



Red Hat Enterprise Linux 9

Upgrading from RHEL 8 to RHEL 9

Instructions for an in-place upgrade from Red Hat Enterprise Linux 8 to Red Hat Enterprise Linux 9

Red Hat Enterprise Linux 9 Upgrading from RHEL 8 to RHEL 9

Instructions for an in-place upgrade from Red Hat Enterprise Linux 8 to Red Hat Enterprise Linux 9

Legal Notice

Copyright © 2023 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, the Red Hat 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 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 document provides instructions on how to perform an in-place upgrade from Red Hat Enterprise Linux 8 to Red Hat Enterprise Linux 9 using the Leapp utility. During the in-place upgrade, the existing RHEL 8 operating system is replaced by a RHEL 9 version.

Table of Contents

MAKING OPEN SOURCE MORE INCLUSIVE	3
PROVIDING FEEDBACK ON RED HAT DOCUMENTATION	4
KEY MIGRATION TERMINOLOGY	5
CHAPTER 1. SUPPORTED UPGRADE PATHS	6
CHAPTER 2. PLANNING AN UPGRADE	7
CHAPTER 3. PREPARING FOR THE UPGRADE	10
3.1. PREPARING A RHEL 8 SYSTEM FOR THE UPGRADE	10
3.2. PREPARING A SATELLITE-REGISTERED SYSTEM FOR THE UPGRADE	13
CHAPTER 4. REVIEWING THE PRE-UPGRADE REPORT	16
4.1. ASSESSING UPGRADABILITY FROM THE COMMAND LINE	16
4.2. ASSESSING UPGRADABILITY AND APPLYING AUTOMATED REMEDIATIONS THROUGH THE WEB CONSOLE	18
CHAPTER 5. PERFORMING THE UPGRADE FROM RHEL 8 TO RHEL 9	22
CHAPTER 6. VERIFYING THE POST-UPGRADE STATE OF THE RHEL 9 SYSTEM	24
CHAPTER 7. PERFORMING POST-UPGRADE TASKS	25
CHAPTER 8. APPLYING SECURITY POLICIES	27
8.1. CHANGING SELINUX MODE TO ENFORCING	27
8.2. SYSTEM-WIDE CRYPTOGRAPHIC POLICIES	28
8.3. UPGRADING A SYSTEM HARDENED TO A SECURITY BASELINE	29
8.4. VERIFYING USBGUARD POLICIES	30
8.5. UPDATING FAPOLICYD DATABASES	31
8.6. UPDATING NSS DATABASES FROM DBM TO SQLITE	32
8.7. MIGRATING CYRUS SASL DATABASES FROM THE BERKELEY DB FORMAT TO GDBM	32
CHAPTER 9. TROUBLESHOOTING	34
9.1. TROUBLESHOOTING RESOURCES	34
9.2. TROUBLESHOOTING TIPS	34
9.3. KNOWN ISSUES	36
9.4. OBTAINING SUPPORT	38
CHAPTER 10. RELATED INFORMATION	39
APPENDIX A. RHEL 8 REPOSITORIES	40
APPENDIX B. RHEL 9 REPOSITORIES	42

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see [our CTO Chris Wright's message](#).

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your feedback on our documentation. Let us know how we can improve it.

Submitting comments on specific passages

1. View the documentation in the **Multi-page HTML** format and ensure that you see the **Feedback** button in the upper right corner after the page fully loads.
2. Use your cursor to highlight the part of the text that you want to comment on.
3. Click the **Add Feedback** button that appears near the highlighted text.
4. Add your feedback and click **Submit**.

Submitting feedback through Jira (account required)

1. Log in to the [Jira](#) website.
2. Click **Create** in the top navigation bar
3. Enter a descriptive title in the **Summary** field.
4. Enter your suggestion for improvement in the **Description** field. Include links to the relevant parts of the documentation.
5. Click **Create** at the bottom of the dialogue.

KEY MIGRATION TERMINOLOGY

While the following migration terms are commonly used in the software industry, these definitions are specific to Red Hat Enterprise Linux (RHEL).

Update

Sometimes called a software patch, an update is an addition to the current version of the application, operating system, or software that you are running. A software update addresses any issues or bugs to provide a better experience of working with the technology. In RHEL, an update relates to a minor release, for example, updating from RHEL 8.1 to 8.2.

Upgrade

An upgrade is when you replace the application, operating system, or software that you are currently running with a newer version. Typically, you first back up your data according to instructions from Red Hat. When you upgrade RHEL, you have two options:

- **In-place upgrade:** During an in-place upgrade, you replace the earlier version with the new version without removing the earlier version first. The installed applications and utilities, along with the configurations and preferences, are incorporated into the new version.
- **Clean install:** A clean install removes all traces of the previously installed operating system, system data, configurations, and applications and installs the latest version of the operating system. A clean install is ideal if you do not need any of the previous data or applications on your systems or if you are developing a new project that does not rely on prior builds.

Operating system conversion

A conversion is when you convert your operating system from a different Linux distribution to Red Hat Enterprise Linux. Typically, you first back up your data according to instructions from Red Hat.

Migration

Typically, a migration indicates a change of platform: software or hardware. Moving from Windows to Linux is a migration. Moving a user from one laptop to another or a company from one server to another is a migration. However, most migrations also involve upgrades, and sometimes the terms are used interchangeably.

- **Migration to RHEL:** Conversion of an existing operating system to RHEL
- **Migration across RHEL:** Upgrade from one version of RHEL to another

CHAPTER 1. SUPPORTED UPGRADE PATHS

The in-place upgrade replaces the RHEL 8 operating system on your system with a RHEL 9 version.



IMPORTANT

It is not possible to perform an in-place upgrade directly from RHEL 7 to RHEL 9. However, you can perform an in-place upgrade from RHEL 7 to RHEL 8 and then perform a second in-place upgrade to RHEL 9. For more information, see [Upgrading from RHEL 7 to RHEL 8](#).

Currently, it is possible to perform an in-place upgrade from the following source RHEL 8 minor versions to the following target RHEL 9 minor versions:

Table 1.1. Supported upgrade paths

System configuration	Source OS version	Target OS version
RHEL	RHEL 8.6	RHEL 9.0
	RHEL 8.8	RHEL 9.2
RHEL with SAP HANA	RHEL 8.6	RHEL 9.0
	RHEL 8.8	RHEL 9.2

For more information about supported upgrade paths, see [Supported in-place upgrade paths for Red Hat Enterprise Linux](#).

CHAPTER 2. PLANNING AN UPGRADE

An in-place upgrade is the recommended and supported way to upgrade your system to the next major version of RHEL.

You should consider the following before upgrading to RHEL 9:

- **Operating system** - The operating system is upgradable by the **Leapp** utility under the following conditions:
 - The source OS version is installed on a system with one of the following supported architectures:
 - 64-bit Intel, AMD, and ARM
 - IBM POWER (little endian)
 - 64-bit IBM Z
 For more information, see [Red Hat certified hardware](#) .
 - Minimum [hardware requirements](#) for RHEL 9 are met.
 - You have access to up-to-date content for the selected source and target OS versions. See [Preparing a RHEL 8 system for the upgrade](#) for more information.
- **Applications** - You can migrate applications installed on your system using **Leapp**. However, in certain cases, you have to create custom actors, which specify actions to be performed by **Leapp** during the upgrade, for example, reconfiguring an application or installing a specific hardware driver. For more information, see [Handling the migration of your custom and third-party applications](#). Note that custom actors are unsupported by Red Hat.



IMPORTANT

The SHA-1 algorithm has been deprecated in RHEL 9. If your system contains any packages with RSA/SHA-1 signatures, the upgrade is inhibited. Before upgrading, either remove these packages or contact the vendor for packages with RSA/SHA-256 signatures. For more information, see [SHA-1 deprecation in Red Hat Enterprise Linux 9](#).

- **Security** - You should evaluate this aspect before the upgrade and take additional steps when the upgrade process completes. Consider especially the following:
 - Before the upgrade, define the security standard your system has to comply with and understand the [security changes in RHEL 9](#) .
 - During the upgrade process, the **Leapp** utility sets SELinux mode to permissive.
 - **Leapp** supports in-place upgrades of RHEL 8.8 and later systems in Federal Information Processing Standard (FIPS) 140 mode to RHEL 9 FIPS-mode-enabled systems. **FIPS mode** stays enabled during the complete upgrade process.
 - After the upgrade is finished, re-evaluate and re-apply your security policies. For information about applying and updating security policies, see [Applying security policies](#) .

- **Storage and file systems**– You should always back up your system prior to upgrading. For example, you can use the [Relax-and-Recover \(ReaR\) utility](#), [LVM snapshots](#), [RAID splitting](#), or a virtual machine snapshot.



NOTE

File systems formats are intact. As a consequence, file systems have the same limitations as when they were originally created.

- **High Availability** – If you are using the High Availability add-on, follow the [Recommended Practices for Applying Software Updates to a RHEL High Availability or Resilient Storage Cluster](#) Knowledgebase article.
- **Downtime** – The upgrade process can take from several minutes to several hours.
- **Satellite** – If you manage your hosts through Satellite, you can upgrade multiple hosts simultaneously from RHEL 8 to RHEL 9 using the Satellite web UI. For more information, see [Upgrading Hosts to Next Major Red Hat Enterprise Linux Release](#) .
- **SAP HANA** – If you are using SAP HANA, follow the [How to in-place upgrade SAP environments from RHEL 8 to RHEL 9](#) guide instead. Note that the upgrade path for RHEL with SAP HANA might differ.
- **Public clouds** – The in-place upgrade is supported for on-demand Pay-As-You-Go (PAYG) instances on Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform with [Red Hat Update Infrastructure \(RHUI\)](#) . The in-place upgrade is also supported for Bring Your Own Subscription instances on all public clouds that use RHSM for a RHEL subscription.
- **Language** – All **Leapp** reports, logs, and other generated documentation are in English, regardless of the language configuration.
- **Bootloader** – It is not possible to switch the bootloader from BIOS to UEFI on RHEL 8 or RHEL 9. If your RHEL 8 system uses BIOS and you want your RHEL 9 system to use UEFI, perform a fresh install of RHEL 9 instead of an in-place upgrade. For more information, see [Is it possible to switch the BIOS boot to UEFI boot on preinstalled Red Hat Enterprise Linux machine?](#)
- **Known limitations** – Notable known limitations of **Leapp** currently include:
 - Encryption of the whole disk or a partition, or file-system encryption currently cannot be used on a system targeted for an in-place upgrade.
 - No network-based multipath and no kind of network storage mount can be used as a system partition (for example, iSCSI, or NFS).
 - The in-place upgrade is currently unsupported for on-demand PAYG instances on the remaining Public Clouds (Huawei Cloud, Alibaba Cloud) that use Red Hat Update Infrastructure but not Red Hat Subscription Manager (RHSM) for a RHEL subscription.
 - The in-place upgrade is not supported for systems with any Ansible products, including Ansible Tower, installed. To use a RHEL 8 Ansible Tower installation on RHEL 9, see the [How do I migrate my Ansible Automation Platform installation from one environment to another?](#) Knowledgebase solution.

See also [Known Issues](#).

You can use [Red Hat Insights](#) to determine which of the systems you have registered to Insights is on a supported upgrade path to RHEL 9. To do so, navigate to the respective [Advisor recommendation](#) in

Insights, enable the recommendation under the *Actions* drop-down menu, and inspect the list under the *Affected systems* heading. Note that the Advisor recommendation considers only the RHEL 8 minor version and does not perform a pre-upgrade assessment of the system. See also [Advisor-service recommendations overview](#).

Additional resources

- [The best practices and recommendations for performing RHEL Upgrade using Leapp](#)

CHAPTER 3. PREPARING FOR THE UPGRADE

To prevent issues after the upgrade and to ensure that your system is ready to be upgraded to the next major version of RHEL, complete all necessary preparation steps before upgrading.

You must perform the preparation steps described in [Preparing a RHEL 8 system for the upgrade](#) on all systems. In addition, on systems that are registered to Satellite Server, you must also perform the preparation steps described in [Preparing a Satellite-registered system for the upgrade](#).

3.1. PREPARING A RHEL 8 SYSTEM FOR THE UPGRADE

This procedure describes the steps that are necessary before performing an in-place upgrade to RHEL 9 by using the **Leapp** utility.

If you do not plan to use Red Hat Subscription Manager (RHSM) during the upgrade process, follow instructions in [Upgrading to RHEL 9 without Red Hat Subscription Manager](#).

Prerequisites

- The system meets conditions listed in [Planning an upgrade](#).

Procedure

1. Ensure your system has been successfully registered to the Red Hat Content Delivery Network (CDN) or Red Hat Satellite by using the Red Hat Subscription Manager.
2. If you have registered your system to Satellite Server, complete the steps in [Preparing a Satellite-registered system for the upgrade](#) to ensure that your system meets the requirements for the upgrade.
3. Verify that the system is subscribed using subscription-manager:
 - a. If your system is registered by using an account with [Simple Content Access](#) (SCA) enabled, verify that the **Content Access Mode is set to Simple Content Access** message appears:

```
# subscription-manager status
+-----+
  System Status Details
+-----+
Overall Status: Disabled
Content Access Mode is set to Simple Content Access. This host has access to content,
regardless of subscription status.
System Purpose Status: Disabled
```

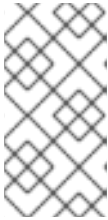
- b. If your system is registered by using an account with SCA disabled, verify that the Red Hat Linux Server subscription is attached, the product name is **Server**, and the status is **Subscribed**. For example:

```
# subscription-manager list --installed
+-----+
  Installed Product Status
+-----+
Product Name:  Red Hat Enterprise Linux for x86_64
Product ID:    479
```

```
Version:    8.6
Arch:      x86_64
Status:    Subscribed
```

4. Ensure you have appropriate repositories enabled. The following command enables the Base and AppStream repositories for the 64-bit Intel architecture; for other architectures, see [RHEL 8 repositories](#).

```
# subscription-manager repos --enable rhel-8-for-x86_64-baseos-rpms --enable rhel-8-for-x86_64-appstream-rpms
```



NOTE

Optionally, you can enable the CodeReady Linux Builder (also known as Optional) or Supplementary repositories. For more information about repository IDs, see [RHEL 8 repositories](#). For more information about the content of these repositories, see the [Package manifest](#).

5. Set the system release version:
 - a. For systems subscribed using RHSM, lock the system to the desired source OS version:

```
# subscription-manager release --set <source_os_version>
```

- b. If you are upgrading by using Red Hat Update Infrastructure (RHUI) on a public cloud, set the expected system release version manually:

```
# rhui-release-set --set <source_os_version>
```



IMPORTANT

If the **rhui-release-set** command is not available on your system, you can set the expected system release version by updating the **/etc/dnf/vars/release** file:

```
# echo "<source_os_version>" > /etc/dnf/vars/releasever
```

Replace *source_os_version* with the source OS version, for example **8.8**.

6. Optional: To use custom repositories, see the [Configuring custom repositories](#) Knowledgebase article.
7. If you use the **dnf versionlock** plugin to lock packages to a specific version, clear the lock by running:

```
# dnf versionlock clear
```

See [How to restrict dnf to install or upgrade a package to a fixed specific package version?](#) for more information.

8. If you are upgrading by using Red Hat Update Infrastructure (RHUI) on a public cloud, enable required RHUI repositories and install required RHUI packages to ensure your system is ready for upgrade:

- a. For AWS:

```
# dnf config-manager --set-enabled rhui-client-config-server-8
# dnf -y install leapp-rhui-aws
```

- b. For Microsoft Azure:

```
# dnf config-manager --set-enabled rhui-microsoft-azure-rhel8
# dnf -y install rhui-azure-rhel8 leapp-rhui-azure
```

- c. For Google Cloud Platform, follow the [Leapp RHUI packages for Google Cloud Platform \(GCP\)](#) Knowledgebase article.

9. Update all packages to the latest RHEL 8 version:

```
# dnf update
```

10. Reboot the system:

```
# reboot
```

11. Install the **Leapp** utility:

```
# dnf install leapp-upgrade
```

Note that currently you need version 0.15.1 or later of the **leapp** package and version 0.18.0 or later of the **leapp-repository** package, which contains the **leapp-upgrade-el8toel9** RPM package.



NOTE

If your system does not have internet access, download the following packages from the [Red Hat Customer Portal](#):

- **leapp**
- **leapp-deps**
- **python3-leapp**
- **leapp-upgrade-el8toel9**
- **leapp-upgrade-el8toel9-deps**

12. The latest release of the **leapp-upgrade-el8toel9** package contains all required data files. If you have replaced these data files with older versions, remove all JSON files in the **/etc/leapp/files** directory and reinstall the **leapp-upgrade-el8toel9** package to ensure your data files are up-to-date.
13. Temporarily disable antivirus software to prevent the upgrade from failing.

14. Ensure that any configuration management system does not interfere with the in-place upgrade process:
 - If you use a configuration management system with a client-server architecture, such as **Puppet**, **Salt**, or **Chef**, disable the system before running the **leapp preupgrade** command. Do not enable the configuration management system until after the upgrade is complete to prevent issues during the upgrade.
 - If you use a configuration management system with agentless architecture, such as **Ansible**, do not execute the configuration and deployment file, such as an Ansible playbook, during the in-place upgrade as described in [Performing the upgrade from RHEL 8 to RHEL 9](#) . Automation of the pre-upgrade and upgrade process by using a configuration management system is not supported by Red Hat. For more information, see [Using configuration management systems to automate parts of the Leapp pre-upgrade and upgrade process on Red Hat Enterprise Linux](#).
15. Ensure your system does not use more than one Network Interface Card (NIC) with a name based on the prefix used by the kernel (**eth**). For instructions on how to migrate to another naming scheme before an in-place upgrade to RHEL 9, see [How to perform an in-place upgrade to RHEL 8 when using kernel NIC names on RHEL 7](#). The process for migrating naming schemes is the same for both the RHEL 7 to RHEL 8 upgrade and the RHEL 8 to RHEL 9 upgrade.
16. If your NSS database was created in RHEL 7 or earlier, verify that the database has been converted from the DBM database format to SQLite. For more information, see [Updating NSS databases from DBM to SQLite](#).
17. RHEL 9 does not support the legacy **network-scripts** package, which was deprecated in RHEL 8. Before upgrading, move your custom network scripts and write a NetworkManager dispatcher script that executes your existing custom scripts. For more information, see [Migrating custom network scripts to NetworkManager dispatcher scripts](#).
18. If you are upgrading using an ISO image, verify that the ISO image contains the target OS version, for example, RHEL 9.0, and is saved to a persistent local mount point to ensure that the **Leapp** utility can access the image throughout the upgrade process.
19. Ensure you have a full system backup or a virtual machine snapshot. You should be able to get your system to the pre-upgrade state if you follow standard disaster recovery procedures within your environment. For example, you can use the Relax-and-Recover (ReaR) utility. For more information, see the [ReaR documentation](#) and [What is Relax and Recover \(ReaR\) and how can I use it for disaster recovery?](#). Alternatively, you can use [LVM snapshots](#), or [RAID splitting](#). In case of upgrading a virtual machine, you can create a snapshot of the whole VM.

3.2. PREPARING A SATELLITE-REGISTERED SYSTEM FOR THE UPGRADE

This procedure describes the steps that are necessary to prepare a system that is registered to Satellite for the upgrade to RHEL 9. These steps are performed on the Satellite Server.



IMPORTANT

Users on Satellite systems must complete the preparatory steps described both in this procedure and in [Preparing a RHEL 8 system for the upgrade](#) .

Prerequisites

- You have administrative privileges for the Satellite Server.

Procedure

1. Verify that Satellite is on a version in full or maintenance support. For more information, see [Red Hat Satellite Product Life Cycle](#).
2. Import a subscription manifest with RHEL 9 repositories into Satellite Server. For more information, see the Managing Red Hat Subscriptions chapter in the Managing Content Guide for the particular version of [Red Hat Satellite](#), for example, for [version 6.12](#).
3. Enable and synchronize all required RHEL 8 and RHEL 9 repositories on the Satellite Server with the latest updates for the source and target OS versions. Required repositories must be available in the Content View and enabled in the associated activation key.



NOTE

For RHEL 9 repositories, enable the target OS version, for example, RHEL 9.0, of each repository. If you enable only the RHEL 9 version of the repositories, the in-place upgrade is inhibited.

For example, for the Intel architecture without an Extended Update Support (EUS) subscription, enable at minimum the following repositories:

- Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMs)
rhel-8-for-x86_64-appstream-rpms

x86_64 8 or `<source_os_version>`

- Red Hat Enterprise Linux 8 for x86_64 - BaseOS (RPMs)
rhel-8-for-x86_64-baseos-rpms

x86_64 8 or `<source_os_version>`

- Red Hat Enterprise Linux 9 for x86_64 - AppStream (RPMs)
rhel-9-for-x86_64-appstream-rpms

x86_64 `<target_os_version>`

- Red Hat Enterprise Linux 9 for x86_64 - BaseOS (RPMs)
rhel-9-for-x86_64-baseos-rpms

x86_64 `<target_os_version>`

Replace `<source_os_version>` and `<target_os_version>` with the source OS version and target OS version respectively, for example, 8.6 and 9.0.

For other architectures, see [RHEL 8 repositories](#) and [RHEL 9 repositories](#).

For more information, see the *Importing Content* chapter in the *Managing Content Guide* for the particular version of [Red Hat Satellite](#), for example, for [version 6.12](#).

4. Attach the content host to a Content View containing the required RHEL 8 and RHEL 9 repositories.

For more information, see the *Managing Content Views* chapter in the *Managing Content Guide* for the particular version of [Red Hat Satellite](#), for example, for [version 6.12](#).

Verification

1. Verify that the correct RHEL 8 and RHEL 9 repositories have been added to the correct Content View on Satellite Server.
 - a. In the Satellite web UI, navigate to **Content > Lifecycle > Content Views** and click the name of the Content View.
 - b. Click the **Repositories** tab and verify that the repositories appear as expected.



NOTE

You can also verify that the repositories have been added to the Content View using the following commands:

```
# hammer repository list --search 'content_label ~ rhel-8' --content-view
<content_view_name> --organization <organization> --lifecycle-
environment <lifecycle_environment>
# hammer repository list --search 'content_label ~ rhel-9' --content-view
<content_view_name> --organization <organization> --lifecycle-
environment <lifecycle_environment>
```

Replace `<content_view_name>` with the name of the Content View, `<organization>` with the organization, and `<lifecycle_environment>` with the name of the lifecycle environment..

2. Verify that the correct RHEL 9 repositories are enabled in the activation key associated with the Content View:
 - a. In Satellite web UI navigate to **Content > Lifecycle > Activation Keys** and click the name of the activation key.
 - b. Click the **Repository Sets** tab and verify that the statuses of the required repositories are **Enabled**.
3. . Verify that all expected RHEL 8 repositories are enabled in the host. For example:

```
# subscription-manager repos --list-enabled | grep "^Repo ID"
Repo ID: rhel-8-for-x86_64-baseos-rpms
Repo ID: rhel-8-for-x86_64-appstream-rpms
```

CHAPTER 4. REVIEWING THE PRE-UPGRADE REPORT

To assess upgradability of your system, start the pre-upgrade process by using the **leapp preupgrade** command. During this phase, the **Leapp** utility collects data about the system, assesses upgradability, and generates a pre-upgrade report. The pre-upgrade report summarizes potential problems and suggests recommended solutions. The report also helps you decide whether it is possible or advisable to proceed with the upgrade.



IMPORTANT

Always review the entire pre-upgrade report, even when the report finds no inhibitors to the upgrade. The pre-upgrade report contains recommended actions to complete before the upgrade to ensure that the upgraded system functions correctly.

Reviewing a pre-upgrade report can also be useful if you want to perform a fresh installation of a RHEL 9 system instead of the in-place upgrade process.

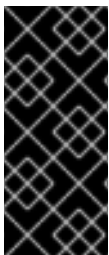
You can assess upgradability in the pre-upgrade phase using either of the following ways:

- Review the pre-upgrade report in the generated **leapp-report.txt** file and manually resolve reported problems using the command-line interface.
- Use the web console to review the report, apply automated remediations where available, and fix remaining problems using the suggested remediation hints.



NOTE

You can process the pre-upgrade report by using your own custom scripts, for example, to compare results from multiple reports across different environments. For more information, see [Automating your Red Hat Enterprise Linux pre-upgrade report workflow](#).



IMPORTANT

The pre-upgrade report cannot simulate the entire in-place upgrade process and therefore cannot identify all inhibiting problems with your system. As a result, your in-place upgrade might still be terminated even after you have reviewed and remediated all problems in the report. For example, the pre-upgrade report cannot detect issues related to broken package downloads.

4.1. ASSESSING UPGRADABILITY FROM THE COMMAND LINE

Identify potential upgrade problems during the pre-upgrade phase by using the command-line interface.

Prerequisites

- The steps listed in [Preparing for the upgrade](#) have been completed.

Procedure

1. On your RHEL 8 system, perform the pre-upgrade phase:

```
# leapp preupgrade
```

- If you are using [custom repositories](#) from the `/etc/yum.repos.d/` directory for the upgrade, enable the selected repositories as follows:

```
# leapp preupgrade --enablerepo <repository_id1> --enablerepo <repository_id2> ...
```

- If you are [upgrading without RHSM](#) or using RHUI, add the `--no-rhsm` option.
 - If you have an [Extended Upgrade Support \(EUS\)](#), [Advanced Update Support \(AUS\)](#), or [Update Services for SAP Solutions \(E4S\)](#) subscription, add the `--channel <channel>` option. Replace `<channel>` with the channel name, for example, `eus`, `aus`, or `e4s`. Note that SAP HANA customers should perform the in-place upgrade using the [How to in-place upgrade SAP environments from RHEL 8 to RHEL 9](#) guide.
2. Examine the report in the `/var/log/leapp/leapp-report.txt` file and manually resolve all the reported problems. Some reported problems contain remediation suggestions. **Inhibitor** problems prevent you from upgrading until you have resolved them. The report contains the following risk factor levels:

High

Very likely to result in a deteriorated system state.

Medium

Can impact both the system and applications.

Low

Should not impact the system but can have an impact on applications.

Info

Informational with no expected impact to the system or applications.

3. In certain system configurations, the **Leapp** utility generates true or false questions that you must answer manually. If the pre-upgrade report contains a **Missing required answers in the answer file** message, complete the following steps:
 - a. Open the `/var/log/leapp/answerfile` file and review the true or false questions.
 - b. Manually edit the `/var/log/leapp/answerfile` file, uncomment the confirm line of the file by deleting the `#` symbol, and confirm your answer as **True** or **False**. For more information, see the [Leapp answerfile](#).



NOTE

Alternatively, you can answer the true or false question by running the following command:

```
# leapp answer --section <question_section>.<field_name>=<answer>
```

For example, to confirm a **True** response to the question **Are all VDO devices, if any, successfully converted to LVM management?**, execute the following command:

```
# leapp answer --section check_vdo.confirm=True
```

4. Repeat the previous steps to rerun the pre-upgrade report to verify that you have resolved all critical issues.

4.2. ASSESSING UPGRADABILITY AND APPLYING AUTOMATED REMEDIATIONS THROUGH THE WEB CONSOLE

Identify potential problems in the pre-upgrade phase and apply automated remediations by using the web console.

Prerequisites

- You have completed the steps listed in [Preparing for the upgrade](#).

Procedure

1. Install the **cockpit-leapp** plug-in:

```
# dnf install cockpit-leapp
```

2. Log in to the web console as **root** or as a user that has permissions to enter administrative commands with **sudo**. See [Managing systems using the RHEL 8 web console](#) for more information about the web console.
3. On your RHEL 8 system, perform the pre-upgrade phase either from the command-line interface or from the web console terminal:

```
# leapp preupgrade
```

- If you are using [custom repositories](#) from the `/etc/yum.repos.d/` directory for the upgrade, enable the selected repositories as follows:

```
# leapp preupgrade --enablerepo <repository_id1> --enablerepo <repository_id2> ...
```

- If you are [upgrading without RHSM](#) or using RHUI, add the **--no-rhsm** option.
 - If you have an [Extended Upgrade Support \(EUS\)](#), Advanced Update Support (AUS), or [Update Services for SAP Solutions \(E4S\)](#) subscription, add the **--channel <channel>** option. Replace `<channel>` with the channel name, for example, **eus**, **aus**, or **e4s**. Note that SAP HANA customers should perform the in-place upgrade using the [How to in-place upgrade SAP environments from RHEL 8 to RHEL 9](#) guide.
4. In the web console, select **Upgrade Report** from the navigation menu to review all reported problems. **Inhibitor** problems prevent you from upgrading until you have resolved them. To view a problem in detail, select the row to open the Detail pane.

Figure 4.1. In-place upgrade report in the web console

Upgrade Report for: localhost.localdomain				
Filters		Remediation plan (0) + Add all remediations to plan (2)		
Title	Risk Factor	Description	Tags	Time
Packages available in excluded repositories will not be installed	High		repository	20.04.2023 12:27:53
Packages not signed by Red Hat found on the system	High		sanity	20.04.2023 12:27:54
Upgrade is unsupported	High		upgrade process sanity	20.04.2023 12:27:54
Leapp detected a processor which is no longer maintained in RHEL 9.	High		kernel boot	20.04.2023 12:27:56
Firewalld Configuration AllowZoneDrifting Is Unsupported	High	Inhibitor Remediation hint Remediation command Links	sanity firewall	20.04.2023 12:27:56
GRUB core will be updated during upgrade	High		boot	20.04.2023 12:27:56
Remote root logins globally allowed using password	High	Remediation hint	authentication security network services	20.04.2023 12:27:58
PostgreSQL (postgresql-server) has been detected on your system	Medium	Remediation hint Links	services	20.04.2023 12:27:55
Detected broken systemd symlinks for existing services	Medium	Remediation hint	filesystem	20.04.2023 12:27:55
Detected broken systemd symlinks for non-existing services	Low	Remediation hint Remediation command	filesystem	20.04.2023 12:27:55

10 per page 1-10 of 14 1 of 2

The report contains the following risk factor levels:

High

Very likely to result in a deteriorated system state.

Medium

Can impact both the system and applications.

Low

Should not impact the system but can have an impact on applications.

Info

Informational with no expected impact to the system or applications.

5. In certain configurations, the **Leapp** utility generates true or false questions that you must answer manually. If the Upgrade Report contains a **Missing required answers in the answer file** row, complete the following steps:
 - a. Select the **Missing required answers in the answer file** row to open the **Detail** pane. The default answer is stated at the end of the remediation command.
 - b. To confirm the default answer, select **Add to Remediation Plan** to execute the remediation later or **Run Remediation** to execute the remediation immediately.
 - c. To select the non-default answer instead, execute the **leapp answer** command in the terminal, specifying the question you are responding to and your confirmed answer.

```
# leapp answer --section <question_section>.<field_name>=<answer>
```

For example, to confirm a **True** response to the question **Are all VDO devices, if any, successfully converted to LVM management?**, execute the following command:

```
# leapp answer --section check_vdo.confirm=True
```

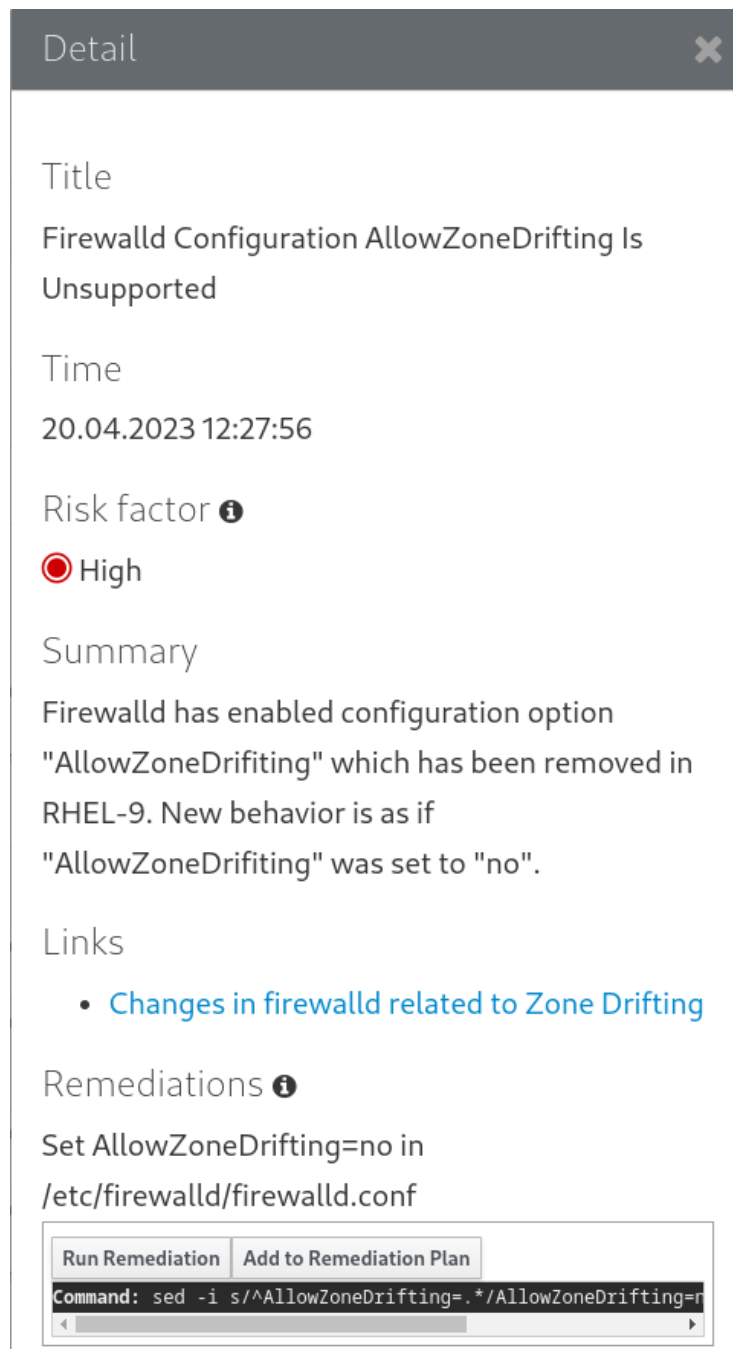


NOTE

You can also manually edit the `/var/log/leapp/answerfile` file, uncomment the confirm line of the file by deleting the `#` symbol, and confirm your answer as **True** or **False**. For more information, see the [Leapp answerfile example](#).

6. Some problems have remediation commands that you can run to automatically resolve the problems. You can run remediation commands individually or all together in the remediation command.
 - a. To run a single remediation command, open the **Detail** pane for the problem and click **Run Remediation**.
 - b. To add a remediation command to the remediation plan, open the **Detail** pane for the problem and click **Add to Remediation Plan**

Figure 4.2. Detail pane



- c. To run the remediation plan containing all added remediation commands, click the **Remediation plan** link in the top right corner above the report. Click **Execute Remediation Plan** to execute all listed commands.
7. After reviewing the report and resolving all reported problems, repeat steps 3-7 to rerun the report to verify that you have resolved all critical issues.

CHAPTER 5. PERFORMING THE UPGRADE FROM RHEL 8 TO RHEL 9

This procedure lists steps required to perform the upgrade from RHEL 8 to RHEL 9 using the **Leapp** utility.

Prerequisites

- The steps listed in [Preparing for the upgrade](#) have been completed, including a full system backup.
- The steps listed in [Reviewing the pre-upgrade report](#) have been completed and all reported issues resolved.

Procedure

1. On your RHEL 8 system, start the upgrade process:

```
# leapp upgrade
```

- If you are using [custom repositories](#) from the `/etc/yum.repos.d/` directory for the upgrade, enable the selected repositories as follows:

```
# leapp upgrade --enablerepo <repository_id1> --enablerepo <repository_id2> ...
```

- If you are [upgrading without RHSM](#) or using RHUI, add the `--no-rhsm` option.
 - If you are upgrading by using an ISO image, add the `--no-rhsm` and `--iso <file_path>` options. Replace `<file_path>` with the file path to the saved ISO image, for example `/home/rhel9.iso`.
 - If you have an [Extended Upgrade Support \(EUS\)](#), [Advanced Update Support \(AUS\)](#), or [Update Services for SAP Solutions \(E4S\)](#) subscription, add the `--channel channel` option. Replace `channel` with the value you used with the `leapp preupgrade` command, for example, `eus`, `aus`, or `e4s`. Note that you must use the same value with the `--channel` option in both the `leapp preupgrade` and `leapp upgrade` commands.
2. At the beginning of the upgrade process, **Leapp** performs the pre-upgrade phase described in [Reviewing the pre-upgrade report](#).
 - If the system is upgradable, **Leapp** downloads necessary data and prepares an RPM transaction for the upgrade.
 - If your system does not meet the parameters for a reliable upgrade, **Leapp** terminates the upgrade process and provides a record describing the issue and a recommended solution in the `/var/log/leapp/leapp-report.txt` file. For more information, see [Troubleshooting](#).
 3. Manually reboot the system:

```
# reboot
```

In this phase, the system boots into a RHEL 9-based initial RAM disk image, `initramfs`. **Leapp** upgrades all packages and automatically reboots to the RHEL 9 system.

Alternatively, you can run the **leapp upgrade** command with the **--reboot** option and skip this manual step.

If a failure occurs, investigate logs and known issues as described in [Troubleshooting](#).

4. Log in to the RHEL 9 system and verify its state as described in [Verifying the post-upgrade state of the RHEL 9 system](#).
5. Perform all post-upgrade tasks described in the upgrade report and in [Performing post-upgrade tasks](#).

CHAPTER 6. VERIFYING THE POST-UPGRADE STATE OF THE RHEL 9 SYSTEM

This procedure lists verification steps recommended to perform after an in-place upgrade to RHEL 9.

Prerequisites

- The system has been upgraded following the steps described in [Performing the upgrade from RHEL 8 to RHEL 9](#) and you have been able to log in to RHEL 9.

Procedure

After the upgrade completes, determine whether the system is in the required state, at least:

- Verify that the current OS version is RHEL 9. For example:

```
# cat /etc/redhat-release
Red Hat Enterprise Linux release 9.0 (Plow)
```

- Check the OS kernel version. For example:

```
# uname -r
5.14.0-70.10.1.el9_0.x86_64
```

Note that **.el9** is important and the version should not be earlier than 5.14.0.

- If you are using the Red Hat Subscription Manager:
 - Verify that the correct product is installed. For example:

```
# subscription-manager list --installed
+-----+
      Installed Product Status
+-----+
Product Name: Red Hat Enterprise Linux for x86_64
Product ID:  479
Version:     9.0
Arch:        x86_64
Status:      Subscribed
```

- Verify that the release version is set to the expected target OS version immediately after the upgrade. For example:

```
# subscription-manager release
Release: 9.0
```

- Verify that network services are operational, for example, try to connect to a server using SSH.
- Check the post-upgrade status of your applications. In some cases, you may need to perform migration and configuration changes manually. For example, to migrate your databases, follow instructions in [Configuring and using database servers](#).

CHAPTER 7. PERFORMING POST-UPGRADE TASKS

This procedure lists major tasks recommended to perform after an in-place upgrade to RHEL 9.

Prerequisites

- The system has been upgraded following the steps described in [Performing the upgrade from RHEL 8 to RHEL 9](#) and you have been able to log in to RHEL 9.
- The status of the in-place upgrade has been verified following the steps described in [Verifying the post-upgrade status of the RHEL 9 system](#).

Procedure

After performing the upgrade, complete the following tasks:

1. Remove any remaining **Leapp** packages from the exclude list in the `/etc/dnf/dnf.conf` configuration file, including the **snactor** package, which is a tool for upgrade extension development. During the in-place upgrade, **Leapp** packages that were installed with the **Leapp** utility are automatically added to the exclude list to prevent critical files from being removed or updated. After the in-place upgrade, these **Leapp** packages must be removed from the exclude list before they can be removed from the system.
 - To manually remove packages from the exclude list, edit the `/etc/dnf/dnf.conf` configuration file and remove the desired **Leapp** packages from the exclude list.
 - To remove all packages from the exclude list:

```
# dnf config-manager --save --setopt exclude=""
```

2. Remove remaining RHEL 8 packages, including remaining **Leapp** packages.

- a. Locate remaining RHEL 8 packages:

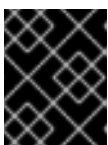
```
# rpm -qa | grep -e '\.el[78]' | grep -vE '(gpg-pubkey|libmodulemd|katello-ca-consumer)' | sort
```

- b. Remove remaining RHEL 8 packages, including the old kernel package, from your RHEL 9 system.
- c. Remove remaining **Leapp** dependency packages:

```
# dnf remove leapp-deps-el9 leapp-repository-deps-el9
```

3. Optional: Remove all remaining upgrade-related data from the system:

```
# rm -rf /var/log/leapp /root/tmp_leapp_py3 /var/lib/leapp
```



IMPORTANT

Removing this data might limit Red Hat Support's ability to investigate and troubleshoot post-upgrade problems.

4. Disable DNF repositories whose packages are not RHEL 9-compatible. Repositories managed by RHSM are handled automatically. To disable these repositories:

```
# dnf config-manager --set-disabled <repository_id>
```

Replace *repository_id* with the repository ID.

5. Replace the old rescue kernel and initial RAM disk with the current kernel and disk:

- a. Remove the existing rescue kernel and initial RAM disk:

```
# rm /boot/vmlinuz-*rescue* /boot/initramfs-*rescue*
```

- b. Reinstall the current kernel to recover the rescue kernel and related initial RAM disk:

```
# dnf reinstall -y kernel-core-$(uname -r)
```



NOTE

If your system's kernel package has a different name, such as on real-time systems, replace **kernel-core** with the correct package name.

6. Re-evaluate and re-apply your security policies. Especially, change the SELinux mode to enforcing. For details, see [Applying security policies](#).

Verification

1. Verify that the previously removed rescue kernel and rescue initial RAM disk files have been created for the current kernel:

```
# ls /boot/vmlinuz-*rescue* /boot/initramfs-*rescue*
# lsinitrd /boot/initramfs-*rescue*.img | grep -qm1 "$(uname -r)/kernel/" && echo "OK" || echo "FAIL"
```

2. Verify the rescue boot entry refers to the existing rescue files. See the grubby output:

```
# grubby --info $(ls /boot/vmlinuz-*rescue*)
```

CHAPTER 8. APPLYING SECURITY POLICIES

During the in-place upgrade process, the SELinux policy must be switched to permissive mode. Furthermore, security profiles might contain changes between major releases. This section guides you when securing your upgraded RHEL systems and covers details for pre-upgrade steps of security-related components.

8.1. CHANGING SELINUX MODE TO ENFORCING

During the in-place upgrade process, the **Leapp** utility sets SELinux mode to permissive. When the system is successfully upgraded, you have to manually change SELinux mode to enforcing.

Prerequisites

- The system has been upgraded and you have performed the verification steps described in [Verifying the post-upgrade state of the RHEL 9 system](#).

Procedure

1. Ensure that there are no SELinux denials, for example, by using the **ausearch** utility:

```
# ausearch -m AVC,USER_AVC -ts boot
```

Note that the previous step covers only the most common scenario. To check for all possible SELinux denials, see the [Identifying SELinux denials](#) section in the Using SELinux title, which provides a complete procedure.

2. Open the **/etc/selinux/config** file in a text editor of your choice, for example:

```
# vi /etc/selinux/config
```

3. Configure the **SELINUX=enforcing** option:

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=enforcing
# SELINUXTYPE= can take one of these two values:
#   targeted - Targeted processes are protected,
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

4. Save the change, and restart the system:

```
# reboot
```

Verification

1. After the system restarts, confirm that the **getenforce** command returns **Enforcing**:

```
$ getenforce
Enforcing
```

Additional resources

- [Troubleshooting problems related to SELinux](#)
- [Changing SELinux states and modes](#)

8.2. SYSTEM-WIDE CRYPTOGRAPHIC POLICIES

The system-wide cryptographic policies is a system component that configures the core cryptographic subsystems, covering the TLS, IPsec, SSH, DNSSec, and Kerberos protocols.

The in-place upgrade process preserves the cryptographic policy you used in RHEL 8. For example, if you used the **DEFAULT** cryptographic policy in RHEL 8, your system upgraded to RHEL 9 also uses **DEFAULT**. Note that specific settings in predefined policies differ, and RHEL 9 cryptographic policies contain more strict and more secure default values. For example, the RHEL 9 **DEFAULT** cryptographic policy restricts SHA-1 usage for signatures and the **LEGACY** policy no longer allows DH and RSA ciphers with less than 2048 bits. See the [Using system-wide cryptographic policies](#) section in the [Security hardening](#) document for more information. Custom cryptographic policies are preserved across the in-place upgrade.

To view or change the current system-wide cryptographic policy, use the update-crypto-policies tool:

```
$ update-crypto-policies --show
DEFAULT
```

For example, the following command switches the system-wide crypto policy level to **FUTURE**, which should withstand any near-term future attacks:

```
# update-crypto-policies --set FUTURE
Setting system policy to FUTURE
```

If your scenario requires the use of SHA-1 for verifying existing or third-party cryptographic signatures, you can enable it by entering the following command:

```
# update-crypto-policies --set DEFAULT:SHA1
```

Alternatively, you can switch the system-wide crypto policies to the **LEGACY** policy. However, **LEGACY** also enables many other algorithms that are not secure.



WARNING

Enabling the **SHA** subpolicy makes your system more vulnerable than the default RHEL 9 settings. Switching to the **LEGACY** policy is even less secure, and you should use it with caution.

You can also customize system-wide cryptographic policies. For details, see the [Customizing system-wide cryptographic policies with subpolicies](#) and [Creating and setting a custom system-wide cryptographic policy](#) sections. If you use a custom cryptographic policy, consider reviewing and updating the policy to mitigate threats brought by advances in cryptography and computer hardware.

Additional resources

- [Using system-wide cryptographic policies](#)
- `update-crypto-policies(8)` man page

8.3. UPGRADING A SYSTEM HARDENED TO A SECURITY BASELINE

To get a fully hardened system after a successful upgrade to RHEL 9, you can use automated remediation provided by the OpenSCAP suite. OpenSCAP remediations align your system with security baselines, such as PCI-DSS, OSPP, or ACSC Essential Eight. The configuration compliance recommendations differ among major versions of RHEL due to the evolution of the security offering.

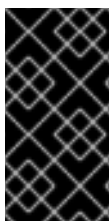
When upgrading a hardened RHEL 8 system, the **Leapp** tool does *not* provide direct means to retain the full hardening. Depending on the changes in the component configuration, the system might diverge from the recommendations for the RHEL 9 during the upgrade.



NOTE

You cannot use the same SCAP content for scanning RHEL 8 and RHEL 9. Update the management platforms if the compliance of the system is managed by tools such as Red Hat Satellite or Red Hat Insights.

As an alternative to automated remediations, you can make the changes manually by following an OpenSCAP-generated report. For information about generating a compliance report, see [Scanning the system for security compliance and vulnerabilities](#).



IMPORTANT

Automated remediations support RHEL systems in the default configuration. Because the system configuration has been altered after the upgrade, running automated remediations might not make the system fully compliant with the required security profile. You might need to fix some requirements manually.

The following example procedure hardens your system settings according to the PCI-DSS profile.

Prerequisites

- The **scap-security-guide** package is installed on your RHEL 9 system.

Procedure

1. Find the appropriate security compliance data stream **.xml** file:

```
$ ls /usr/share/xml/scap/ssg/content/
...
ssg-rhel9-ds.xml
...
```

See the [Viewing compliance profiles](#) section for more information.

2. Remediate the system according to the selected profile from the appropriate data stream:

```
# oscap xccdf eval --profile pci-dss --remediate /usr/share/xml/scap/ssg/content/ssg-rhel9-ds.xml
```

You can replace the **pci-dss** value in the **--profile** argument with the ID of the profile according to which you want to harden your system. For a full list of profiles supported in RHEL 9, see [SCAP security profiles supported in RHEL](#) .



WARNING

If not used carefully, running the system evaluation with the **--remediate** option enabled might render the system non-functional. Red Hat does not provide any automated method to revert changes made by security-hardening remediations. Remediations are supported on RHEL systems in the default configuration. If your system has been altered after the installation, running remediation might not make it compliant with the required security profile.

3. Restart your system:

```
# reboot
```

Verification

1. Verify that the system is compliant with the profile, and save the results in an HTML file:

```
$ oscap xccdf eval --report pcidss_report.html --profile pci-dss /usr/share/xml/scap/ssg/content/ssg-rhel9-ds.xml
```

Additional resources

- **scap-security-guide(8)** and **oscap(8)** man pages
- [Scanning the system for security compliance and vulnerabilities](#)
- [Red Hat Insights Security Policy](#)
- [Red Hat Satellite Security Policy](#)

8.4. VERIFYING USBGUARD POLICIES

With the USBGuard software framework, you can protect your systems against intrusive USB devices by using lists of permitted and forbidden devices based on the USB device authorization feature in the kernel.

Prerequisites

- You have created a rule set for USB devices that reflected the requirements of your scenario before the upgrade.
- The **usbguard** service is installed and running on your RHEL 9 system.

Procedure

1. Back up your *.conf files stored in the **/etc/usbguard/** directory.
2. Use the **usbguard generate-policy** to generate a new policy file. Note that the command generates rules for the currently present USB devices only.
3. Compare the newly generated rules against the rules in the previous policy:
 - a. If you identify differences in the rules for the devices that were present when you generated the new policy and the pre-upgrade rules for the same devices, modify the original rules correspondingly also for devices that might be inserted later.
 - b. If there are no differences between the newly generated and the pre-upgrade rules, you can use the policy files created in RHEL 8 without any modification.

Additional resources

- [Protecting systems against intrusive USB devices](#) .

8.5. UPDATING FAPOLICYD DATABASES

The **fapolicyd** software framework controls the execution of applications based on a user-defined policy.

In rare cases, a problem with the **fapolicyd** trust database format can occur. To rebuild the database:

1. Stop the service:

```
# systemctl stop fapolicyd
```

2. Delete the database:

```
# fapolicyd-cli --delete-db
```

3. Start the service:

```
# systemctl start fapolicyd
```

If you added custom trust files to the trust database, update them either individually by using the **fapolicyd-cli -f update <FILE>** command or altogether by using **fapolicyd-cli -f update**. To apply the changes, use either the **fapolicyd-cli --update** command or restart the **fapolicyd** service.

Additionally, custom binaries might require a rebuild for the new RHEL version. Perform any such updates before you update the fapolicyd database.

Additional resources

- [Blocking and allowing applications using fapolicyd](#)

8.6. UPDATING NSS DATABASES FROM DBM TO SQLITE

Many applications automatically convert the NSS database format from DBM to SQLite after you set the **NSS_DEFAULT_DB_TYPE** environment variable to the **sql** value on the system. You can ensure that all databases are converted by using the **certutil** tool.



NOTE

Convert your NSS databases stored in the DBM format before you upgrade to RHEL 9. In other words, perform the following steps on RHEL systems (6, 7, and 8) from which you want to upgrade to RHEL 9.

Prerequisites

- The **nss-tools** package is installed on your system.

Procedure

1. Set **NSS_DEFAULT_DB_TYPE** to **sql** on the system:

```
# export NSS_DEFAULT_DB_TYPE=sql
```

2. Use the conversion command in every directory^[1] that contains NSS database files in the DBM format, for example:

```
# certutil -K -X -d /etc/ipsec.d/
```

Note that you have to provide a password or a path to a password file as a value of the **-f** option if your database file is password-protected, for example:

```
# certutil -K -X -f /etc/ipsec.d/nsspassword -d /etc/ipsec.d/
```

Additional resources

- **certutil(1)** man page.

8.7. MIGRATING CYRUS SASL DATABASES FROM THE BERKELEY DB FORMAT TO GDBM

The RHEL 9 **cyrus-sasl** package is built without the **libdb** dependency, and the **sasldb** plugin uses the GDBM database format instead of Berkeley DB.

Prerequisites

- The **cyrus-sasl-lib** package is installed on your system.

Procedure

- To migrate your existing Simple Authentication and Security Layer (SASL) databases stored in the old Berkeley DB format, use the **cyrusbdb2current** tool with the following syntax:

```
# cyrusbdb2current <sasldb_path> <new_path>
```

Additional resources

- **cyrusbdb2current(1)** man page

[1] RHEL contains a system-wide NSS database in the **/etc/pki/nssdb** directory. Other locations depend on applications you use. For example, Libreswan stores its database in the **/etc/ipsec.d/** directory and Firefox uses the **/home/<username>/.mozilla/firefox/** directory.

CHAPTER 9. TROUBLESHOOTING

You can refer to the following tips to troubleshoot upgrading from RHEL 8 to RHEL 9.

9.1. TROUBLESHOOTING RESOURCES

You can refer to the following troubleshooting resources.

Console output

By default, only error and critical log level messages are printed to the console output by the **Leapp** utility. To change the log level, use the **--verbose** or **--debug** options with the **leapp upgrade** command.

- In *verbose* mode, **Leapp** prints info, warning, error, and critical messages.
- In *debug* mode, **Leapp** prints debug, info, warning, error, and critical messages.

Logs

- The **/var/log/leapp/leapp-upgrade.log** file lists issues found during the initramfs phase.
- The **/var/log/leapp/dnf-debugdata/** directory contains transaction debug data. This directory is present only if the **leapp upgrade** command is executed with the **--debug** option.
- The **/var/log/leapp/answerfile** contains questions required to be answered by **Leapp**.
- The **journalctl** utility provides complete logs.

Reports

- The **/var/log/leapp/leapp-report.txt** file lists issues found during the pre-upgrade phase. The report is also available in the web console, see [Assessing upgradability and applying automated remediations through the web console](#).
- The **/var/log/leapp/leapp-report.json** file lists issues found during the pre-upgrade phase in a machine-readable format, which enables you to process the report using custom scripts. For more information, see [Automating your Red Hat Enterprise Linux pre-upgrade report workflow](#).

9.2. TROUBLESHOOTING TIPS

You can refer to the following troubleshooting tips.

Pre-upgrade phase

- Verify that your system meets all conditions listed in [Planning an upgrade](#).
- Make sure you have followed all steps described in [Preparing for the upgrade](#) for example, your system does not use more than one Network Interface Card (NIC) with a name based on the prefix used by the kernel (**eth**).
- Make sure you have answered all questions required by **Leapp** in the **/var/log/leapp/answerfile** file. If any answers are missing, **Leapp** inhibits the upgrade. For example:
 - Are there no VDO devices on the system?

- Make sure you have resolved all problems identified in the pre-upgrade report, located at **/var/log/leapp/leapp-report.txt**. To achieve this, you can also use the web console, as described in [Assessing upgradability and applying automated remediations through the web console](#).

Example 9.1. Leapp answerfile

The following is an example of an unedited **/var/log/leapp/answerfile** file that has one unanswered question:

```
[check_vdo]
# Title:          None
# Reason:         Confirmation
# ===== check_vdo.confirm
=====
# Label:          Are all VDO devices, if any, successfully converted to LVM management?
# Description:    Enter True if no VDO devices are present on the system or all VDO devices on
the system have been successfully converted to LVM management. Entering True will circumvent
check of failures and undetermined devices. Recognized VDO devices that have not been
converted to LVM management can still block the upgrade despite the answer. All VDO devices
must be converted to LVM management before upgrading.
# Reason:         To maximize safety all block devices on a system that meet the criteria as
possible VDO devices are checked to verify that, if VDOs, they have been converted to LVM
management. If the devices are not converted and the upgrade proceeds the data on unconverted
VDO devices will be inaccessible. In order to perform checking the 'vdo' package must be
installed. If the 'vdo' package is not installed and there are any doubts the 'vdo' package should be
installed and the upgrade process re-run to check for unconverted VDO devices. If the check of
any device fails for any reason an upgrade inhibiting report is generated. This may be problematic
if devices are dynamically removed from the system subsequent to having been identified during
device discovery. If it is certain that all VDO devices have been successfully converted to LVM
management this dialog may be answered in the affirmative which will circumvent block device
checking.
# Type:           bool
# Default:        None
# Available choices: True/False
# Unanswered question. Uncomment the following line with your answer
# confirm =
```

The **Label** field specifies the question that requires an answer. In this example, the question is **Are all VDO devices, if any, successfully converted to LVM management?**

To answer the question, uncomment the last line and enter an answer of **True** or **False**. In this example, the selected answer is **True**:

```
[check_vdo]
...
# Available choices: True/False
# Unanswered question. Uncomment the following line with your answer
confirm = True
```

Download phase

- If a problem occurs during downloading RPM packages, examine transaction debug data located in the **/var/log/leapp/dnf-debugdata/** directory.



NOTE

The `/var/log/leapp/dnf-debugdata/` directory is empty or does not exist if no transaction debug data was produced. This might occur when the required repositories are not available.

Initramfs phase

- During this phase, potential failures redirect you to the Dracut shell. Check the Journal log:

```
# journalctl
```

Alternatively, restart the system from the Dracut shell using the **reboot** command and check the `/var/log/leapp/leapp-upgrade.log` file.

Post-upgrade phase

- If your system seems to be successfully upgraded but booted with the old RHEL 8 kernel, restart the system and check the kernel version of the default entry in GRUB.
- Make sure you have followed the recommended steps in [Verifying the post-upgrade state of the RHEL 9 system](#).
- If your application or a service stops working or behaves incorrectly after you have switched SELinux to enforcing mode, search for denials using the **ausearch**, **journalctl**, or **dmesg** utilities:

```
# ausearch -m AVC,USER_AVC -ts boot
# journalctl -t setroubleshoot
# dmesg | grep -i -e selinux -e type=1400
```

The most common problems are caused by incorrect labeling. See [Troubleshooting problems related to SELinux](#) for more details.

9.3. KNOWN ISSUES

The following are Known Issues you may encounter when upgrading from RHEL 8 to RHEL 9.

- Network teaming currently does not work when the in-place upgrade is performed while Network Manager is disabled or not installed.
- If you use an HTTP proxy, Red Hat Subscription Manager must be configured to use such a proxy, or the **subscription-manager** command must be executed with the **--proxy <hostname>** option. Otherwise, an execution of the **subscription-manager** command fails. If you use the **--proxy** option instead of the configuration change, the upgrade process fails because **Leapp** is unable to detect the proxy. To prevent this problem from occurring, manually edit the **rhsm.conf** file as described in [How to configure HTTP Proxy for Red Hat Subscription Management](#). (BZ#1689294)
- If your RHEL 8 system uses a device driver that is provided by Red Hat but is not available in RHEL 9, **Leapp** inhibits the upgrade. However, if the RHEL 8 system uses a third-party device driver that **Leapp** does not have data for in the `/etc/leapp/files/device_driver_deprecation_data.json` file, **Leapp** does not detect such a driver and proceeds with the upgrade. Consequently, the system might fail to boot after the upgrade.

- If the name of a third-party package (not signed by Red Hat) installed on your system is the same as the name of a package provided by Red Hat, the in-place upgrade fails. To work around this problem, choose one of the following options prior to upgrading:
 - a. Remove the third-party package
 - b. Replace the third-party package with the package provided by Red Hat
- In RHEL 8, you can manage Virtual Data Optimizer (VDO) volumes using either the VDO manager or the Logical Volume Manager (LVM). In RHEL 9, it is only possible to manage VDO volumes using LVM. To continue using VDO-managed volumes on RHEL 9, import those volumes to LVM-managed VDO volumes before the upgrade. For more information, see [Importing existing VDO volumes to LVM](#).
- The in-place upgrade fails on systems with Software Redundant Array of Independent Disks (RAID). (BZ#[1957192](#))
- During the in-place upgrade, the **Leapp** utility usually preserves the network interface controller (NIC) names between RHEL 8 and RHEL 9. However, on some systems, such as systems with network bonding, the NIC names might need to be updated between RHEL 8 and RHEL 9. On those systems, perform the following steps:
 - a. Set the **LEAPP_NO_NETWORK_RENAMING=1** environment variable to prevent the Leapp utility from incorrectly preserving the original RHEL 8 NIC names.
 - b. Perform the in-place upgrade.
 - c. Verify that your network is working correctly. If needed, manually update the network configuration.
(BZ#[1919382](#))
- The in-place upgrade might fail if there is not enough available disk space. The error messages and logs might contain misleading or invalid information about the problem and resolution. To resolve this issue, see the [leapp fails with "There is not enough space on the file system hosting /var/lib/leapp directory to extract the packages"](#) Knowledgebase solution. (BZ#[1832730](#), BZ#[2210300](#))
- If your system boots by using BIOS, the in-place upgrade fails when upgrading the GRUB2 bootloader if the boot disk's embedding area does not contain enough space for the core image installation. This results in a broken system, and can occur when the disk has been partitioned manually, for example using the RHEL 6 **fdisk** utility. To verify whether this issue affects you, perform the following steps:
 - a. Determine which sector starts the first partition on the disk with the installed bootloader:

```
# fdisk -l
```

The standard partitioning, which ensures enough space for the core image, starts on sector 2048.

- b. Determine whether the starting sector provides enough space. The RHEL 9.0 core image requires at least 36 KiB. For example, if the sector size is the standard 512 bytes, then starting on sector 73 or lower would not provide enough space.

**NOTE**

The RHEL 9 core image might be larger than 36 KiB and require a higher starting sector. Always verify how much space the current RHEL 9 core requires.

- c. If the embedding area does not contain enough storage space, perform a fresh installation of the RHEL 9 system instead of performing an in-place upgrade. (BZ#[2181380](#))
- After the in-place upgrade, SSH keys are no longer auto-generated if the system meets the following conditions:
 - The system is on a cloud.
 - The cloud-init package is installed.
 - The `ssh_genkeytypes` configuration is set to `~` in the `/etc/cloud/cloud.cfg` file, which is the default.

This issue prevents the system from connecting by using SSH if the original keys have been removed. To prevent this issue, see the [Unable to SSH to new Virtual Machine after upgrading the template to RHEL 8.7 or 9](#) Knowledgebase solution. (BZ# [2210012](#))

9.4. OBTAINING SUPPORT

To open a support case, select *RHEL 8* as the product, and provide a **sosreport** from your system.

- To generate a **sosreport** on your system, run:

```
# sosreport
```

Note that you can leave the case ID empty.

For details on generating a sosreport, see the solution [What is an sosreport and how to create one in Red Hat Enterprise Linux?](#).

For more information about opening and managing a support case on the Customer Portal, see the article [How do I open and manage a support case on the Customer Portal?](#) .

CHAPTER 10. RELATED INFORMATION

You can refer to the following instructional materials:

- [Red Hat Enterprise Linux technology capabilities and limits](#)
- [Supported in-place upgrade paths for Red Hat Enterprise Linux](#)
- [Considerations in adopting RHEL 9](#)
- [Customizing your Red Hat Enterprise Linux in-place upgrade](#)
- [Automating your Red Hat Enterprise Linux pre-upgrade report workflow](#)
- [Using configuration management systems to automate parts of the Leapp pre-upgrade and upgrade process on Red Hat Enterprise Linux](#)
- [Upgrading from RHEL 7 to RHEL 8](#)
- [Converting from an RPM-based Linux distribution to RHEL](#)
- [How to in-place upgrade SAP environments from RHEL 8 to RHEL 9](#)
- [Red Hat Insights Documentation](#)
- [Upgrades-related Knowledgebase articles and solutions](#)
- [The best practices and recommendations for performing RHEL Upgrade using Leapp](#)
- [Leapp upgrade FAQ \(Frequently Asked Questions\)](#)

APPENDIX A. RHEL 8 REPOSITORIES

Before the upgrade, ensure you have appropriate repositories enabled as described in step 4 of the procedure in [Preparing a RHEL 8 system for the upgrade](#).

If you plan to use Red Hat Subscription Manager during the upgrade, you **must enable** the following repositories before the upgrade by using the **subscription-manager repos --enable *repository_id*** command:

Table A.1. RHEL 8 repositories

Architecture	Repository	Repository ID
64-bit Intel and AMD	Base	rhel-8-for-x86_64-baseos-rpms
	AppStream	rhel-8-for-x86_64-appstream-rpms
64-bit ARM	Base	rhel-8-for-aarch64-baseos-rpms
	Extras	rhel-8-for-aarch64-appstream-rpms
IBM POWER (little endian)	Base	rhel-8-for-ppc64le-baseos-rpms
	AppStream	rhel-8-for-ppc64le-appstream-rpmss
IBM Z	Base	rhel-8-for-s390x-baseos-rpms
	AppStream	rhel-8-for-s390x-appstream-rpms

You **can enable** the following repositories before the upgrade by using the **subscription-manager repos --enable *repository_id*** command:

Table A.2. Voluntary RHEL 8 repositories

Architecture	Repository	Repository ID
64-bit Intel and AMD	Code Ready Linux Builder	codeready-builder-for-rhel-8-x86_64-rpms
	Supplementary	rhel-8-for-x86_64-supplementary-rpms
64-bit ARM	Code Ready Linux Builder	codeready-builder-for-rhel-8-aarch64-rpms
	Supplementary	rhel-8-for-aarch64-supplementary-rpms
IBM POWER (little endian)	Code Ready Linux Builder	codeready-builder-for-rhel-8-ppc64le-rpms

Architecture	Repository	Repository ID
	Supplementary	rhel-8-for-ppc64le-supplementary-rpms
IBM Z	Code Ready Linux Builder	codeready-builder-for-rhel-8-s390x-rpms
	Supplementary	rhel-8-for-s390x-supplementary-rpms



NOTE

If you have enabled a RHEL 8 Code Ready Linux Builder or a RHEL 8 Supplementary repository before an in-place upgrade, **Leapp** enables the RHEL 8 CodeReady Linux Builder or the RHEL 8 Supplementary repositories, respectively. For more information, see the [Package manifest](#).

If you decide to use custom repositories, enable them per instructions in [Configuring custom repositories](#).

APPENDIX B. RHEL 9 REPOSITORIES

If your system is registered to the Red Hat Content Delivery Network (CDN) using the Red Hat Subscription Manager (RHSM), RHEL 9 repositories are automatically enabled during the in-place upgrade. However, on systems registered to Red Hat Satellite using RHSM, you must manually enable and synchronize both RHEL 8 and RHEL 9 repositories before running the pre-upgrade report.



NOTE

Make sure to enable the target OS version of each repository, for example 9.0. If you have enabled only the RHEL 9 version of the repositories, the in-place upgrade is inhibited.

If you plan to use Red Hat Satellite during the upgrade, you **must enable and synchronize** at least the following RHEL 9 repositories before the upgrade using either the Satellite web UI or the **hammer repository-set enable** and **hammer product synchronize** commands:

Table B.1. RHEL 9 repositories

Architecture	Repository	Repository ID	Repository name	Release version
64-bit Intel and AMD	BaseOS	rhel-9-for-x86_64-baseos-rpms	Red Hat Enterprise Linux 9 for x86_64 - BaseOS (RPMs)	x86_64 <target_os_version>
	AppStream	rhel-9-for-x86_64-appstream-rpms	Red Hat Enterprise Linux 9 for x86_64 - AppStream (RPMs)	x86_64 <target_os_version>
64-bit ARM	BaseOS	rhel-9-for-aarch64-baseos-rpms	Red Hat Enterprise Linux 9 for ARM 64 - BaseOS (RPMs)	aarch64 <target_os_version>
	AppStream	rhel-9-for-aarch64-appstream-rpms	Red Hat Enterprise Linux 9 for ARM 64 - AppStream (RPMs)	aarch64 <target_os_version>
IBM Power (little endian)	BaseOS	rhel-9-for-ppc64le-baseos-rpms	Red Hat Enterprise Linux 9 for Power, little endian - BaseOS (RPMs)	ppc64le <target_os_version>

Architecture	Repository	Repository ID	Repository name	Release version
	AppStream	rhel-9-for-ppc64le-appstream-rpms	Red Hat Enterprise Linux 9 for Power, little endian - AppStream (RPMs)	ppc64le <target_os_version>
IBM Z	BaseOS	rhel-9-for-s390x-baseos-rpms	Red Hat Enterprise Linux 9 for IBM z Systems - BaseOS (RPMs)	s390x <target_os_version>
	AppStream	rhel-9-for-s390x-appstream-rpms	Red Hat Enterprise Linux 9 for IBM z Systems - AppStream (RPMs)	s390x <target_os_version>

Replace <target_os_version> with the target OS version, for example **9.0**.