.3.1. Migrate Referenced Modules

A configuration that is migrated from a source server to a target server might reference or depend on a module that is not installed on the target server. The JBoss Server Migration Tool detects this and automatically migrates the referenced modules, plus their dependent modules, from the source server to the target server.

A module referenced by a host server configuration is migrated using the following process.

- A module referenced by a security realm configuration is migrated as a plug-in module.
- A module referenced by a vault configuration is migrated to the new configuration.

The console logs a message noting the module ID for any module that is migrated. It is possible to exclude the migration of specific modules by specifying the module ID in the **`modules.excludes`** environment property. See Configuring the Migration of Modules for more information.

7.3.2. Migrate Referenced Paths

A configuration that is migrated from a source server to a target server might reference or depend on file paths and directories that must also be migrated to the target server. The JBoss Server Migration Tool does not migrate absolute path references. It only migrates files or directories that are configured as relative to the source configuration. The console logs a message noting each path that is migrated.

The JBoss Server Migration Tool automatically migrates the following path references:

- Vault keystore and encrypted file's directory.

To skip the migration of referenced paths, set the **`paths.migrate-paths-requested-by-configuration.vault.skip`** environment property to **`true`**.

7.3.3. Add the Core Management Subsystem

The JBoss EAP 7.1 **`core-management`** subsystem provides management-related resources, which were previously configured in the management core service. Examples of these resources include the ability to view a history of configuration changes made to the server and the ability to monitor for server lifecycle events. The JBoss Server Migration Tool automatically adds the default **`core-management`** subsystem configuration to the migrated configuration file.

To skip the addition of the **`core-management`** subsystem configuration, set the **`subsystem.core-management.add.skip`** environment property to **`true`**.

7.3.4. Add the Elytron Subsystem

The JBoss EAP 7.1 **`elytron`** subsystem provides a single unified security framework that can manage

and configure access for both standalone servers and managed domains. It can also be used to configure security access for applications deployed to JBoss EAP servers. The JBoss Server Migration Tool automatically adds the default **`elytron`** subsystem configuration to the migrated configuration file.

To skip the addition of the **`elytron`** subsystem configuration, set the **`subsystem.elytron.add.skip`** environment property to **`true`**.

7.3.5. Migrate Compatible Security Realms

Because the JBoss EAP 7.1 security realm configurations are fully compatible with the JBoss EAP 7.0 security realm configurations, they require no update by the JBoss Server Migration Tool. However, if the **`application-users.properties`**, **`application-roles.properties`**, **`mgmt-users.properties`**, and **`mgmt-groups.properties`** files are not referenced using an absolute path, the tool copies them to the path expected by the migrated configuration file.

To skip the security realms migration, set the **`security-realms.migrate-properties.skip environment`** property to **`true`**.

7.3.6. Add the Default SSL Server Identity to the ApplicationRealm

The JBoss EAP 7.1 default configuration includes an SSL server identity for the default **`ApplicationRealm`** security realm. The JBoss Server Migration Tool automatically adds this identity to the migrated configuration files.

To skip the addition of this identity, set the **`security-realm.ApplicationRealm.add-ssl-server-identity.skip`** environment property to **`true`**.

APPENDIX A. REFERENCE MATERIAL

A.1. EXAMPLE TASK SUMMARY REPORT

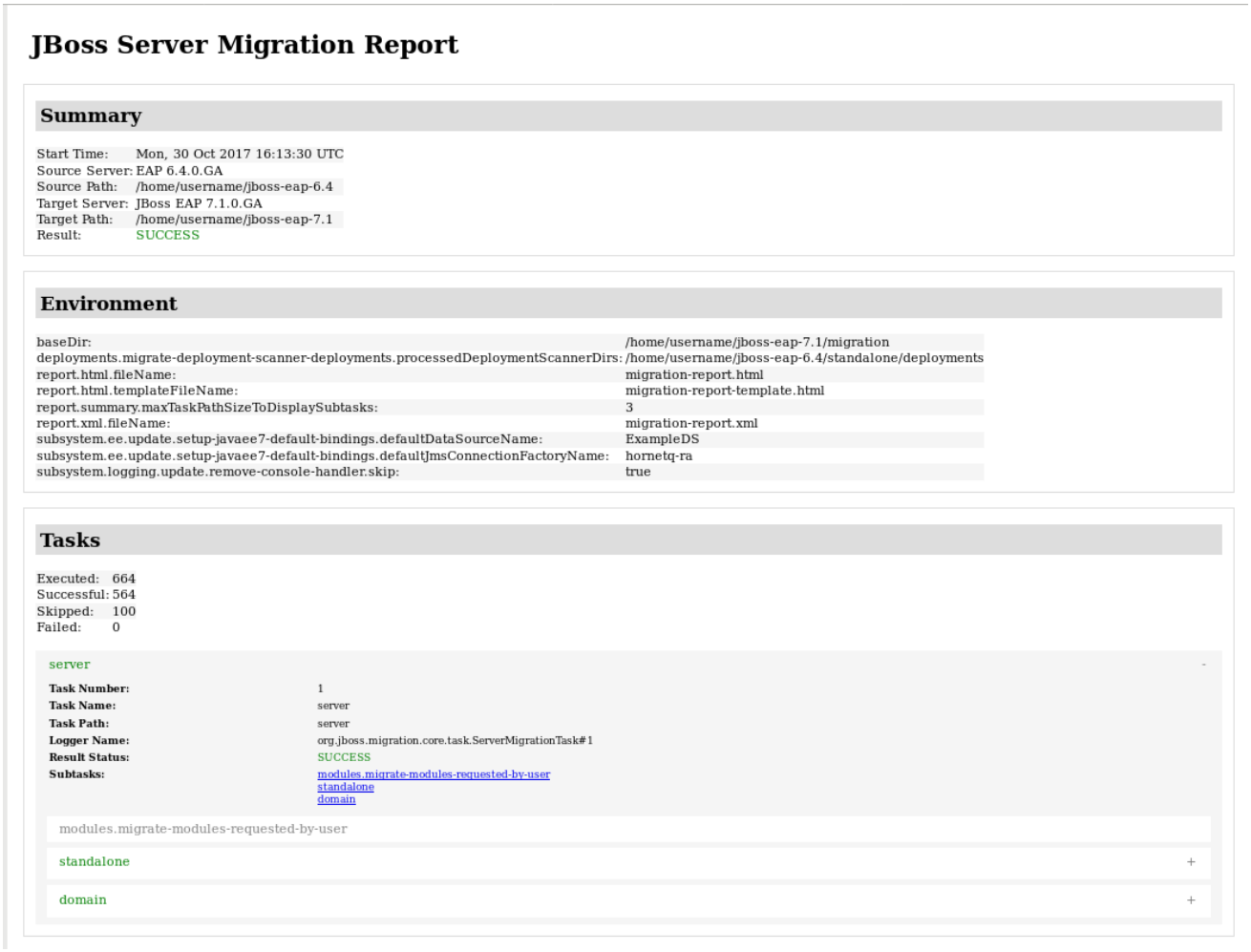
The following is an example of the Task Summary report.

```
-------------
 Task Summary
-------------

 server
.....................................................................
.............................. SUCCESS
  standalone
.....................................................................
.......................... SUCCESS
   standalone-configurations
.....................................................................
............. SUCCESS
    standalone-configuration(source=/home/username/jboss-eap-
6.4/standalone/configuration/standalone-full-ha.xml) .. SUCCESS
    standalone-configuration(source=/home/username/jboss-eap-
6.4/standalone/configuration/standalone-full.xml) ..... SUCCESS
    standalone-configuration(source=/home/username/jboss-eap-
6.4/standalone/configuration/standalone-ha.xml) ....... SUCCESS
    standalone-configuration(source=/home/username/jboss-eap-
6.4/standalone/configuration/standalone-osgi.xml) ..... SUCCESS
    standalone-configuration(source=/home/username/jboss-eap-
6.4/standalone/configuration/standalone.xml) .......... SUCCESS
  domain
.....................................................................
.............................. SUCCESS
   domain-configurations
.....................................................................
................ SUCCESS
    domain-configuration(source=/home/username/jboss-eap-
6.4/domain/configuration/domain.xml) ..................... SUCCESS
   host-configurations
.....................................................................
.................. SUCCESS
    host-configuration(source=/home/username/jboss-eap-
6.4/domain/configuration/host-master.xml) ................. SUCCESS
    host-configuration(source=/home/username/jboss-eap-
6.4/domain/configuration/host-slave.xml) .................. SUCCESS
    host-configuration(source=/home/username/jboss-eap-
6.4/domain/configuration/host.xml) ........................ SUCCESS

-------------------------
 Migration Result: SUCCESS
-------------------------
```

A.2. EXAMPLE HTML REPORT

The following is an example of the JBoss Server Migration HTML report.

Figure A.1. Example: Overview of Sections



JBoss Server Migration Report

Summary

Start Time:	Mon, 30 Oct 2017 16:13:30 UTC
Source Server:	EAP 6.4.0.GA
Source Path:	/home/username/jboss-eap-6.4
Target Server:	JBoss EAP 7.1.0.GA
Target Path:	/home/username/jboss-eap-7.1
Result:	SUCCESS

Environment

baseDir:	/home/username/jboss-eap-7.1/migration
deployments.migrate-deployment-scanner-deployments.processedDeploymentScannerDirs:	/home/username/jboss-eap-6.4/standalone/deployments
report.html.fileName:	migration-report.html
report.html.templateFileName:	migration-report-template.html
report.summary.maxTaskPathSizeToDisplaySubtasks:	3
report.xml.fileName:	migration-report.xml
subsystem.ee.update.setup-javaee7-default-bindings.defaultDataSourceName:	ExampleDS
subsystem.ee.update.setup-javaee7-default-bindings.defaultJmsConnectionFactoryName:	hornetq-ra
subsystem.logging.update.remove-console-handler.skip:	true

Tasks

Executed:	664
Successful:	564
Skipped:	100
Failed:	0

server -

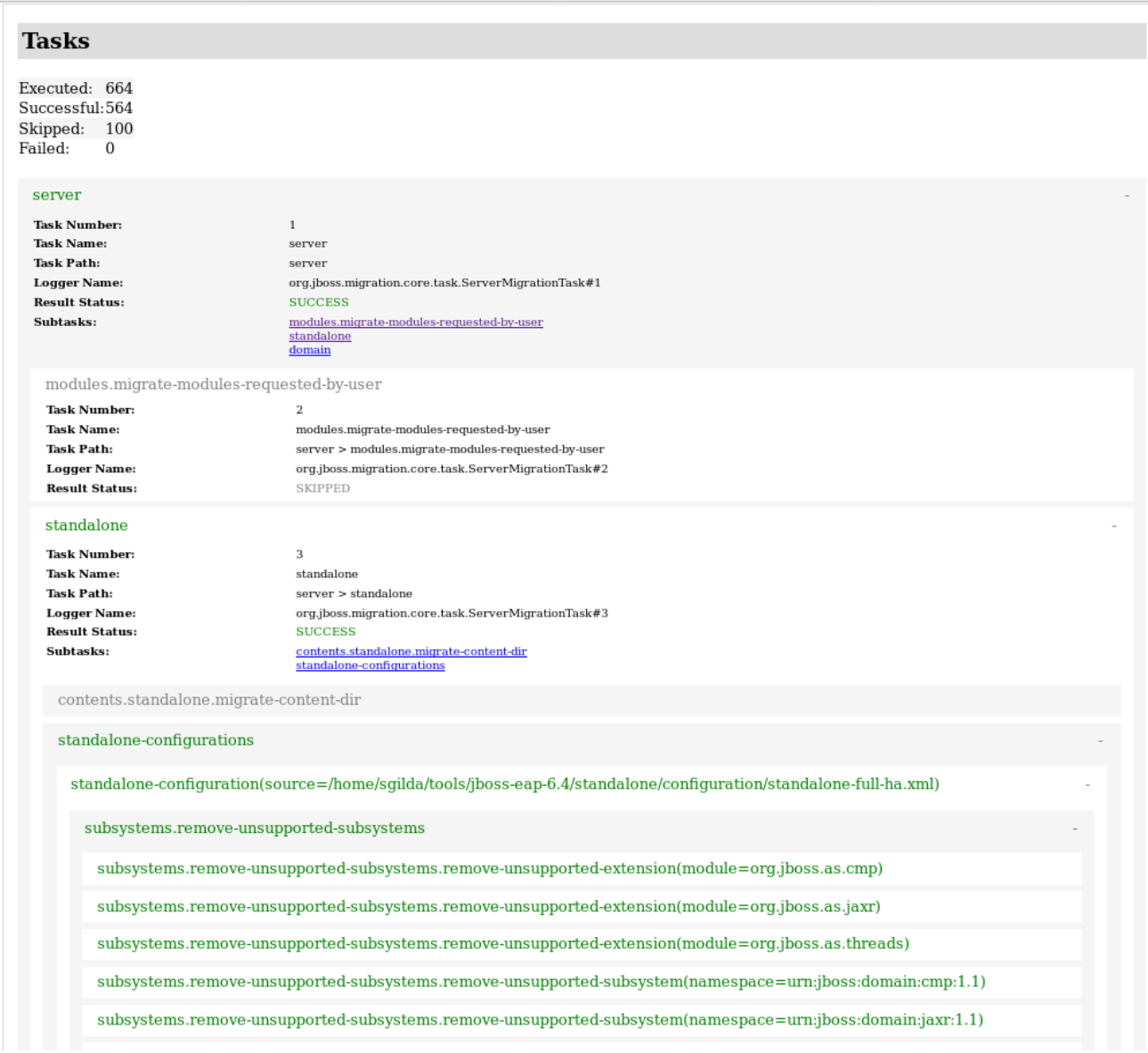
Task Number:	1
Task Name:	server
Task Path:	server
Logger Name:	org.jboss.migration.core.task.ServerMigrationTask#1
Result Status:	SUCCESS
Subtasks:	modules.migrate-modules-requested-by-user standalone domain

modules.migrate-modules-requested-by-user

standalone +

domain +

Figure A.2. Example: Tasks Detail Page



A.3. EXAMPLE XML REPORT

The following is an example of the JBoss Server Migration XML report.

```
<?xml version="1.0" ?>

<server-migration-report xmlns="urn:jboss:server-migration:1.0" start-
time="Mon, 30 Oct 2017 16:13:30 UTC">
  <servers>
    <source name="EAP" version="6.4.0.GA" base-
dir="/home/username/tools/jboss-eap-6.4"/>
    <target name="JBoss EAP" version="7.1.0.GA" base-
dir="/home/username/tools/jboss-eap-7.1"/>
  </servers>
  <environment>
    <property name="baseDir" value="/home/username/tools/jboss-eap-
7.1/migration"/>
    <property name="deployments.migrate-deployment-scanner-
deployments.processedDeploymentScannerDirs"
value="/home/username/tools/jboss-eap-6.4/standalone/deployments"/>
```

```
    <property name="report.html.fileName" value="migration-report.html"/>
    <property name="report.html.maxTaskPathSizeToDisplaySubtasks"
value="4"/>
    <property name="report.html.templateFileName" value="migration-report-
template.html"/>
    <property name="report.summary.maxTaskPathSizeToDisplaySubtasks"
value="3"/>
    <property name="report.xml.fileName" value="migration-report.xml"/>
    <property name="subsystem.ee.update.setup-javaee7-default-
bindings.defaultDataSourceName" value="ExampleDS"/>
    <property name="subsystem.ee.update.setup-javaee7-default-
bindings.defaultJmsConnectionFactoryName" value="hornetq-ra"/>
    <property name="subsystem.logging.update.remove-console-handler.skip"
value="true"/>
  </environment>
  <task number="1" name="server">
    <logger logger="org.jboss.migration.core.task.ServerMigrationTask#1"/>
    <result status="SUCCESS"/>
    <subtasks>
      <task number="2" name="modules.migrate-modules-requested-by-user">
        <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#2"/>
        <result status="SKIPPED"/>
      </task>
      <task number="3" name="standalone">
        <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#3"/>
        <result status="SUCCESS"/>
        <subtasks>
          <task number="4" name="contents.standalone.migrate-content-
dir">
            <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#4"/>
            <result status="SKIPPED"/>
          </task>
          <task number="5" name="standalone-configurations">
            <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#5"/>
            <result status="SUCCESS"/>
            <subtasks>
              <task number="6" name="standalone-
configuration(source=/home/username/tools/jboss-eap-
6.4/standalone/configuration/standalone-full-ha.xml)">
                <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#6"/>
                <result status="SUCCESS"/>
                <subtasks>
                  <task number="7" name="subsystems.remove-unsupported-
subsystems">
                    <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#7"/>
                    <result status="SUCCESS"/>
                    <subtasks>
                      <task number="8" name="subsystems.remove-
unsupported-subsystems.remove-unsupported-
extension(module=org.jboss.as.cmp)">
```

```
                        <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#8"/>
                        <result status="SUCCESS"/>
                      </task>
                    </substasks>
                  </task>
                  ...
                  <task number="644" name="hosts">
                    <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#644"/>
                    <result status="SUCCESS"/>
                    ...
                    <subtasks>
                      ...
                      <task number="645" name="host(name=master)">
                        <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#645"/>
                        <result status="SUCCESS"/>
                        <subtasks>
                          ...
                          <task number="661" name="security-
realms.migrate-properties">
                            <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#661"/>
                            <result status="SUCCESS"/>
                            <subtasks>
                              <task number="662" name="security-
realm.ManagementRealm.migrate-properties">
                                <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#662"/>
                                <result status="SUCCESS"/>
                              </task>
                              <task number="663" name="security-
realm.ApplicationRealm.migrate-properties">
                                <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#663"/>
                                <result status="SUCCESS"/>
                              </task>
                            </subtasks>
                          </task>
                          <task number="664" name="security-
realm.ApplicationRealm.add-ssl-server-identity">
                            <logger
logger="org.jboss.migration.core.task.ServerMigrationTask#664"/>
                            <result status="SUCCESS"/>
                          </task>
                        </subtasks>
                      </task>
                    </subtasks>
                  </task>
                </subtasks>
              </task>
            </subtasks>
          </task>
        </subtasks>
      </task>
```

```
    </subtasks>
  </task>
</server-migration-report>
```

Revised on 2018-08-13 08:36:23 EDT