Red Hat Enterprise Linux 10

Upgrading from RHEL 9 to RHEL 10

Instructions for an in-place upgrade from Red Hat Enterprise Linux 9 to Red Hat Enterprise Linux 10

Last Updated: 2025-06-09

Red Hat Enterprise Linux 10 Upgrading from RHEL 9 to RHEL 10

Instructions for an in-place upgrade from Red Hat Enterprise Linux 9 to Red Hat Enterprise Linux 10

Legal Notice

Abstract

This document provides instructions on how to perform an in-place upgrade from Red Hat Enterprise Linux 9 to Red Hat Enterprise Linux 10 using the Leapp utility. During the in-place upgrade, the existing RHEL 9 operating system is replaced by a RHEL 10 version.

Table of Contents

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

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

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 9 operating system on your system with a RHEL 10 version.

IMPORTANT

You can perform the in-place upgrade only from one major RHEL version to the next consecutive one, for example, RHEL 8 to RHEL 9 or RHEL 9 to RHEL 10. If you want to upgrade a system across multiple versions, such as from RHEL 8 to RHEL 10, you must perform multiple in-place upgrades to reach your target version. For more information, see In-place upgrades over multiple RHEL major versions by using Leapp .

Currently, it is possible to perform an in-place upgrade from the following source RHEL 9 minor versions to the following target RHEL 10 minor versions:

Table 1.1. Supported upgrade paths

System configuration	Source OS version	Target OS version
RHEL	RHEL 9.6	RHEL 10.0

For more information about supported upgrade paths, see Supported in-place upgrade paths for Red Hat Enterprise Linux and the In-place upgrade Support Policy .

CHAPTER 2. PLANNING AN UPGRADE TO RHEL 10

Before beginning your upgrade from RHEL 9 to RHEL 10, review system requirements, limitations, and other considerations.

2.1. PLANNING AN UPGRADE FROM RHEL 9 TO RHEL 10

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

Consider the following before upgrading to RHEL 10:

- **Applications** - You can migrate applications installed on your system by using the **Leapp** utility. 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.

- **Boot loader** - It is not possible to switch the boot loader from BIOS to UEFI on RHEL 9 or RHEL 10. If your RHEL 9 system uses BIOS and you want your RHEL 10 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?
- **Customization** - To use custom repositories, see the Configuring custom repositories Knowledgebase article.
- **Downtime** - The upgrade process can take from several minutes to several hours.
- **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.
- **Language** - All **Leapp** reports, logs, and other generated documentation are in English, regardless of the language configuration.
- **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 10 are met.

  - You have access to up-to-date content for the selected source and target OS versions. See Preparing a RHEL 9 system for the upgrade for more information.

- **Public clouds** - The in-place upgrade is not currently supported for on-demand Pay-As-You-Go (PAYG) instances using Red Hat Update Infrastructure (RHUI) .

- **Real Time for Network Functions Virtualization (NFV) in Red Hat OpenStack Platform** - Upgrades on real-time systems are supported.

- **RHEL for Real Time** - Upgrades on real-time systems are supported.

- **SAP HANA** - Upgrades with SAP HANA are currently unsupported.

- **Satellite**

  - **Client** - If you manage your hosts through Satellite, you can upgrade multiple hosts simultaneously from RHEL 9 to RHEL 10 using the Satellite web UI. For more information, see Upgrading Hosts to Next Major Red Hat Enterprise Linux Release .

  - **Server and Capsule** - You can upgrade Satellite Servers and Capsules starting in Satellite 6.16. For more information, see Upgrading Satellite or Capsule to RHEL 9 in-place by using Leapp.

- **Security** - 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 10 .

  - During the upgrade process, the **Leapp** utility sets SELinux mode to permissive.

  - **Leapp** supports in-place upgrades of RHEL 9.6 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**

  - **Backup** - 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.

  - **Encryption** - Systems with encrypted storage can be upgraded if the storage uses the LUKS2 format configured with the Clevis TPM 2.0 token. For more information, see Configuring manual enrollment of LUKS-encrypted volumes by using a TPM 2.0 policy .

Notable known limitations of the **Leapp** utility include:

- **Known limitations** - Notable known limitations of **Leapp** currently include:
  - Network based multipath and network storage that use Ethernet or Infiniband are not supported for the upgrade. This includes SAN using FCoE and booting from SAN using FC. Note that SAN using FC are supported.
  - The in-place upgrade is currently unsupported for on-demand PAYG instances on the remaining Public Clouds 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. For more information about using a RHEL 9 Ansible Tower installation on RHEL 9, see the Red Hat Knowledgebase solution How do I migrate my Ansible Automation Platform installation from one environment to another? Knowledgebase solution.
  - Red Hat JBoss Enterprise Application Platform (EAP) is not supported for the upgrade to RHEL 10. You must manually install and configure JBoss EAP on your system after the upgrade. For more information, see the Red Hat Knowledgebase solution In-place Migrating of Jboss EAP and websphere servers along with Linux using leapp utility.

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 10. Note that the Advisor recommendation considers only the RHEL 9 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
- Leapp upgrade FAQ (Frequently Asked Questions)

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 9 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 9 SYSTEM FOR THE UPGRADE

This procedure describes the steps that are necessary before performing an in-place upgrade to RHEL 10 by using the **Leapp** utility.

If you do not plan to use Red Hat Subscription Manager (RHSM) during the upgrade process, follow instructions in Performing an in-place upgrade without Red Hat Subscription Manager .

Prerequisites

- The system meets conditions listed in Planning an upgrade.
- If the system has been previously upgraded from RHEL 8 to RHEL 9, ensure that all required post-upgrade steps have been completed. For more information, see Performing post-upgrade tasks in the Upgrading from RHEL 8 to RHEL 9 guide.
- Optional: You have reviewed the best practices in The best practices and recommendations for performing RHEL Upgrade using Leapp Knowledgebase article.
- You have ensured that your system has been successfully registered to the Red Hat Content Delivery Network (CDN) or Red Hat Satellite by using RHSM.
- Satellite-registered systems only: You have completed the steps in Preparing a Satellite system for the upgrade to ensure that your system meets the requirements for the upgrade.

Procedure

1. Optional: Unmount non-system OS file systems that are not required for the upgrade and comment them out from the **/etc/fstab** file. For example, this includes file systems containing only data files unrelated to the system itself. This can reduce the amount of time needed for the upgrade process and prevent potential issues related to third-party applications that are not migrated properly during the upgrade by custom or third-party actors.

2. If you are upgrading by using RHSM, verify that the system is registered to an account with Simple Content Access (SCA) enabled:

```
# 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
```

3. 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 9 repositories.

```
# subscription-manager repos --enable rhel-9-for-x86_64-baseos-rpms --enable rhel-9-for-x86_64-appstream-rpms
```

NOTE

Optional: Enable the CodeReady Linux Builder (also known as Optional) or Supplementary repositories. For more information about the content of these repositories, see the Package manifest.

4. Set the system release version:

```
# subscription-manager release --set 9.6
```

5. If you use the **dnf versionlock** plugin to lock packages to a specific version, clear the lock by running:

```
# dnf versionlock clear
```

6. Ensure that you have up-to-date **leapp** and **leapp-repository** packages:

   a. RHEL 9.6: version **0.19.0** of the **leapp** package and version **0.22.0** of the **leapp-repository** package.
   The **leapp-repository** package contains the **leapp-upgrade-el9toel10** RPM package.

NOTE

Disconnected systems only:download the following packages from the Red Hat Customer Portal:

- **leapp**
- **leapp-deps**
- **python3-leapp**
- **leapp-upgrade-el9toel10**
- **leapp-upgrade-el9toel10-deps**

7. Install the **Leapp** utility:

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

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

```
# dnf update
# reboot
```

9. Optional: Review, remediate, and then remove the **rpmnew** and **rpmsave** files.

10. If you use a configuration management system, ensure that it does not interfere with the in-place upgrade process:

    - If your configuration management system has 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 your configuration management system has agentless architecture, do not execute the configuration and deployment file. For example, if your system has Ansible, do not execute an Ansible playbook during the upgrade.

    > WARNING
    >
    > 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.

11. If you are upgrading by using an ISO image, verify that the ISO image contains the target OS version, for example, RHEL 10.0, and is saved to a persistent local mount point to ensure that the **Leapp** utility can access the image throughout the upgrade process.

3.2. PREPARING A SATELLITE-REGISTERED SYSTEM FOR THE UPGRADE

Before you can perform an in-place upgrade to RHEL 10 of a system that is registered to Satellite, you must prepare your system.. 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 9 system for the upgrade .

Prerequisites

- You have administrative privileges for the Satellite Server.

- Satellite is on a version in full or maintenance support. For more information, see Red Hat Satellite Product Life Cycle.

Procedure

1. 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 .

2. Enable and synchronize all required RHEL 9 and RHEL 10 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 10 repositories, enable the target OS version, for example, RHEL 10.0, of each repository. If you enable only the RHEL 10 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 9 for x86_64 - AppStream (RPMs)
     rhel-9-for-x86_64-appstream-rpms

     x86_64 <*source_os_version*>

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

     x86_64 <*source_os_version*>

   - Red Hat Enterprise Linux 10 for x86_64 - AppStream (RPMs)
     rhel-10-for-x86_64-appstream-rpms

     x86_64 <*target_os_version*>

   - Red Hat Enterprise Linux 10 for x86_64 - BaseOS (RPMs)
     rhel-10-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, 9.6 and 10.0.

     For other architectures, see RHEL 9 repositories and RHEL 10 repositories.

     For more information, see the *Importing Content* chapter in the *Managing Content Guide* for the particular version of Red Hat Satellite.

3. Attach the content host to a Content View containing the required RHEL 9 and RHEL 10 repositories.
   For more information, see the *Managing Content Views* chapter in the *Managing Content Guide* for the particular version of Red Hat Satellite.

Verification

1. Verify that the correct RHEL 9 and RHEL 10 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-9' --content-view <content_view_name> --organization <organization> --lifecycle-environment <lifecycle_environment>
# hammer repository list --search 'content_label ~ rhel-10' --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_environement>* with the name of the lifecycle environment..

2. Verify that the correct RHEL 10 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 9 repositories are enabled in the host. For example:

```
# subscription-manager repos --list-enabled | grep "^Repo ID"
Repo ID:   rhel-9-for-x86_64-baseos-rpms
Repo ID:   rhel-9-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

The pre-upgrade assessment does not modify the system configuration, but it does consume non-negligible space in the **/var/lib/leapp** directory. In most cases, the pre-upgrade assessment requires up to 4 GB of space, but the actual size depends on your system configuration. If there is not enough space in the hosted file system, the pre-upgrade report might not show complete results of the analysis. To prevent issues, ensure that your system has enough space in the **/var/lib/leapp** directory or move the directory to a dedicated partition so that space consumption does not affect other parts of the system.

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.
- 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 OF RHEL 9 TO RHEL 10 FROM THE COMMAND LINE

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

Prerequisites

- The steps listed in Preparing for the upgrade have been completed.
- You are logged in to root with the unconfined SELinux role.

> **NOTE**
>
> If you are using **sudo**, you must use the **-r unconfined_r -t unconfined_t** options when running each leapp command, for example:
>
> ```
> $ sudo -r unconfined_r -t unconfined_t leapp preupgrade
> ```

Procedure

1. On your RHEL 9 system, perform the pre-upgrade phase:

   ```
   # leapp preupgrade --target <_target_os_version_>
   ```

   Replace *target_os_version* with the target OS version, for example **10.0**. If no target OS version is defined, **Leapp** uses the default target OS version specified in the table 1.1 in Supported upgrade paths.

   - 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> ...
     ```

     Replace *repository_id* with the repository IDs.

   - If you are upgrading without RHSM or by using RHUI, add the **--no-rhsm** option.

   - If you have an Extended Upgrade Support (EUS) or Advanced Update Support (AUS) subscription, add the **--channel *<channel>*** option. Replace *<channel>* with the channel name, for example, **eus** or **aus**.

   - If you are using RHEL for Real Time or the Real Time for Network Functions Virtualization (NFV) in your Red Hat OpenStack Platform, enable the deployment by using the **--enablerepo** option. For example:

     ```
     # leapp preupgrade --enablerepo rhel-10-for-x86_64-rt-rpms
     ```

     For more information, see Configuring Real-Time Compute.

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 Troubleshooting tips.

   NOTE

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

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

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

4.2. ASSESSING UPGRADABILITY OF RHEL 9 TO RHEL 10 AND APPLYING AUTOMATED REMEDIATIONS THROUGH THE WEB CONSOLE

Identify potential problems in the pre-upgrade phase before the upgrade and apply automated remediations by using the web console. See Getting started using the RHEL web console for more information about the web console.

Prerequisites

- You have completed the steps listed in Preparing for the upgrade.
- You are logged in to root with the unconfined SELinux role.

NOTE

If you are using **sudo**, you must use the **-r unconfined_r -t unconfined_t** options when running each leapp command, for example:

```
$ sudo -r unconfined_r -t unconfined_t leapp preupgrade
```

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**.

3. On your RHEL 9 system, perform the pre-upgrade phase either from the command line or from the web console terminal:

   ```
   # leapp preupgrade --target <target_os_version>
   ```

   Replace *target_os_version* with the target OS version, for example **10.0**. If no target OS version is defined, **Leapp** uses the default target OS version specified in the table 1.1 in Supported upgrade paths.

   - 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 by using RHUI, add the **--no-rhsm** option.

   - If you have an Extended Upgrade Support (EUS) or Advanced Update Support (AUS) subscription, add the **--channel *<channel>*** option. Replace *<channel>* with the channel name, for example, **eus** or`aus`.

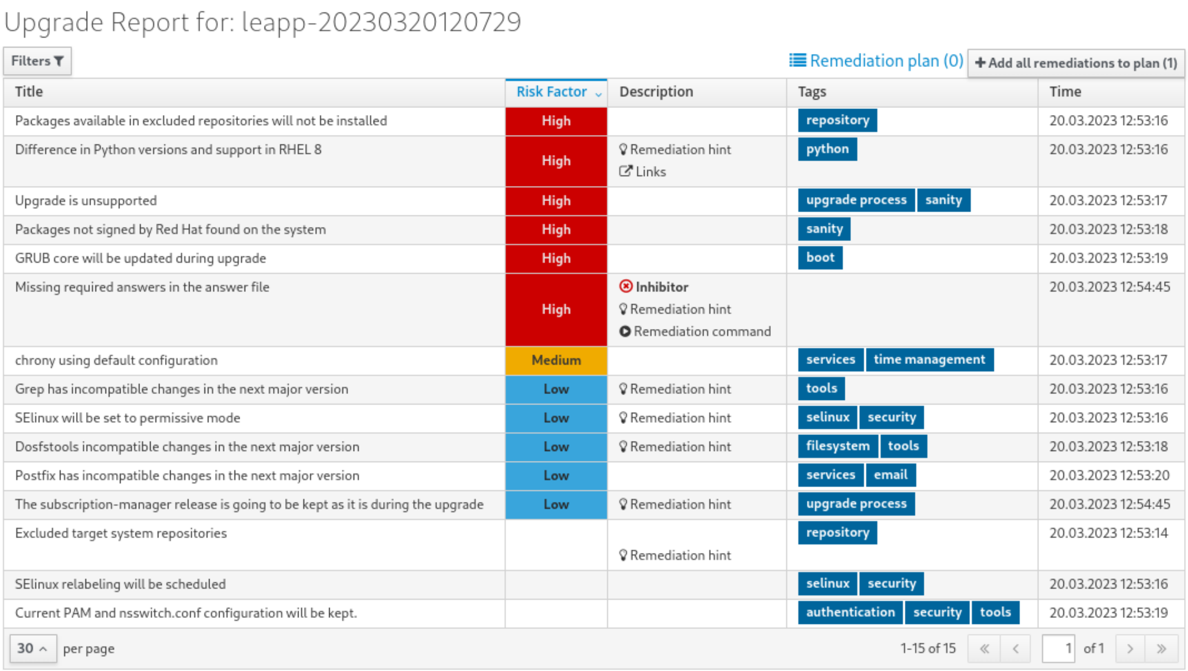
   - If you are using RHEL for Real Time or the Real Time for Network Functions Virtualization (NFV) in your Red Hat OpenStack Platform, enable the deployment by using the **--enablerepo** option. For example:

     ```
     # leapp preupgrade --enablerepo rhel-10-for-x86_64-rt-rpms
     ```

     For more information, see Configuring Real-Time Compute.

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: leapp-20230320120729

Title	Risk Factor	Description	Tags	Time
Packages available in excluded repositories will not be installed	High		repository	20.03.2023 12:53:16
Difference in Python versions and support in RHEL 8	High	Remediation hint Links	python	20.03.2023 12:53:16
Upgrade is unsupported	High		upgrade process sanity	20.03.2023 12:53:17
Packages not signed by Red Hat found on the system	High		sanity	20.03.2023 12:53:18
GRUB core will be updated during upgrade	High		boot	20.03.2023 12:53:19
Missing required answers in the answer file	High	Inhibitor Remediation hint Remediation command		20.03.2023 12:54:45
chrony using default configuration	Medium		services time management	20.03.2023 12:53:17
Grep has incompatible changes in the next major version	Low	Remediation hint	tools	20.03.2023 12:53:16
SElinux will be set to permissive mode	Low	Remediation hint	selinux security	20.03.2023 12:53:16
Dosfstools incompatible changes in the next major version	Low	Remediation hint	filesystem tools	20.03.2023 12:53:18
Postfix has incompatible changes in the next major version	Low		services email	20.03.2023 12:53:20
The subscription-manager release is going to be kept as it is during the upgrade	Low	Remediation hint	upgrade process	20.03.2023 12:54:45
Excluded target system repositories		Remediation hint	repository	20.03.2023 12:53:14
SElinux relabeling will be scheduled			selinux security	20.03.2023 12:53:16
Current PAM and nsswitch.conf configuration will be kept.			authentication security tools	20.03.2023 12:53:19

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>
   ```

   NOTE

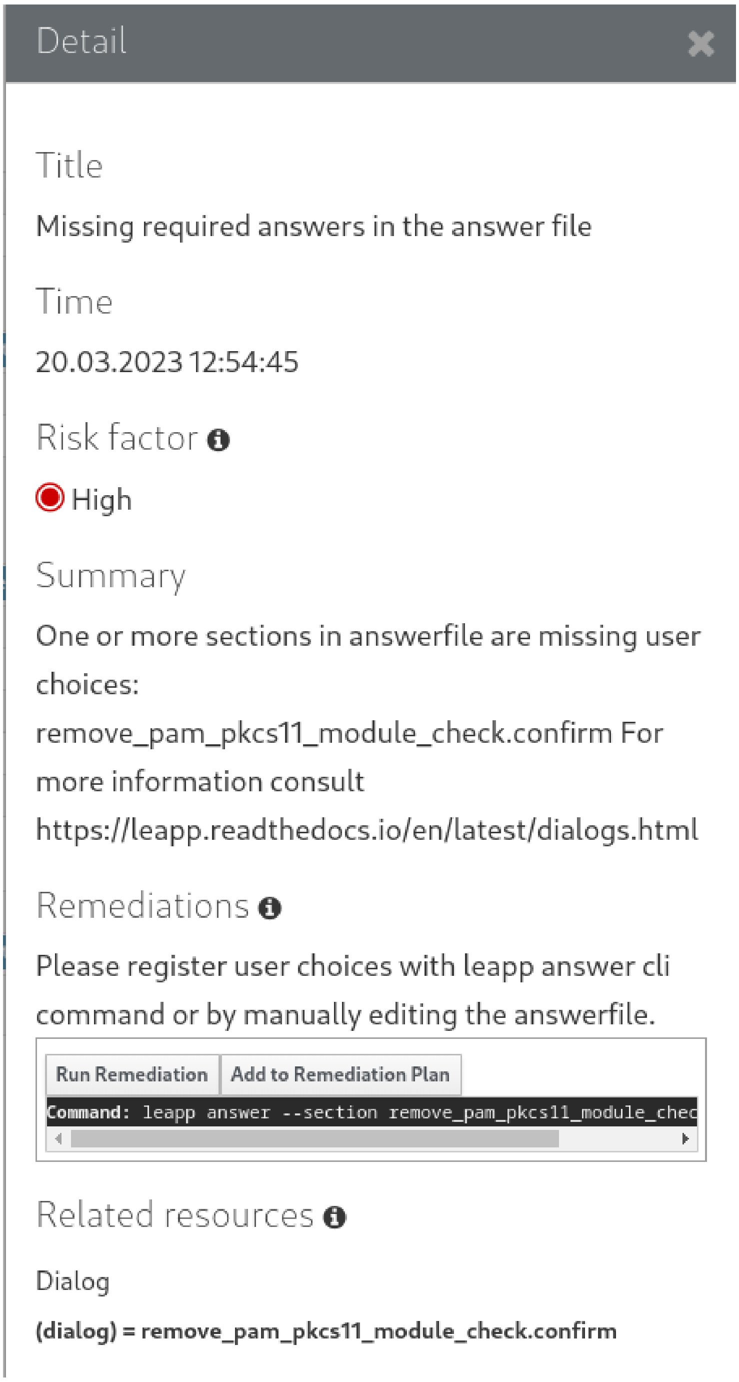
   You can also manually edit the **/var/log/leapp/answerfile** file, uncomment the confirm line of the file by deleting the **#** symbol, and confirm your answer as **True** or **False**. For more information, see the Troubleshooting tips.

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



Title

Missing required answers in the answer file

Time

20.03.2023 12:54:45

Risk factor

● High

Summary

One or more sections in answerfile are missing user choices: remove_pam_pkcs11_module_check.confirm For more information consult https://leapp.readthedocs.io/en/latest/dialogs.html

Remediations

Please register user choices with leapp answer cli command or by manually editing the answerfile.

```
Command: leapp answer --section remove_pam_pkcs11_module_chec
```

Related resources

Dialog

(dialog) = remove_pam_pkcs11_module_check.confirm

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

After you have completed the preparatory steps and reviewed and resolved the problems found in the pre-upgrade report, you can perform the in-place upgrade on your system.

5.1. PERFORMING THE UPGRADE FROM RHEL 9 TO RHEL 10

This procedure lists steps required to perform the upgrade frp, RHEL 9 to RHEL 10 by 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.
- You have temporarily disabled antivirus software to prevent the upgrade from failing.

Procedure

1. Ensure that 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. You can use the following backup options:
   - Create a full backup of your system by using 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?.
   - Create a snapshot of your system by using LVM snapshots or RAID splitting. In case of upgrading a virtual machine, you can create a snapshot of the whole VM. You can also manage snapshot and rollback boot entries by using the Boom utility. For more information, see What is BOOM and how to install it? and Managing system upgrades with snapshots.

     NOTE

     Because LVM snapshots do not create a full backup of your system, you might not be able to recover your system after certain upgrade failures. As a result, it is safer to create a full backup by using the ReaR utility.

2. On your RHEL 9 system, start the upgrade process:

   ```
   # leapp upgrade --target <_target_os_version_>
   ```

   Replace *target_os_version* with the target OS version, for example **10.0**. If no target OS version is defined, **Leapp** uses the default target OS version specified in the table 1.1 in Supported upgrade paths.

   - 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) or Advanced Update Support (AUS) subscription, add the **--channel *channel*** option. Replace *channel* with the value you used with the **leapp preupgrade** command, for example, **eus** or **aus**. Note that you must use the same value with the **--channel** option in both the **leapp preupgrade** and **leapp upgrade** commands.
- If you are using RHEL for Real Time or the Real Time for Network Functions Virtualization (NFV) in your Red Hat OpenStack Platform, enable the deployment by using the **--enablerepo** option. For example:

  ```
  # leapp upgrade --enablerepo rhel-10-for-x86_64-rt-rpms
  ```

  For more information, see Configuring Real-Time Compute.

3. At the beginning of the upgrade process, **Leapp** repeats 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.
4. Manually reboot the system:

   ```
   # reboot
   ```

   In this phase, the system boots into a RHEL 10-based initial RAM disk image, initramfs. **Leapp** upgrades all packages and automatically reboots to the RHEL 10 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.
5. Log in to the RHEL 10 system and verify its state as described in Verifying the post-upgrade state.
6. Perform all post-upgrade tasks described in the upgrade report and in Performing post-upgrade tasks.

CHAPTER 6. VERIFYING THE POST-UPGRADE STATE

After performing the in-place upgrade to RHEL 10, verify that the system is in the correct state. Doing so allows you to identify and correct any critical errors that could impact your system.

6.1. VERIFYING THE POST-UPGRADE STATE OF THE RHEL 10 SYSTEM

This procedure lists Verification recommended to perform after an in-place upgrade to RHEL 10.

Prerequisites

- The system has been upgraded following the steps described in Performing the upgrade and you have been able to log in to RHEL 10.

Procedure

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

- Verify that the current OS version is RHEL 10. For example:

  ```
  # cat /etc/redhat-release
  Red Hat Enterprise Linux release 10.0 (Coughlan)
  ```

- Check the OS kernel version. For example:

  ```
  # uname -r
  6.12.0-55.2.1.el10_0.x86_64
  ```

  Note that **.el10** is important and the version should not be earlier than 6.12.0.

- 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:      10.0
    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: 10.0
    ```

- 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 ON THE RHEL 10 SYSTEM

After the in-place upgrade, clean up your RHEL 10 system by removing unneeded packages, disable incompatible repositories, and update the rescue kernel and initial RAM disk.

7.1. 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 and you have been able to log in to RHEL 10.
- The status of the in-place upgrade has been verified following the steps described in Verifying the post-upgrade state.

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 9 packages, including remaining **Leapp** packages.
   a. Locate remaining RHEL 9 packages:

   ```
   # rpm -qa | grep -e '\.el[789]' | grep -vE '^(gpg-pubkey|libmodulemd|katello-ca-consumer)'
   | sort
   ```

   b. Remove remaining RHEL 9 packages from your RHEL 10 system. To ensure that RPM dependencies are maintained, use **DNF** when performing this action. Review the transaction before accepting to ensure no packages are unintentionally removed.
   For example:

   ```
   # dnf remove $(rpm -qa | grep \.el[789] | grep -vE 'gpg-pubkey|libmodulemd|katello-ca-
   consumer')
   ```

   c. Remove remaining **Leapp** dependency packages:

   ```
   # dnf remove leapp-deps-el10 leapp-repository-deps-el10
   ```

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 10-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 rescue kernel and related initial RAM disk:

      ```
      # /usr/lib/kernel/install.d/51-dracut-rescue.install add "$(uname -r)" /boot "/boot/vmlinuz-$(uname -r)"
      ```

   c. If your system is on the IBM Z architecture, update the **zipl** boot loader:

      ```
      # zipl
      ```

6. . Optional: Check existing configuration files:

   - Review, remediate, and then remove the **rpmnew**, **rpmsave**, and **leappsave** files. Note that **rpmsave** and **leappsave** are equivalent and can be handled similarly. For more information, see What are rpmnew & rpmsave files?

   - Remove configuration files for RHEL 9 DNF modules from the **/etc/dnf/modules.d/** directory that are no longer valid. Note that these files have no effect on the system when related DNF modules do not exist.

7. 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 /boot/vmlinuz-*rescue*
```

3. Review the grubby output and verify that no RHEL 9 boot entries are configured:

```
# grubby --info ALL
```

4. Verify that no files related to previous RHEL are present in the /boot/loader/entries file:

```
# grep -r ".el9" "/boot/loader/entries/" || echo "Everything seems ok."
```

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. To restore system security, switch SELinux to enforcing mode again and verify the system-wide cryptographic policy. You may also want to remediate the system to be compliant with a specific security profile. Also, some security-related components require pre-update steps for a correct upgrade.

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 described in Verifying the post-upgrade state.

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 9. For example, if you used the **DEFAULT** cryptographic policy in RHEL 9, your system upgraded to RHEL 10 also uses **DEFAULT**. Note that specific settings in predefined policies differ, and RHEL 10 cryptographic policies contain more strict and more secure default values. 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
```

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 on your system

8.3. UPGRADING A SYSTEM HARDENED TO A SECURITY BASELINE

To get a fully hardened system after a successful upgrade to RHEL 10, 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 9 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 RHEL 10 during the upgrade.

NOTE

You cannot use the same SCAP content for scanning RHEL 9 and RHEL 10. 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 configuration compliance.

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 10 system.

Procedure

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

   ```
   $ ls /usr/share/xml/scap/ssg/content/
   ...
   ssg-rhel10-ds.xml
   ...
   ```

   See the Viewing profiles for configuration compliance section for more information.

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

   ```
   # oscap xccdf eval --profile <profile_ID> --remediate /usr/share/xml/scap/ssg/content/ssg-rhel10-ds.xml
   ```

   Replace ***<profile_ID>*** with the ID of the profile according to which you want to harden your system. For a full list of profiles supported in RHEL 10, see SCAP Security Guide profiles supported in RHEL 10.

> **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-rhel10-ds.xml
```

Additional resources

- **scap-security-guide(8)** and **oscap(8)** man pages on your system
- Scanning the system for configuration compliance
- 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 10 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 9 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

CHAPTER 9. TROUBLESHOOTING

You can refer to the following tips to troubleshoot upgrading from RHEL 9 to RHEL 10.

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 9 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.
- 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 FOR THE RHEL 9 TO RHEL 10 UPGRADE

The following are Known Issues you may encounter when upgrading.

- If your RHEL 9 system uses a device driver that is provided by Red Hat but is not available in RHEL 10, **Leapp** inhibits the upgrade. However, if the RHEL 9 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
- The in-place upgrade might fail 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 9 and RHEL 10. However, on some systems, such as systems with

network bonding, the NIC names might need to be updated between RHEL 9 and RHEL 10. 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 9 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)

- If any of the mounted file systems that are defined in the **/etc/fstab** file do not have the **shared** propagation flag set, the upgrade might fail. To prevent this issue, remount these file systems to set them as shared:

  ```
  # mount -o remount --make-shared <mountpoint>
  ```

  Replace *mountpoint* with the mountpoint of each file system.

  For more information, see the Red Hat Knowledgebase solution Leapp "Can not load RPM file" during the DNF transaction check. (RHEL-23449)

- 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. For more information, see the Red Hat Knowledgebase solution How to configure HTTP Proxy for Red Hat Subscription Management. (BZ#1689294)

- For systems that require a proxy to access RHEL 9 content, you usually need to configure the use of the proxy by DNF in the **/etc/dnf/dnf.conf** configuration file. If the current DNF configuration is not compatible with the DNF version on the target system, specify the valid target configuration in the **/etc/leapp/files/dnf.conf** configuration file. For more information, see the Red Hat Knowledgebase solution How does Leapp work with a proxy?

- The kerberos client might break after the upgrade if it is configured to use the deprecated **/etc/ssl/certs/ca-certificates.crt** file for root certificates. To fix the configuration, use the **/etc/pki/ca-trust/extracted/pem/tls-ca-bundle.pem** file instead. (RHEL-65265)

9.4. OBTAINING SUPPORT

To open a support case, select *RHEL 9* 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 more information about generating a sosreport, see the Red Hat Knowledgebase 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 Red Hat Knowledgebase solution, 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:

- Upgrade your Red Hat Enterprise Linux Infrastructure
- Red Hat Enterprise Linux technology capabilities and limits
- Supported in-place upgrade paths for Red Hat Enterprise Linux
- In-place upgrade Support Policy
- Considerations in adopting RHEL 10
- 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 8 to RHEL 9
- Red Hat Insights Documentation
- Upgrades-related Knowledgebase articles and solutions (Red Hat Knowledgebase)
- The best practices and recommendations for performing RHEL Upgrade using Leapp
- Leapp upgrade FAQ (Frequently Asked Questions)

APPENDIX A. 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 9 and RHEL 10 repositories before running the pre-upgrade report.

NOTE

Make sure to enable the source OS version of each repository, for example 9.6. 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 A.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 *<source_os_version>*
	AppStream	**rhel-9-for-x86_64-appstream-rpms**	Red Hat Enterprise Linux 9 for x86_64 - AppStream (RPMs)	x86_64 *<source_os_version>*
64-bit ARM	BaseOS	**rhel-9-for-aarch64-baseos-rpms**	Red Hat Enterprise Linux 9 for ARM 64 - BaseOS (RPMs)	aarch64 *<source_os_version>*
	AppStream	**rhel-9-for-aarch64-appstream-rpms**	Red Hat Enterprise Linux 9 for ARM 64 - AppStream (RPMs)	aarch64 *<source_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 *<source_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 *<source_os_version>*
IBM Z	BaseOS	**rhel-9-for-s390x-baseos-rpms**	Red Hat Enterprise Linux 9 for IBM z Systems – BaseOS (RPMs)	s390x *<source_os_version>*
	AppStream	**rhel-9-for-s390x-appstream-rpms**	Red Hat Enterprise Linux 9 for IBM z Systems – AppStream (RPMs)	s390x *<source_os_version>*

Replace *<source_os_version>* with the source OS version, for example **9.6**.

APPENDIX B. RHEL 10 REPOSITORIES

If your system is registered to the Red Hat Content Delivery Network (CDN) by using the Red Hat Subscription Manager (RHSM), RHEL 10 repositories are automatically enabled during the in-place upgrade. However, on systems registered to Red Hat Satellite by using RHSM, you must manually enable and synchronize both RHEL 9 and RHEL 10 repositories before running the pre-upgrade report.

NOTE

Make sure to enable the target OS version of each repository, for example 10.0. If you have enabled only the RHEL 10 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 10 repositories before the upgrade by using either the Satellite web UI or the **hammer repository-set enable** and **hammer product synchronize** commands:

Table B.1. RHEL 10 repositories

Architecture	Repository	Repository ID	Repository name	Release version
64-bit Intel and AMD	BaseOS	**rhel-10-for-x86_64-baseos-rpms**	Red Hat Enterprise Linux 10 for x86_64 - BaseOS (RPMs)	x86_64 <*target_os_version*>
	AppStream	**rhel-10-for-x86_64-appstream-rpms**	Red Hat Enterprise Linux 10 for x86_64 - AppStream (RPMs)	x86_64 <*target_os_version*>
64-bit ARM	BaseOS	**rhel-10-for-aarch64-baseos-rpms**	Red Hat Enterprise Linux 10 for ARM 64 - BaseOS (RPMs)	aarch64 <*target_os_version*>
	AppStream	**rhel-10-for-aarch64-appstream-rpms**	Red Hat Enterprise Linux 10 for ARM 64 - AppStream (RPMs)	aarch64 <*target_os_version*>
IBM Power (little endian)	BaseOS	**rhel-10-for-ppc64le-baseos-rpms**	Red Hat Enterprise Linux 10 for Power, little endian - BaseOS (RPMs)	ppc64le <*target_os_version*>

Architecture	Repository	Repository ID	Repository name	Release version
	AppStream	**rhel-10-for-ppc64le-appstream-rpms**	Red Hat Enterprise Linux 10 for Power, little endian – AppStream (RPMs)	ppc64le *<target_os_version>*
IBM Z	BaseOS	**rhel-10-for-s390x-baseos-rpms**	Red Hat Enterprise Linux 10 for IBM z Systems – BaseOS (RPMs)	s390x *<target_os_version>*
	AppStream	**rhel-10-for-s390x-appstream-rpms**	Red Hat Enterprise Linux 10 for IBM z Systems – AppStream (RPMs)	s390x *<target_os_version>*

Replace *<target_os_version>* with the target OS version, for example **10.0**.