rtbrick

Technical Documentation

RBFS Installation and Licensing

Version 26.3.1 : latest, 14 October 2026

Table of Contents

1. RBFS Installation Overview . . . 2
   1.1. Overview . . . 2
   1.2. RBFS Installation Methods . . . 2
      1.2.1. ONIE-Based Installation Methods . . . 3
      1.2.2. RBFS-Based Installation Methods . . . 4
      1.2.3. Installation Tools and Components . . . 4
      1.2.4. Installation Guidelines . . . 9
      1.2.5. Self-Service Portal Sign-Up / Sign-In . . . 10
      1.2.6. Using the Self-Service Portal . . . 12
   1.3. rtb-buddy: Simplifying JWKS, JWT, and mTLS Workflows . . . 13
      1.3.1. Overview . . . 13
      1.3.2. Installation . . . 13
      1.3.3. Usage . . . 13
      1.3.4. Shell Completion . . . 14
      1.3.5. Commands . . . 14
2. RBFS Image Download . . . 20
   2.1. Generating a Client Certificate . . . 20
      2.1.1. About the RtBrick APT Tool (rtb-apt) . . . 20
      2.1.2. Installing the rtb-apt Tool . . . 21
      2.1.3. Generating a TLS Client Certificate . . . 22
   2.2. Uploading the Client Certificate to the Self-Service Portal . . . 22
   2.3. Obtaining Approval and verification of the Client Certificate . . . 24
   2.4. Identifying and Activating the Image Repository . . . 24
      2.4.1. Finding the Image Repositories . . . 24
      2.4.2. Activating an Image Repository . . . 25
      2.4.3. Verifying the Active Repositories . . . 25
   2.5. Installing the rtb-image Tool and Verifying Access to Image Stores . . . 25
      2.5.1. Installing the rtbrick-imgstore Package . . . 25
      2.5.2. Verifying Access (Authentication) to Image Stores . . . 26
   2.6. Downloading the Host Image . . . 27
      2.6.1. Updating the Local Cached Copy of the Remote Image Store . . . 28
      2.6.2. Finding the Host Image . . . 28
      2.6.3. Downloading the Host Image . . . 29
      2.6.4. Displaying the Location of the Downloaded Image . . . 30
3. RBFS Manual Installation . . . 32
   3.1. RBFS Software Upgrade . . . 32
   3.2. Prerequisites for Manual Installation . . . 32
   3.3. ONIE-Based Installation Methods . . . 32

3.3.1. Zero Touch Provisioning (ZTP) . . . 33
3.3.2. Manual Installation Methods . . . 33
3.4. RBFS-Based Installations . . . 37
3.4.1. RtBrick Host System Commands for Installation Operations . . . 37
3.4.2. RBFS Commands for Installation Operations . . . 42
3.4.3. RBFS APIs for Installation Tasks . . . 44
3.5. Operational Commands . . . 44
3.5.1. software partition . . . 44
3.5.2. Job Data . . . 47
3.5.3. Job Status . . . 48
4. RBFS Automated Installation (ZTP) . . . 49
4.1. Overview . . . 49
4.2. ZTP Workflow . . . 49
4.2.1. ZTP Process . . . 50
4.3. DHCP Service . . . 52
4.4. HTTP Service (Management Server) . . . 53
4.4.1. ZTP installation . . . 53
4.4.2. ZTP configuration . . . 54
4.5. Control Daemon . . . 55
4.5.1. Trigger the ZTP process . . . 56
4.5.2. Trigger the reinstall . . . 56
4.5.3. Trigger Firmware Update . . . 56
4.5.4. Management Server URL Discovery . . . 56
4.5.5. Request configurations . . . 57
4.5.6. Business Events . . . 57
4.5.7. Overall Process Flow . . . 57
4.6. References . . . 59
5. RBFS Licensing . . . 60
5.1. Overview . . . 60
5.2. Obtaining or Extending Licenses . . . 60
5.3. Managing Licenses via Self-Service Portal . . . 60
5.3.1. Accessing the license key . . . 60
5.3.2. Working with the license list view . . . 61
5.3.3. Requesting a new license . . . 61
5.3.4. Renewing an existing license . . . 62
5.4. Installing a License . . . 63
5.5. Installing Multiple Licenses . . . 64
5.6. Viewing the installed license . . . 64
5.7. Deleting a License . . . 65
5.8. License Expiry . . . 65
5.8.1. Restricted Configuration Commits after License Expiry . . . 65

5.8.2. License Validation 65
5.8.3. Operational Commands to View Active Licenses 66

Get started with deploying RBFS on supported hardware platforms. This guide walks you through the entire installation lifecycle — from securely downloading RBFS images and preparing your environment, to installing RBFS via manual or Zero Touch Provisioning methods, and managing software licenses. You will also gain insights into the image partitioning, upgrades and rollbacks, efficient image management with RtBrick tools, and streamlined license management through the Self-Service Portal.

The following guides are available under the Installation and Licensing section.

- RBFS Installation
- Licensing

1. RBFS Installation Overview

1.1. Overview

RBFS software images are hosted in the RtBrick Image Store, where users can download and install role-specific images on supported hardware platforms. The Image Store provides the latest RBFS releases. For a complete list of supported hardware platforms, see the Supported Platforms section in the Platform Guide.

RBFS is also be delivered as a virtual machine image with limited functionalities for testing purposes. This document does not include information related to RBFS VM image deployments. For information about RBFS VM image deployments, see RBFS VM Image.

1.2. RBFS Installation Methods

RBFS supports multiple installation methods to accommodate different deployment requirements. Choose the method that best fits your system environment. You can install RBFS using either following manual installation steps or through Zero Touch Provisioning (ZTP) on a bare-metal switch. ONIE serves as the foundation for initial installation methods and provides a standardized environment for deploying RBFS.

This document provide information about the various RBFS installation methods. At a high level, RBFS supports two installation approaches. ONIE-based installation for initial installation on a bare-metal system, and RBFS-based installation for subsequent installation and upgrade operations. RBFS-based installation methods can be used only if RBFS is already present on the system.

RBFS supports software reinstallations, upgrades, and other software installation tasks using CLI commands after the initial ONIE-based installation. While the ONIE-based installation methods can technically be used for reinstallations and upgrades, it is recommended to use the RBFS-based methods for their ease of use.

The following diagram provides a high-level overview of the RBFS installation methods.

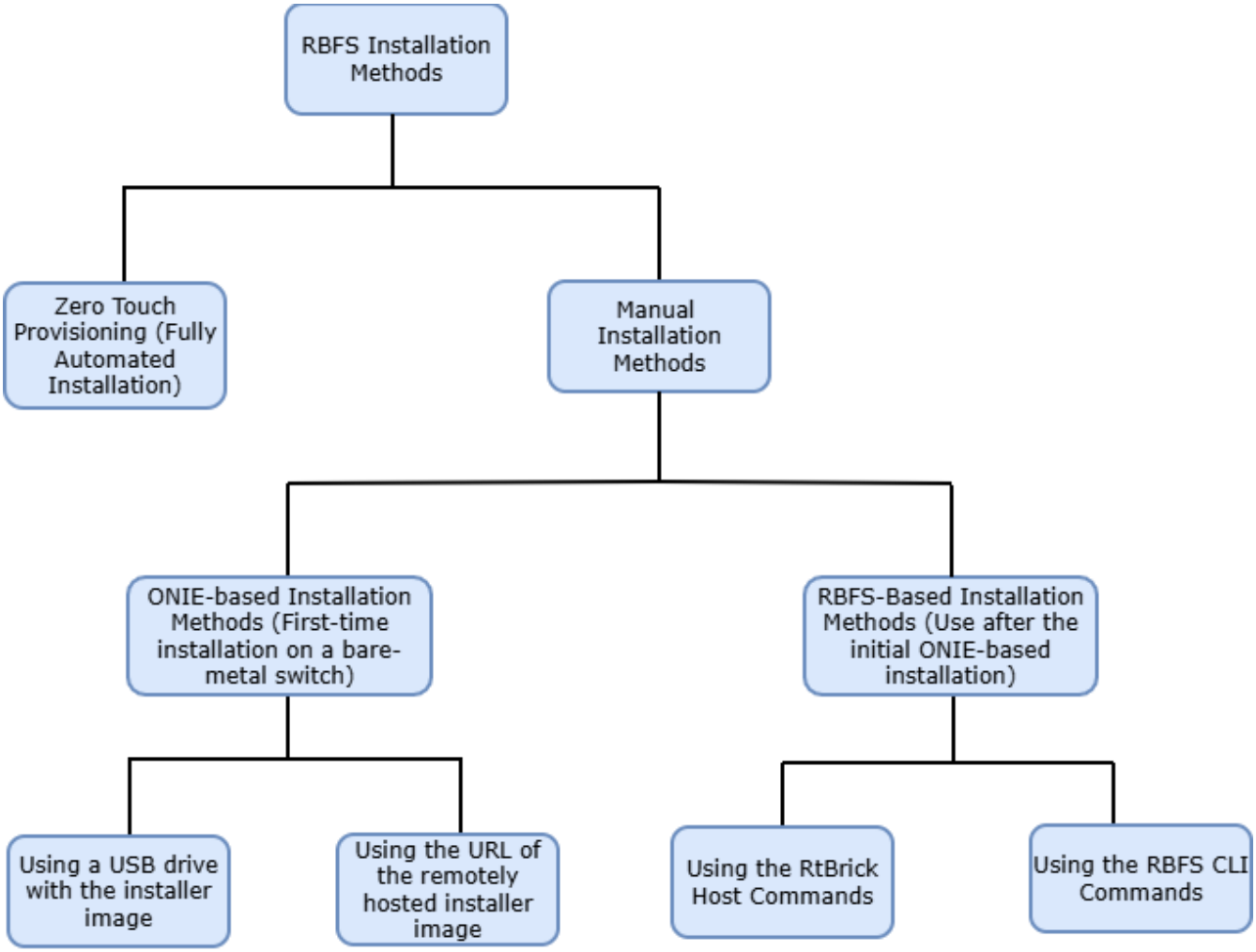


Figure 1. RBFS Installation methods high level view

1.2.1. ONIE-Based Installation Methods

The ONIE-based installation methods include:

Zero Touch Provisioning: This is a fully automated installation method that enables a device to fetch installation instructions and install RBFS without manual intervention. For information on the automated installation process, see section RBFS Automated Installation (Zero Touch Provisioning).

Manual Installation: Manual installation includes two different methods:

- **Using a USB drive with the installer image**: Install RBFS using a USB drive containing the RBFS installer image. This method is used when network-based installation is not available. For detailed step-by-step instructions on the installation using a USB thumb drive, see section Installing RBFS Using a USB Thumb Drive.
- **Using the URL of the remotely hosted installer image**: Install RBFS by providing an HTTP URL to the RBFS installer image. ONIE downloads the image from the specified URL and installs it on the device. For detailed step-by-step instructions on the installation using the URL of the remotely hosted installer image, see section Using the URL of the remotely hosted installer image.

1.2.2. RBFS-Based Installation Methods

RBFS provides CLI commands for subsequent installation operations and software upgrades after the initial RBFS provisioning. These CLI commands enable installation, upgrades and recovery tasks with minimal impact to the running software. The commands include RtBrick Host system commands and RBFS CLI commands to perform software installation tasks such as downloading images, installing software on the alternate image partition, deleting existing images, and maintaining multiple software versions.

RtBrick Host Commands

You can use the Host system commands to manage RBFS images and boot behavior. These commands allow you to reboot the system using a specific image partition (Image A or Image B), permanently set the default boot target, and control whether the system reboots immediately after changing the boot configuration. The dual-image partition model enables you to install and maintain two different RBFS software versions—one on Image A and another on Image B—allowing safe upgrades, rollbacks, and version testing.

RBFS CLI Commands

You can use a set of CLI commands to manage software images and partitions on the device. These commands enable you to download software images, delete existing images, install software on a specified partition, and mark a partition as the active boot target.

RBFS also provides show commands to display software partition information and track the execution of software management jobs. You can view the list of software partitions and their installed images, monitor real-time job logs, and check the final execution status of software jobs using the show commands.

For information about the CLI commands, see the section RBFS-Based Installation Methods.

1.2.3. Installation Tools and Components

It is essential to familiarize with the following components before beginning the RBFS Image Download process.

Downloading RBFS Image

Before you start the installation process, download the RtBrick Host image. For details on downloading the RtBrick Host image, see the RBFS Image Download section.

RtBrick Image Store

RBFS software images are stored in the RtBrick Image Store and can be downloaded after providing the required certificate.

Image stores containing the Host installer images are published on https://releases.rtbrick.com/ and updated when new image versions are available.

The `rtb-image` command (CLI tool) provided by the rtbrick-imgstore package is used to interact with

"image stores".

Access to the Image Store and Debian package repositories on https://releases.rtbrick.com/ is secured using mutual TLS (mTLS). Authentication requires a valid TLS client certificate.

RBFS Host Image

The RBFS software (NOS) available on the RtBrick Image Store is provided as the RBFS Host installer image for installation on qualified OCP-compliant switches.

RtBrick Tools

In addition to RBFS software, other RtBrick software tools are delivered in Debian package format compatible with Debian 12 (Bookworm). The software delivered as Debian 12 packages includes a set of CLI tools and/or daemons designed to facilitate working with RBFS containers and the RBFS API. Debian package repositories containing these packages are available at /https://releases.rtbrick.com/ and are updated whenever a new version becomes available.

Open Network Install Environment

The Open Network Install Environment (ONIE) comes pre-installed on OCP-compliant switches. The ONIE environment is used for installing the RtBrick Host installer image. It provides an environment for installing the RBFS software to run on those switches. For more details about ONIE, see https://opencomputeproject.github.io/onie/.

RBFS Release Versioning

An RBFS release can be defined as a set of software packages (currently, in the Debian package format). However, it is delivered as an image, either a container image or as a complete Host installation image. The Host installation image may or may not contain a container image pre-installed in it. An image can be defined as the archived root file system of a Linux OS installation with the needed software packages pre-installed and with a default configuration. In the current context, the terms 'RBFS release' and 'image' are used interchangeably.

RBFS uses the following versioning format:

<year>.<release>.<minor>[.<fix>][-<label>

Examples:

24.3.1
24.3.1.1

In the version example 24.3.1, the first number, "24," represents the year 2024. The second number, "3," indicates the release version, where "1" corresponds to the first release of the year, and this number will be incremented with each subsequent release. The third number, "1," denotes the minor release, which will also be incremented with each future minor release.

RtBrick also uses a four-number versioning format, represented as 24.3.1.1. In this format, the

fourth number indicates the bug-fix release. Bug-fix releases are delivered only when necessary and are based on an existing RBFS release, such as 24.3.1. The bug-fix release numbers will also be incremented with each subsequent minor release.

Candidate releases will use a label such as "candidate.6", which will be incremented with each subsequent candidate release.

Image Formats

RtBrick images delivered through the RtBrick Image Store and the `rtb-image` utility have the following attributes:

- `format`: This is the file format in which the image is packaged and archived. The available format is `host-installer`.
- `role`: The role inside a network of the device which will be running the image. For example, `multiservice-edge` signifies the full BNG functionality on a single image.
- `platform`: Identifies the hardware platform in which the image can run. For example, `q2a` signifies the switch ASIC Broadcom Qumran-2A.
- `model`: Identifies the hardware model. For example, `s9510-28dc` signifies the hardware model UfiSpace S9510-28DC.
- `ver-range`: Identifies the image version. For example, 24.8.1 signifies the RBFS release 24.8.1.

RtBrick images intended to be installed on supported hardware devices contain `format`, `platform`, and `model` set accordingly to the specific switching hardware.

You can see this using `sudo rtb-image list` command and look for the `Format` column.

Image Partitions

An image partition is a dedicated storage area on the device where a specific version of the system software image is installed. RBFS Image partition allows the device to install multiple software images (one RBFS image per partition, for example, Image-A and Image-B). Image Partition provides an isolated environment for installing and operating software images without affecting the active system. It allows to install a new image in a partition, and if the new image fails, the system can rollback to the previous functioning image.

Each partition can boot independently. The system can switch between the partitions, if required.

The Image Partition feature applies exclusively to RBFS image partitions. It does not apply to other partitions, such as ONIE or boot partitions.

This image partition layout describes how RBFS organizes the storage into logical partitions, each serving a specific purpose.

Name	Mount Point	Survives Reinstall	Description
RTB-IMAGE-A	/ (root)	No	One of the two bootable partition images. This partition contains the OS files and binaries for Image A. During a reinstall or software upgrade, this partition will be overwritten.
RTB-IMAGE-B	/ (root)	No	The alternate bootable system image. Similar to IMAGE-A, it holds OS files and is also overwritten during reinstall. The device usually boots into one of these partitions.
RTB-CONFIG	/var/config	Yes	Stores persistent configuration files such as user settings and system configuration. This partition survives reinstallation and the settings are preserved across upgrades.
RTB-LOG	/var/log	No	Intended to store system logs.
RTB-CRASH	/var/crash	No	Stores crash data (core dumps).

Firmware Upgrade Using Host-Installer

For details about the firmware version supported by a specific platform, see the Firmware Versions section in the Platform Guide.

RBFS allows you to perform firmware upgrades directly on the device by executing the CLI commands. RBFS includes the firmware updater files within the host-installer image for each supported device model. It allows you to perform firmware upgrades directly on the device using CLI commands whenever required.

It ensures that device-specific firmware is readily available for reliable upgrades. Since the firmware is already bundled as part of the image, no external components are needed to initiate the upgrade process.

You can perform firmware upgrades directly on the device by executing the CLI commands. The system includes the firmware updater files within the host installer image for each supported device model.

The firmware upgrade is performed using the ONIE firmware package.

In some cases, platform-level requirements for firmware upgrades may require a complete reinstallation of the RBFS software. It is recommended to schedule these updates within a maintenance window to ensure the environment can be completely restored if needed.

You must run the following commands on the Host System to locally stage and verify a firmware upgrade using the ONIE firmware package.

1. On the Host system, navigate to the firmware directory using the `cd /var/rtbrick/firmware/`

command.

Example:

```
supervisor@host>rtbrick.net:~ $ cd /var/rtbrick/firmware/
```

2. List and verify the firmware files available in the directory using the `ls -l` command. Ensure that the firmware updater file (for example, `onie-firmware-···updater` and `multi_updater.json`) is present in the directory.

Example:

```
supervisor@host>rtbrick.net:/var/rtbrick/firmware $ ls -l
total 57088
-rwxr--r-- 1 root root      110 Mar 18 17:30 multi_updater.json
-rwxr--r-- 1 root root 58451962 Mar 18 17:30 onie-firmware-x86_64-
ufispace_skl_ncplite_series-r0_v3.3.0.updater
```

3. Stage the firmware update using the command `sudo /mnt/onie-boot/onie/tools/bin/onie-fwpkg add <firmware-updater-file>`

Example:

```
supervisor@host>rtbrick.net:/var/rtbrick/firmware $ sudo /mnt/onie-
boot/onie/tools/bin/onie-fwpkg add onie-firmware-x86_64-ufispace_skl_ncplite_series-
r0_v3.3.0.updater
[sudo] password for supervisor:
Staging firmware update: onie-firmware-x86_64-ufispace_skl_ncplite_series-
r0_v3.3.0.updater
```

4. Verify the staged firmware using the `sudo /mnt/onie-boot/onie/tools/bin/onie-fwpkg` command. It displays the current firmware package status, including the staged update. It confirms that the firmware package has been successfully added.

Example:

```
supervisor@host>rtbrick.net:/var/rtbrick/firmware $ sudo /mnt/onie-
boot/onie/tools/bin/onie-fwpkg
** Pending firmware update information:
Name                                                                | Version |
Attempts |Size (Bytes)  | Date
=====================================================================+=========+========
==+==============+====================
onie-firmware-x86_64-ufispace_skl_ncplite_series-r0_v3.3.0.updater | v3.3.0  |
0 |     58451962 | 2026-03-20 04:52:43
=====================================================================+=========+========
```

```
==+==============+====================
```

For more detailed information, run the `onie-fwpkg show-pending <update_name>` command.

Example:

```
** Firmware update results information:
Name                                                                | Version    |
Result   | Date
====================================================================+============+=====
=====+====================
onie-firmware-updater                                               | v3.3.0     |
Success  | 2025-10-03 17:35:12
onie-firmware-x86_64-ufispace_skl_ncplite_series-r0_v2.2.2.updater | v2.2.2     |
Success  | 2023-11-16 09:25:50
onie-updater                                                        | 2020.02v01 |
Success  | 2000-01-01 00:01:05
onie-updater-x86_64-ufispace_s9600_72xc-r0_2020.02v01               | 2020.02v01 |
Success  | 2000-01-01 00:49:05
====================================================================+============+=====
=====+====================
```

5. Run the `sudo rtb-reboot update` command to trigger the firmware upgrade.
6. Reboot the Host using the `sudo reboot` command.
7. After reboot, verify the firmware version using the `show firmware` command within the container
8. Perform a power-cycle operation after the firmware upgrade. For instructions, see API Endpoint to Perform Power-Cycle Operation.

1.2.4. Installation Guidelines

The following high-level installation guidelines apply to all types of RBFS installations. Perform these steps in the specified order to prepare the system for RBFS deployment.

1. **Power-cycle the system:** Perform a power-cycle operation before starting the installation. For instructions, see API Endpoint to Perform Power-Cycle Operation.
2. **Upgrade the firmware:** Upgrade the firmware to the recommended version for the platform. For the supported versions, see Recommended Firmware Versions and to know how to perform firmware upgrade, see Firmware Upgrade Using Host-Installer.

After upgrading the firmware, you must perform another power-cycle operation before proceeding.

3. **Load the RBFS image:** Install the appropriate RBFS image by following the procedure for the applicable installation type.

4. **Verify the RBFS license status:** After installing RBFS, verify the license status displayed in the system banner to ensure that the license is valid.

1.2.5. Self-Service Portal Sign-Up / Sign-In

RtBrick customers use the self-service portal to request access to the RBFS image download servers and request RBFS licenses. Every user of the Self-Service portal is associated with a specific organization, which is determined by the domain of their company email address. For example, all users with an email address under the domain @rtbrick.com are affiliated with RtBrick. If your email domain is not registered with RtBrick, contact RtBrick Support for assistance.

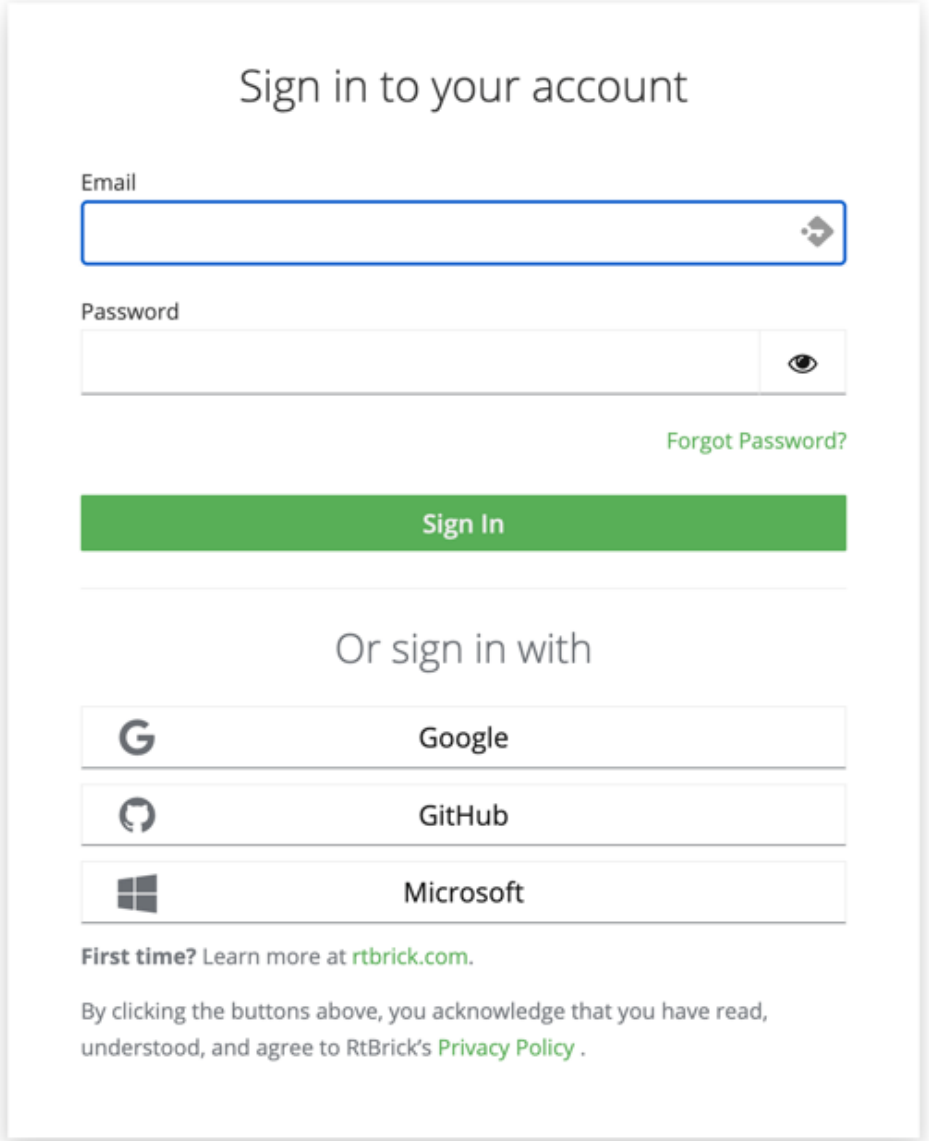
RtBrick supports a single sign-on (SSO) mechanism that authenticates users across Freshdesk, Self-Service Portal, and `auth.rtbrick.com` identity management system, forming a unified SSO domain. All users with access to RtBrick services are managed centrally within `auth.rtbrick.com`.

RtBrick Authentication Method

RtBrick provides its own authentication service with RtBrick Identity Management System through `auth.rtbrick.com`, allowing you to sign in to the Self-Service Portal using their RtBrick account.

To sign in using an RtBrick account:

1. Open the Self-Service Portal.
2. Select Sign in with RtBrick (or enter RtBrick credentials when prompted). You are redirected to `auth.rtbrick.com` for authentication.
3. Enter the RtBrick email address and password. After successful authentication, you are redirected back to the Self-Service Portal.



If you do not already have a RtBrick account, click Sign Up on the authentication page and complete the registration process before signing in.

Third Party Authentication Methods

In addition, RtBrick supports third party authentication services. The Self-Service portal uses OpenID/Connect to delegate user authentication to third-party authorization services. These authorization services ensure the secure storage of user credentials and provide additional security measures, including two-factor authentication and account recovery options for users who may have forgotten their passwords.

If you do not have an account with any of the three supported third-party authorization services, create an account with RtBrick's identity management system using `auth.rtbrick.com`.

The portal supports the following authentication service providers:

- GitHub
- Google

- Microsoft

Selecting Google, GitHub, or Microsoft redirects you to the respective third-party provider to authenticate the credentials.

When a user logs into the portal for the first time, their membership is created. The member will be assigned to an organization based on the domain of their email address. This domain must be a trusted domain, meaning it should be listed in the trusted domains list of exactly one organization.

Attempts to sign up / sign in to the portal with an email address of an untrusted domain will be rejected.

GitHub

GitHub allows users to create new accounts for free. A user must declare their company email address as the public email in their GitHub profile to enable the portal to read the email address during the OpenID/Connect authentication process.

A user cannot sign up / sign in to the portal if the portal is not allowed to read the user's email address.

Google

The user must confirm that the portal has permission to access the email address from their user profile for the sign-up process. After the initial sign-up, subsequent logins will not require the user to grant access to their profile again.

Microsoft

Microsoft allows domain administrators to decide which sites can delegate authentication to the Microsoft's OpenID/Connect authorization services. This adds an additional level of security, because a user can not accidentally share profile data with an untrusted site.

A user will only be prompted for granting the portal access to its profile if the domain administrator has allowed the portal to delegate the login to Microsoft for its organization. In case the portal is not allowed to delegate authentication to Microsoft for the paritcular organization, the user attempting to sign-up to the portal will be prompted to request a domain administrator to grant the portal access to Microsoft authentication services.

In large enterprises with strict security processes granting the portal access to Microsoft authentication service might take a considerable amount of time. An alternative would be to create a GitHub account.

1.2.6. Using the Self-Service Portal

The Self-Service Portal can be used for generating and uploading client certificates. Also, it is required for obtaining new RBFS licenses or extend the existing licenses. For more information, see the Uploading the Certificate to the Self-Service Portal section of the RBFS Image Download Guide and Managing Licenses via Self-Service Portal section of the RBFS Licensing Guide.

1.3. rtb-buddy: Simplifying JWKS, JWT, and mTLS Workflows

1.3.1. Overview

rtb-buddy is a command-line tool for the RtBrick Full Stack Network Operating System (RBFS). It generates APIGW and LI X1 cryptographic artifacts without relying on external web pages. The tool streamlines a single CLI workflow for creating JWKS files, signed JWT tokens, and mutual TLS (mTLS) certificates.

1.3.2. Installation

Quick Install

curl installer

```
curl -fsSL https://raw.githubusercontent.com/rtbrick/tools/main/install_rtb_buddy.sh |
sudo bash
```

This detects your OS and architecture, downloads the latest release from GitHub, verifies the checksum, and installs rtb-buddy to /usr/local/bin.

From Source

Requires Go 1.27+

```
git clone https://github.com/rtbrick/tools.git
cd tools
make build
```

The binary `rtb-buddy` is placed in the project root.

Go Install

```
go install github.com/rtbrick/tools/cmd/rtb-buddy@latest
```

The binary is placed in `$(go env GOPATH)/bin`.

1.3.3. Usage

```
# Generate key pair
rtb-buddy apigw generate jwks

# Issue a token
rtb-buddy apigw generate token --sub alice --name Alice --preferred-username alice
--scope supervisor
```

```
# Verify a token
echo "<token>" | rtb-buddy apigw inspect token
```

1.3.4. Shell Completion

```
# zsh (current session)
source <(rtb-buddy completion zsh)

# zsh (persistent - Linux)
rtb-buddy completion zsh > "${fpath[1]}/_rtb-buddy"

# zsh (persistent - macOS)
rtb-buddy completion zsh > "$(brew --prefix)/share/zsh/site-functions/_rtb-buddy"

# bash
source <(rtb-buddy completion bash)
```

1.3.5. Commands

apigw

APIGWD related commands.

apigw generate jwks

Generates an RSA key pair as two JWKS files.

- `--priv-key` — output path for the private JWKS (default: `apigw-jwks-priv.json`)
- `--pub-key` — output path for the public JWKS (default: `apigw-jwks.json`)
- `--kid` — key ID for the JWKS entries (default: `access`)

The private JWKS contains two entries (private key + public key, same KID). The public JWKS contains one entry (public key only, for distribution).

```
rtb-buddy apigw generate jwks
rtb-buddy apigw generate jwks --kid "prod" --priv-key prod-jwks-priv.json --pub-key
prod-jwks.json
```

apigw generate token

Generates a signed JWT. The private key is read from the JWKS file.

- `--priv-key` — private JWKS file path (default: `apigw-jwks-priv.json`)* `--kid` — key ID to select from the JWKS (empty = first key)
- `--override-kid` — override the kid in the JWT header (default: use the JWKS entry's kid)

- `--sub` — token subject (required)
- `--name` — name claim (required)
- `--preferred-username` — preferred username claim (required)
- `--scope` — scope claim (default: `supervisor`)
- `--iss` — issuer (default: `rtbrick`)
- `--dur` — token duration (default: `30s`)

```
# Single key in JWKS — --kid not needed
rtb-buddy apigw generate token \
  --sub 1234567890 \
  --name rtbrick \
  --preferred-username rtbrick \
  --scope supervisor

# Multiple keys — select by KID
rtb-buddy apigw generate token \
  --kid "prod" \
  --sub 1234567890 \
  --name rtbrick \
  --preferred-username rtbrick

# Override the kid in the JWT header
rtb-buddy apigw generate token \
  --kid "access" \
  --override-kid "rotated" \
  --sub 1234567890 \
  --name rtbrick \
  --preferred-username rtbrick
```

apigw generate tls

```
rtb-buddy apigw generate tls --org "rtbrick"
```

apigw inspect token

Decode a JWT from stdin. Verify signature if public JWKS is available. Reports expiration status.

- `--pub-key` — public JWKS file path for signature verification (default: `apigw-jwks.json`)
- `--override-kid` — override the kid for key lookup in the JWKS

```
# Verify with matching kid
rtb-buddy apigw generate token --kid access --sub 123 --name admin --preferred
-username admin \
  | rtb-buddy apigw inspect token
```

```
# Override kid for key lookup (e.g. token has kid "test", JWKS has kid "access")
rtb-buddy apigw generate token --kid access --override-kid test --sub 123 --name admin
--preferred-username admin \
  | rtb-buddy apigw inspect token --override-kid access
```

lix1

LI X1 mTLS certificate management commands for Lawful Intercept.

mTLS Setup

LI X1 uses mutual TLS (mTLS) between ADMF (Administrative Function, the LI management system) and RBFS (network element, the network device).

CA

Trust anchor — issues certificates for all components

RBFS

Server — presents certificate, verifies ADMF clients

ADMF

Client — presents certificate, verifies RBFS server

Generating the CA and certificates

```
# 1. Generate CA
rtb-buddy lix1 generate ca --cn "LI X1 CA" --org "RtBrick" --ou "Engineering"

# 2. Generate RBFS certificate (server)
rtb-buddy lix1 generate cert --cn "rbfs-router-01" --org "RtBrick" --ou "Engineering"
--crt rbfs.crt --crt-key rbfs.key

# 3. Generate ADMF certificate (client)
rtb-buddy lix1 generate cert --cn "admf-prod" --org "RtBrick" --ou "Engineering" --crt
admf.crt --crt-key admf.key
```

Deploy

- **RBFS**: `rbfs.crt` + `rbfs.key` + `ca.crt`
- **ADMF**: `admf.crt` + `admf.key` + `ca.crt`

Both sides need `ca.crt` to verify each other's certificates.

lix1 generate ca

Generates a root CA certificate and private key.

- `--ca-crt` — CA certificate output path (default: `ca.crt`)

- `--ca-crt-key` — CA private key output path (default: `ca.key`)
- `--cn` — Common Name (required)
- `--org` — Organization (required)
- `--ou` — Organizational Unit (required)
- `--duration` — Certificate validity duration (default: `87600h` / 10 years)

```
rtb-buddy lix1 generate ca --cn "RtBrick LIX1" --org "RtBrick" --ou "Engineering"
rtb-buddy lix1 generate ca --cn "My CA" --org "My Org" --ou "Eng" --ca-crt my-ca.crt
--ca-crt-key my-ca.key
```

lix1 generate cert

Generates a certificate signed by the CA.

- `--ca-crt` — CA certificate input path (default: `ca.crt`)
- `--ca-crt-key` — CA private key input path (default: `ca.key`)
- `--crt` — Certificate output path (required)
- `--crt-key` — Private key output path (required)
- `--cn` — Common Name (required)
- `--org` — Organization (required)
- `--ou` — Organizational Unit (required)
- `--duration` — Certificate validity duration (default: `87600h` / 10 years)

```
rtb-buddy lix1 generate cert --cn "Test Device 1" --org "RtBrick" --ou "Testing" --crt
device_test_1.crt --crt-key device_test_1.key
rtb-buddy lix1 generate cert --cn "ADMF" --org "RtBrick" --ou "Engineering" --crt
admf.crt --crt-key admf.key
rtb-buddy lix1 generate cert --cn "RBFS" --org "RtBrick" --ou "Engineering" --crt
rbfs.crt --crt-key rbfs.key
```

lix1 test cert

Validates the certificate and it’s trust chain.

- `--crt` — Certificate to test (required)
- `--ca-crt` — CA certificate for chain validation (default: `ca.crt`)

```
rtb-buddy lix1 test cert --crt device_test_1.crt
rtb-buddy lix1 test cert --crt admf.crt --ca-crt ca.crt
```

Example output:

```
Subject: CN=Test Device 1,OU=Testing,O=RtBrick
Issuer: CN=RtBrick LIX1,OU=Engineering,O=RtBrick
Serial: 1658
Signature Algorithm: SHA256-RSA
Not Before: 2026-09-16T17:11:06Z
Not After: 2036-09-13T17:11:06Z
Is CA: false

Status: VALID (expires in 87600h0m0s)

Verifying certificate chain against CA...
Chain verification: OK
```

rbfs

RBFS router management commands.

rbfs config restconf

RESTCONF operations for RBFS routers.

rbfs config restconf get

Get RESTCONF configuration from an RBFS router.

- `--auth-mode` — Use HTTPS with JWT authentication (port 12321); default is HTTP (port 19091)
- `--element` — Element name (defaults to basepath)
- `-o, --output` — Output file path (default: print to stdout)
- `-p, --path` — RESTCONF data path (e.g. `rtbrick-config:lldp`, default: full config)
- `--priv-key` — Private JWKS file path for token generation (default: `apigw-jwks-priv.json`)
- `--kid` — Key ID to select from the JWKS (default: `access`)
- `--sub` — Token subject (default: `78228339`)
- `--name` — Token name claim (default: `rtb-buddy`)
- `--scope` — Token scope (default: `supervisor`)
- `--iss` — Token issuer (default: `rtbrick`)
- `--dur` — Token duration (default: `30s`)

```
# Get full config to stdout
rtb-buddy rbfs config restconf get ufi29.q2c.u35.r3.nbg.rtbrick.net
# Get specific path to stdout
rtb-buddy rbfs config restconf get ufi29.q2c.u35.r3.nbg.rtbrick.net -p rtbrick-
config:lldp
# Get specific path to file
rtb-buddy rbfs config restconf get ufi29.q2c.u35.r3.nbg.rtbrick.net -p rtbrick-
```

```
config:system -o system.json
# HTTPS with JWT
rtb-buddy rbfs config restconf get ufi29.q2c.u35.r3.nbg.rtbrick.net --auth-mode -p
rtbrick-config:system
```

rbfs config restconf put

Put RESTCONF configuration to an RBFS router.

- `--auth-mode` — Use HTTPS with JWT authentication (port 12321); default is HTTP (port 19091)
- `--element` — Element name (defaults to basepath)
- `-f, --file` — Input file path (default: stdin)
- `-p, --path` — RESTCONF data path (e.g. `rtbrick-config:system`)
- `--priv-key` — Private JWKS file path for token generation (default: `apigw-jwks-priv.json`)
- `--kid` — Key ID to select from the JWKS (default: `access`)
- `--sub` — Token subject (default: `78228339`)
- `--name` — Token name claim (default: `rtb-buddy`)
- `--scope` — Token scope (default: `supervisor`)
- `--iss` — Token issuer (default: `rtbrick`)
- `--dur` — Token duration (default: `30s`)

```
# Put config from stdin
cat config.json | rtb-buddy rbfs config restconf put ufi29.q2c.u35.r3.nbg.rtbrick.net
-p rtbrick-config:system
# Put config from file
rtb-buddy rbfs config restconf put ufi29.q2c.u35.r3.nbg.rtbrick.net -p rtbrick-
config:system -f config.json
# HTTPS with JWT
rtb-buddy rbfs config restconf put ufi29.q2c.u35.r3.nbg.rtbrick.net --auth-mode -p
rtbrick-config:system -f config.json
```

2. RBFS Image Download

The RtBrick image download functionality enables authenticated users to download and install the RtBrick software (packages or images). Access to *image stores* and *Debian package repositories* on **/https://releases.rtbrick.com/** is **restricted** through the use of mutual TLS authentication with TLS client certificates (TLS client certificates can be self-signed).

The diagram below provides an overview of the RBFS software download process.

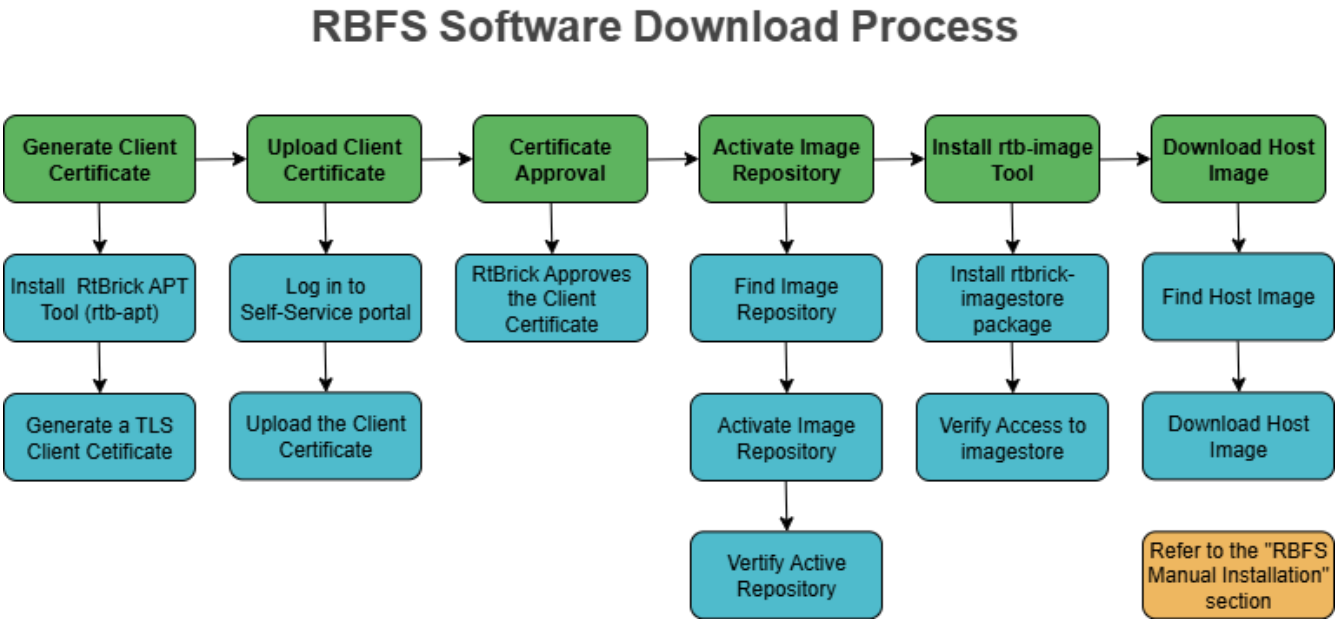


Figure 2. RBFS Software Download Process

The process of downloading software involves the following tasks:

2.1. Generating a Client Certificate
2.2. Uploading the Certificate to the Self-Service Portal
2.3. Obtaining Approval and Verification of Client Certificate
2.4. Identifying and Activating the Image Repository
2.5. Installing the rtb-image Tool and Verifying Access to Image Stores
2.6. Downloading the Host Image

2.1. Generating a Client Certificate

RtBrick provides the `rtb-apt` tool to generate a client certificate. This section contains the following topics:

2.1.1. About the RtBrick APT Tool (rtb-apt)
2.1.2. Installing the rtb-apt Tool
2..3. Generating a TLS Client Certificate

2.1.1. About the RtBrick APT Tool (rtb-apt)

The `rtb-apt` tool is an APT utility application that provides an easier way for managing the system

configuration of RtBrick package repositories which can be used with the usual `apt` commands to install RtBrick software.

Some RtBrick package repositories require authentication via TLS client certificates and the `rtb-apt` tool provides commands for managing those repositories and the required `apt` authentication configuration.

The `rtb-apt` tool is a statically compiled Linux 64-bit executable file. Currently, it is verified to run on **Ubuntu 22.04**.

2.1.2. Installing the rtb-apt Tool

This section contains the following topics:

2.1.2.1. Prerequisites to Install the rtb-apt Tool
2.1.2.2. Downloading and Installing the rtb-apt Tool
2.1.2.3. Verifying the Version of the rtb-apt Tool

Prerequisites to Install the rtb-apt Tool

Before you install `rtb-apt`, ensure that you have installed the following software:

- GNU Privacy Guard (GPG), which is used by `apt` to validate package repositories. To install GPG, enter the following command:

```
sudo apt install gnupg
```

- HTTPS support for `apt` is required to access the package repositories via HTTPS. To do this, enter the following command:

```
sudo apt install apt-transport-https ca-certificates
```

Downloading and Installing the rtb-apt Tool

The following example shows how to download and install the `rtb-apt` tool. It shows the URL where the latest version of the `rtb-apt` tool is available for download:

```
 curl -o /tmp/rtb-apt https://releases.rtbrick.com/_/dl/sw/rtb-
apt/latest/linux_amd64/rtb-apt  \
    && sudo mv /tmp/rtb-apt /usr/local/bin/     \
    && sudo chown root:root /usr/local/bin/rtb-apt  \
    && sudo chmod 0755 /usr/local/bin/rtb-apt
```

Verifying the Version of the rtb-apt Tool

The following example shows the `rtb-apt` tool version. The `rtb-apt` version 2.1.2 or later is required.

```
 rtb-apt --version
2.1.2
```

2.1.3. Generating a TLS Client Certificate

The following example shows how to generate a TLS client certificate using the `rtb-apt` tool.

```
 sudo rtb-apt auth generate
A new self-signed TLS client certificate has been generated for this system:

Subject:     CN=bb59a25d-6b38-4f3c-81e0-065e525c8335,OU=rtb-apt
Valid until: 2024-09-06 10:30:26 +0000 UTC

The following additional auto-generated information is included in the certificate and
can be used to uniquely identify this system:

DNS names:       [hostname.example.net]
Email addresses: [root@hostname.example.net user@hostname.example.net]
< ..................................... >

If you already have a working account on https://portal.rtbrick.com then you can use
the Self-Service section to upload this certificate. If you DO NOT yet have an account
on https://portal.rtbrick.com, send the certificate to your RtBrick support contact:


-----BEGIN CERTIFICATE-----
MIIHHzCCBYegAwIBAgIRAJcI5pqSK9O+g6yJGB15i7YwDQYJKoZIhvcNAQELBQAw
QTEQMA4GA1UECxMHcnRiLWFwdDEtMCsGA1UEAxMkYmI1OWEyNWQtNmIzOC00ZjNj
< ..................................... >
NuLIKfmwrcyXmzAOe1bRtlJrRw0zofxX4rFcMmJReNqOV0obP5r7TCtnWtAqkFx/
7JJa
-----END CERTIFICATE-----
```

After generating the TLS Client Certificate, you need to upload it to the the **Certificates** section on https://portal.rtbrick.com. For details about uploading a certificate, see section Upload the Certificate to the Self-Service Portal below.

2.2. Uploading the Client Certificate to the Self-Service Portal

If your domain is registered with https://portal.rtbrick.com, you will be able to log into your account. If not, reach out to your sales/partner contact to initially have your domain registered with the portal.

To upload a new client certificate, perform the following steps:

1. Log in to Self-Service Portal.

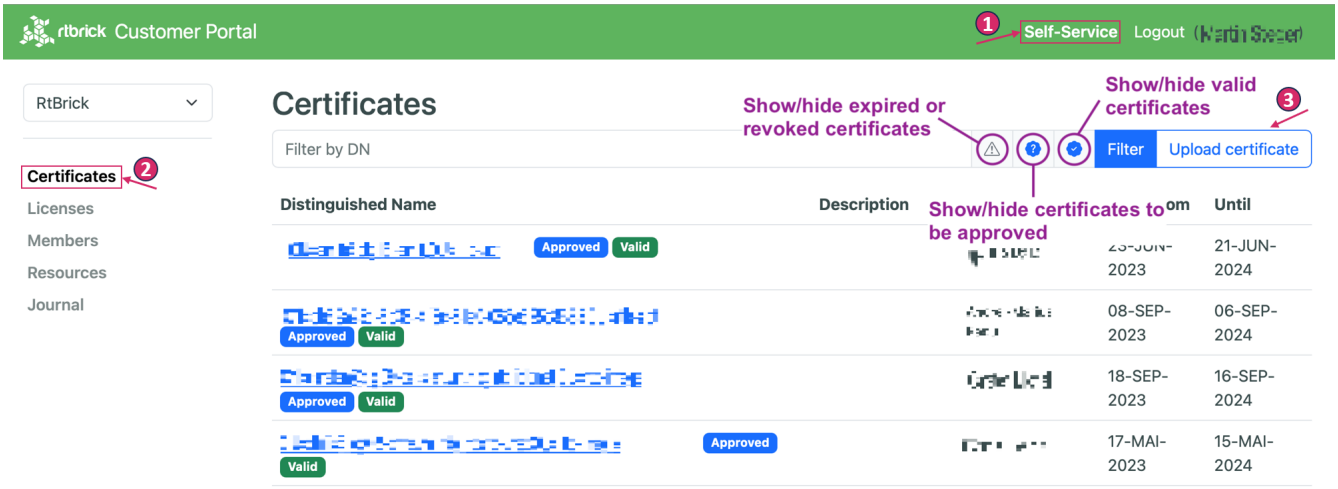


Figure 3. Certificate List

2. Click **Certificates** on the left navigation panel. The Certificates list page appears. The organization's certificate list shows all certificates of that particular organization.

The filter options allows filtering certificates by their distinguished name or lifecycle status.

3. Click the **Upload certificate** button in the organization's certificate list view to upload a new client certificate.

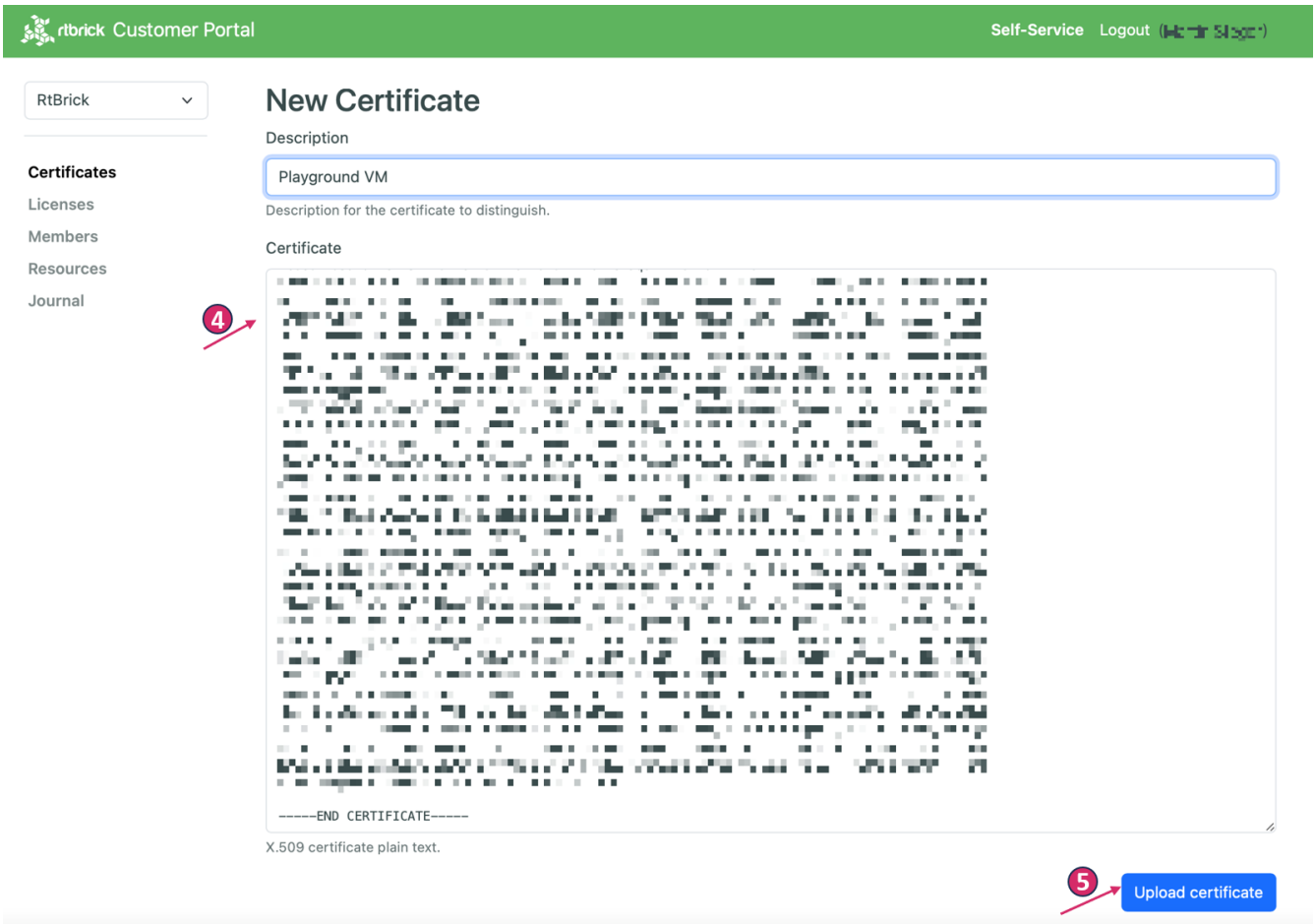


Figure 4. Client Certificate Upload Form

4. Copy the certificate content in PEM format into the text area. The description field is optional,

but it can be used to provide additional information about the certificate.

5. Click the **Upload certificate** button to upload a new certificate.

2.3. Obtaining Approval and verification of the Client Certificate

1. RtBrick reviews and approves the client's certificate that is uploaded on the Self-Service portal.
2. After RtBrick approves the certificate, verify it by entering the command "sudo rtb-apt auth check".

```
 sudo rtb-apt auth check
Repository: releases/latest/rtbrick-tools ... restricted ... TLS client certificate
accepted
```

If the client certificate is not accepted by RtBrick, the following message will appear. Contact the customer support team.

```
 sudo rtb-apt auth check
Repository: releases/latest/rtbrick-tools ... restricted ... TLS client certificate
NOT accepted
```

2.4. Identifying and Activating the Image Repository

You can install additional RtBrick Tools that help simplifying tasks related to debian package repositories. For details see Installing the rtb-image Tool and Verifying Access to Image Stores

This section contains the following topics:

2.4.1. Finding the Image Repository
2.4.2. Activating the Repository
2.4.3. Verifying Active Repositories

2.4.1. Finding the Image Repositories

To find the available repositories, enter the "sudo rtb-apt repo list" command.

The following example shows how to find the available repositories:

```
 sudo rtb-apt repo list
Group              Repository      Distribution    Release Active  Restricted
releases/latest    rtbrick-tools   ubuntu          jammy   No      No
< ..................................... >
```

2.4.2. Activating an Image Repository

To activate an image repository, enter the "sudo rtb-apt repo activate" command.

The following example shows how to activate the "releases/latest/rtbrick-tools" repository.

```
 sudo rtb-apt repo activate releases/latest/rtbrick-tools
```

`rtb-apt` activated repository is added to /etc/apt/sources.list.d/rtbrick.list so that the repository can then be used with commands such as `apt update` and `apt install` to install the RtBrick Debian tool packages.

```
 cat /etc/apt/sources.list.d/rtbrick.list
deb [arch=amd64 signed-by=/etc/rtbrick/RtBrick-Support.pubkey.asc]
https://releases.rtbrick.com/_/latest/ubuntu/jammy/rtbrick-tools    jammy    rtbrick-
tools
```

2.4.3. Verifying the Active Repositories

To verify the active repositories, use the "sudo rtb-apt repo list" command. For example in the below output `releases/latest` repository is active because its status is set to YES.

```
 sudo rtb-apt repo list
Group           Repository      Distribution    Release Active  Restricted
releases/latest rtbrick-tools   ubuntu          jammy   Yes     Yes
< ..................................... >
```

2.5. Installing the rtb-image Tool and Verifying Access to Image Stores

Once the TLS client certificate for the current system is trusted by RtBrick and once RtBrick package repositories have been activated with rtb-apt, the apt commands can be used to install the RtBrick software contained in those package repositories.

`rtb-image` version 3.11.0 or later is required to correctly work with managed downloads.

This section contains the following topics:

2.5.1. Installing the rtbrick-imgstore Package
2.5.2. Verifying access (authentication) to Image Stores

2.5.1. Installing the rtbrick-imgstore Package

If you have any existing RtBrick tools packages, it is essential to upgrade to the

> latest version because some of the RtBrick tools Debian packages have changed and have been upgraded several times. You can remove the exisiting RtBrick tools package using the below command:
> `apt list --installed | egrep -i rtbrick-imgstore | awk -F '/' '{print $1;}' | xargs sudo apt remove -y`

The following shows the installation of the `rtbrick-imgstore` package which provides the `rtb-image` CLI tool.

```
 sudo apt update
Hit:1 https://releases.rtbrick.com/_/latest/ubuntu/jammy/rtbrick-tools jammy InRelease
Hit:3 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:4 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:7 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:8 http://archive.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:9 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [970 kB]
Get:10 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [979 kB]
< ..................................... >
```

```
 sudo apt install rtbrick-imgstore
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  rtbrick-imgstore
0 upgraded, 1 newly installed, 0 to remove and 46 not upgraded.
Need to get 7,731 kB of archives.
After this operation, 26.3 MB of additional disk space will be used.
Get:1 https://releases.rtbrick.com/_/latest/ubuntu/jammy/rtbrick-tools jammy/rtbrick-
tools amd64 rtbrick-imgstore amd64 3.3.0 [7,731 kB]
Fetched 7,731 kB in 0s (41.4 MB/s)
Selecting previously unselected package rtbrick-imgstore.
< ..................................... >
```

2.5.2. Verifying Access (Authentication) to Image Stores

The `rtb-image` command (CLI tool) provided by the `rtbrick-imgstore` package is used to interact with "image stores". The "image stores" are used for delivery of RBFS container images and RtBrick Host installer images.

Similarly to package repositories some of the image stores are *restricted* meaning that they require the client application (`rtb-image` in this case) to authenticate with a TLS client certificate. `rtb-image` re-uses the TLS client certificate already generated by `rtb-apt` for the current system.

This section contains the following topics:

2.5.1. Viewing Available Image Stores
2.5.2. Activating a Restricted Image Store

2.5.3. Verifying Access to Image Stores

Viewing Available Image Stores

The following example shows how to view the available image stores:

```
 sudo rtb-image stores list

Index   UUID                                    Name        RemoteURL
Active  Restricted
0       af73c0a6-40e7-4775-b74b-aadafeabe86d    latest
https://releases.rtbrick.com/_/images/latest    Yes     No
1       c4c896b0-52c5-4343-8a21-e2ca3ea440f1    resources
https://releases.rtbrick.com/_/resources        No      No
2                                               22.5.1
https://releases.rtbrick.com/_/images/22.5.1    No      No
3                                               22.6.1
https://releases.rtbrick.com/_/images/22.6.1    No      No
4                                               22.7.1
https://releases.rtbrick.com/_/images/22.7.1    No      No
< ..................................... >
```

Activating a Restricted Image Store

The following example shows how to activate a (possibly restricted) image store:

```
 sudo rtb-image stores activate 0
```

Verifying Access to Image Stores

If the TLS client certificate for the current system is already trusted by RtBrick, you can use `rtb-image` to download the images. Before downloading the image, you can verify the access to the image stores using the `sudo rtb-image auth check` command.

The following example shows how to verify the access to the image stores:

```
 sudo rtb-image auth check
Image store: latest (af73c0a6-40e7-4775-b74b-aadafeabe86d) ... restricted ... TLS
client certificate accepted
```

2.6. Downloading the Host Image

Image stores contain the Host installer images.

To download Host installer images, perform the following steps:

2.6.1. Updating the Local Cached Copy of the Remote Image Store

2.6.2. Finding the Host Image
2.6.3. Pulling the Host Image
2.6.4. Verifying the Location of the Downloaded Image

2.6.1. Updating the Local Cached Copy of the Remote Image Store

Enter the following command to update the local cached copy of the remote image store for RBFS container and Host images.

```
$ sudo rtb-image update
Local image store cached copy updated to: Store:
/var/cache/rtbrick/imagestores/847c6ecd-df58-462e-a447-38c620a12fe1 Version:
2.4.140875 ValidUntil: 2525-06-20 11:58:44
```

2.6.2. Finding the Host Image

To find the Host image, enter the "sudo rtb-image list" command with the following options.

```
-f, --format=FORMAT Filter images with a specific format. This must be an exact match
of the image format attribute.
-r, --role=ROLE Filter images with a specific role. This must be an exact match of the
image role attribute.
-p, --platform=PLATFORM Filter images for a specific platform. This must be an exact
match of the image platform attribute.
-m, --model=MODEL Filter images for a specific model. This must be an exact match of
the image model attribute.
-v, --ver-range=VER-RANGE Filter images with versions that fall in the provided
version range. See the syntax for version ranges at
```

The following example shows how to find the Host image details for UfiSpace S9510-28DC Multiservice Edge image.

```
$ sudo rtb-image list -f host-installer --role multiservice-edge --ver-range latest
--model s9510-28dc

Store: /var/cache/rtbrick/imagestores/847c6ecd-df58-462e-a447-38c620a12fe1 Version:
2.4.140885 ValidUntil: 2525-07-20 11:58:44

 ┌──────────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────────────────────────────────────┐
 │  UUID Version Role Model Platform Format Cached  │
 │──────────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────────────────────────────────────┤
```

```
| 81c3d77c-a4cd-4ed0-903d-d5d523d81415 25.3.1 multiservice-edge s9510-28dc q2a host-
install··· false |
└──────────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────────────────────────────────────────┘
```

2.6.3. Downloading the Host Image

There are two options available for downloading the Host image:

- Option 1: Downloading the image to the current working directory
- Option 2: Downloading the image to a specific directory

Option 1: Downloading the image to the current working directory

To download the Host image, use the UUID (for example, 81c3d77c-a4cd-4ed0-903d-d5d523d81415) of the Host image in the "sudo rtb-image pull" command. Use the "--here" option to download the image to the current working directory.

```
$ sudo rtb-image pull --here 81c3d77c-a4cd-4ed0-903d-d5d523d81415
bookworm-installer-multiservice-edge-q2a-s9510-28dc-25.3.1.sha512 234 B / 234 B
[==============================================================================]
100.00% 0s
bookworm-installer-multiservice-edge-q2a-s9510-28dc-25.3.1.asc 833 B / 833 B
[==============================================================================]
100.00% 0s
bookworm-installer-multiservice-edge-q2a-s9510-28dc-25.3.1 1.38 GiB / 1.38 GiB
[==============================================================================]
100.00% 18s
bookworm-installer-multiservice-edge-q2a-s9510-28dc-25.3.1: decompressing 100 B / 100
B [============================================================================] 100.00% 8s
81c3d77c-a4cd-4ed0-903d-d5d523d81415 downloaded as /home/user
```

The image will be downloaded to the current working directory under the `rtbrick-host-installer/` directory as shown below:

```
$ ls -al
total 13628
drwxr-xr-x 21 rtbuser rtbuser 4096 Sep 9 07:37 .
drwxr-xr-x 19 root root 4096 Jun 6 2024 ..
drwxr-xr-x 2 rtbuser rtbuser 4096 Sep 9 09:15 rtbrick-host-installer


$ ls -al
total 2909948
drwxr-xr-x 2 rtbuser rtbuser 4096 Sep 9 09:15 .
drwxr-xr-x 21 rtbuser rtbuser 4096 Sep 9 07:37 ..
```

```
rw-r--r- 1 rtbuser rtbuser 1480346261 Sep 9 09:15 bookworm-installer-multiservice-
edge-q2a-s9510-28dc-25.3.1
rw-r--r- 1 rtbuser rtbuser 833 Sep 9 09:15 bookworm-installer-multiservice-edge-q2a-
s9510-28dc-25.3.1.asc
rw-r--r- 1 rtbuser rtbuser 234 Sep 9 09:15 bookworm-installer-multiservice-edge-q2a-
s9510-28dc-25.3.1.sha512
```

Option 2: Downloading the image to a specific directory

Another method to save the image to a specific directory is shown below:

```
$ sudo rtb-image pull --dst=/home/supervisor/ 81c3d77c-a4cd-4ed0-903d-d5d523d81415
bookworm-installer-multiservice-edge-q2a-s9510-28dc-25.3.1.sha512 234 B / 234 B
[==============================================================================]
100.00% 0s
bookworm-installer-multiservice-edge-q2a-s9510-28dc-25.3.1.asc 833 B / 833 B
[==============================================================================]
100.00% 0s
bookworm-installer-multiservice-edge-q2a-s9510-28dc-25.3.1 1.38 GiB / 1.38 GiB
[==============================================================================]
100.00% 18s
bookworm-installer-multiservice-edge-q2a-s9510-28dc-25.3.1: decompressing 100 B / 100
B [============================================================================] 100.00% 0s
81c3d77c-a4cd-4ed0-903d-d5d523d81415 downloaded as /home/supervisor

$ cd /home/supervisor

$ ls -al
total 13628
drwxr-xr-x 21 rtbuser rtbuser 4096 Sep 9 07:37 .
drwxr-xr-x 19 root root 4096 Jun 6 2024 ..
drwxr-xr-x 2 rtbuser rtbuser 4096 Sep 9 09:15 rtbrick-host-installer

$ ls -al
total 2909948
drwxr-xr-x 2 rtbuser rtbuser 4096 Sep 9 09:15 .
drwxr-xr-x 21 rtbuser rtbuser 4096 Sep 9 07:37 ..
rw-r--r- 1 rtbuser rtbuser 1480346261 Sep 9 09:15 bookworm-installer-multiservice-
edge-q2a-s9510-28dc-25.3.1
rw-r--r- 1 rtbuser rtbuser 833 Sep 9 09:15 bookworm-installer-multiservice-edge-q2a-
s9510-28dc-25.3.1.asc
rw-r--r- 1 rtbuser rtbuser 234 Sep 9 09:15 bookworm-installer-multiservice-edge-q2a-
s9510-28dc-25.3.1.sha512
```

2.6.4. Displaying the Location of the Downloaded Image

The details of the downloaded image can be viewed using the following command:

```
$ sudo rtb-image show 81c3d77c-a4cd-4ed0-903d-d5d523d81415

Store: /var/cache/rtbrick/imagestores/847c6ecd-df58-462e-a447-38c620a12fe1 Version:
2.4.140885 ValidUntil: 2525-07-20 11:58:44


UUID: 81c3d77c-a4cd-4ed0-903d-d5d523d81415
Version: 25.3.1
Extra versions:
Tags:
Creation Date: 2025-08-27 07:25:59 +0000 UTC (1 week ago)
Role: multiservice-edge
Platform: q2a
Model: s9510-28dc
Format: host-installer
Architecture: amd64
Filename: rtbrick-host-installer/bookworm-installer-multiservice-edge-q2a-s9510-28dc-
25.3.1
FullPath/URL: /var/cache/rtbrick/imagestores/847c6ecd-df58-462e-a447-
38c620a12fe1/rtbrick-host-installer/bookworm-installer-multiservice-edge-q2a-s9510-
28dc-25.3.1···
SHA512:
897eb424e8e17afabd96e0fdb6542ff9848d51d17fc760f36c8288964f6ba2f653fe5ffd5f3be18e113bf9
64ee79f3b6765aa18eeec18a27c3ee07c082b6aef8
Base Image: -
Embedded Packages: 15
Embedded Images: 1
IsLayered: false
Cached: false
ExtractedPath:
```

The `sudo rtb-image show` command displays only symlink information, so you need to copy the source file.

Once the image has been downloaded successfully, proceed to install it using ONIE. For details, see Installing Host Manually downloaded.

3. RBFS Manual Installation

You can install RtBrick Host manually on an OCP-compliant bare-metal switch. The Open Network Install Environment (ONIE) is an open-source utility that provides an installation environment for OCP-compliant bare-metal switches. ONIE is used to install different network operating systems (NOS) on a device.

ONIE provides several methods for locating a Network Operating System (NOS) installer image. Detailed information about these methods can be found in the ONIE User Guide. The Host image can be installed using any of these methods.

- When installing Host, any existing configurations on the switch will be deleted.
- The current RBFS configurations can be retrieved via a REST call from the RESTCONF endpoint. If you have saved the RBFS configuration using this method, you can load it onto the switch through a RESTCONF endpoint. For more information, refer to the following sections of the RtBrick documentation.
  - Using the Proxy Endpoint
  - RESTCONF API: Use Cases and Examples
- The URL 'server.example.com' used in this document is a fictitious URL used for demonstration purposes only and needs to be replaced with the URL for the webserver which hosts the RBFS image in your deployment.

3.1. RBFS Software Upgrade

If RBFS is already installed on your device and you want to upgrade the software, see the section RBFS-Based Installations. You can perform software upgrade tasks using CLI commands. While the ONIE-based installation methods can technically be used for software upgrades, it is recommended to use the RBFS-based installation methods for their ease of use.

3.2. Prerequisites for Manual Installation

- Ensure that you have downloaded the RtBrick Host image as described in the RBFS Image Download section.
- Provision the out-of-band management interface with an IP address either via DHCP or manual configuration (as described in Manual Configuration of the Management Interface IP).

3.3. ONIE-Based Installation Methods

RBFS supports multiple installation methods using ONIE, allowing both automated and manual deployments.

3.3.1. Zero Touch Provisioning (ZTP)

Automated installation method that allows a device to retrieve installation instructions and install RBFS without any manual intervention. For information on the automated installation process, see section RBFS Automated Installation (Zero Touch Provisioning).

3.3.2. Manual Installation Methods

You can install RBFS using either a USB drive containing the RBFS installer image or by specifying an HTTP URL to a remotely hosted installer image. The USB method is used when network-based installation is unavailable, while the URL method allows ONIE to download and install the image directly from a remote server.

Installing RBFS Using a USB Thumb Drive

This section describes how to install image using a USB thumb drive.

Prerequisites

- Format the USB drive with the FAT32 file system format because we need to place the RBFS image on the root directory of the USB drive.
- Ensure that you have downloaded the RBFS Host image as described in the RBFS Image Download section.

Installation Procedure

You can also find instructions for installing via a USB thumb drive in the ONIE User Guide.

To install via USB, insert the USB drive to your computer and assume the USB drive appears as `/dev/sda1` and is mounted at `/media/rtbuser/4356-00B1` on Linux. This may vary depending on your system and operating system.

```
$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1        29G   16K   29G   1% /media/rtbuser/4356-00B1
```

It is crucial to rename the RBFS Host image to `onie-installer`, as ONIE only recognizes images with this name at the root of the USB drive.

To install via USB, simply copy the installer image (in this example, the image name is `bookworm-installer-multiservice-edge-q2c-s9600-72xc-25.3.1`) to the root directory of the USB thumb drive, as shown below:

```
$ cp bookworm-installer-multiservice-edge-q2c-s9600-72xc-25.3.1 /media/rtbuser/4356-
00B1/onie-installer

$ ls -al /media/rtbuser/4356-00B1/
```

```
total 1256820
drwxr-xr-x  2 rtbuser rtbuser      16384 Jan  1  1970 .
drwxr-x---+ 3 root     root           4096 Jan  9 11:49 ..
-rw-r--r--  1 rtbuser rtbuser 1286955159 Jan  9 11:49 onie-installer
```

- Remove the USB drive from your computer and insert it into one of the USB ports on the front or rear panel of your ONIE-enabled device.
- Insert the cable into the console port and connect to the console port of the device.
- Power on the device and reboot it. ONIE will automatically detect the onie-installer file located at the root of the USB drive and execute it.

```
root@bl1-pod1:~# reboot
```

- Wait for the device to show the "login:" prompt after installing the image. You can then log in and check the image version.

```
ufi13 login:
ufi13 login: supervisor
Password:

+------------------------------------------------------------------------+
|                                                                        |
    RtBrick Host system ufi13.q2c.u19.r5.nbg.rtbrick.net:
        Date:       Wed Jan  7 05:51:30 AM UTC 2026
        Uptime:     up 0 minutes

    Image metadata:
        UUID:       bc4c74eb-db7c-4183-af87-1d31ae2abee3
        Version:    25.4.0-g4internal.20260106120631+BDO2078.C1fcf7e8e
        Role:       accessleaf
        Platform:   q2c
        Model:      s9600-72xc
        Format:     host-installer
        Build date: 2026-01-06 12:06:31 UTC
        Based on:   Debian GNU/Linux 12 (bookworm)
|                                                                        |
+------------------------------------------------------------------------+
```

The system by default selects Image Partition A for installation. If Partition A is already in use, it automatically selects Partition B. It ensures an unused partition is always chosen for installation.

Configuring the Management Interface IP Address

If DHCP is not available, you need to manually configure the IP address, subnet mask, and default gateway for the device's management port while still logged in from its console port.

1. Identify the management port. Check the device documentation to determine which network interface is designated as the management interface (labeled "ma1").
2. Modify the `ma1` interface network parameters by adding IP address, Netmask, and gateway using your preferred editor. The example below shows how to modify these parameters using the Vim editor.

```
supervisor@host:/etc/network/interfaces.d $ vim ma1
auto ma1
iface ma1 inet static        <----- modify ma1 inet assignment as static
      address 192.0.2.187    <----- ma1 management interface ip address
      netmask 255.255.255.0  <----- subnet mask
      gateway 192.0.2.10     <----- configure gateway
```

3. Restart the networking service by disabling and enabling the `ma1` interface, as shown in the example below. By default, the default route will point to the gateway IP address.

```
sudo ifdown ma1
sudo ifup ma1
```

Installation Using the URL of the Installer Image

In this installation mode, the system boots into ONIE and retrieves the installer image from the remote server over the network. ONIE requires the installer image to be hosted on an HTTP-accessible server that is reachable during the boot through a dedicated Out-of-Band management connection.

Prerequisites

- Ensure that you have downloaded the RBFS Host image as described in the RBFS Image Download section.
- Ensure that you have configured an HTTP server to host the downloaded images, making them accessible for ONIE during installation.

Installation Procedure

You can also find instructions for installing the Host image over the network in the ONIE User Guide.

To install the RtBrick Host image over the network, perform the following steps:

On a fresh box, **host prompt** is not available, so skip to **ONIE prompt** section.

host prompt section:

Manually select ONIE boot mode

1. Connect to the console port

2. Reboot the device

```
root@bl1-pod1:~# reboot
```

3. Select "**ONIE: Install OS**" from the next selection menu displayed.

```
 +----------------------------------------------------------------------------+
 |*ONIE: Install OS   <---- Select this one                                   |
 | ONIE: Rescue                                                               |
 | ONIE: Uninstall OS                                                         |
 | ONIE: Update ONIE                                                          |
 | ONIE: Embed ONIE                                                           |
 | RTB: Image-A Kernel                                                        |
 | RTB: Image-B Kernel                                                        |
 |                                                                            |
 |                                                                            |
 |                                                                            |
 |                                                                            |
 |                                                                            |
 +----------------------------------------------------------------------------+


      Use the ^ and v keys to select which entry is highlighted.
      Press enter to boot the selected OS, `e' to edit the commands
      before booting or `c' for a command-line.
```

4. Wait for the **ONIE:**/ # prompt.

```
NOTICE: ONIE started in NOS install mode.  Install mode persists
NOTICE: until a NOS installer runs successfully.

** Installer Mode Enabled **
ONIE:/ #
ONIE:/ #
ONIE:/ #
```

5. Provide the URL of the Host installer image location.

The following is an example URL:

```
onie-nos-install http://server.example.com/rtbrick.net/_/images/latest/rtbrick-host-
installer/bookworm-installer-multiservice-edge-q2a-s9510-28dc-25.3.1
```

6. Wait until the device displays the "**login:**" prompt after the image upgrade completes. You can then log into the device and verify the image version.

```
ufi13 login:
ufi13 login: supervisor
Password:

+------------------------------------------------------------------------+
|                                                                        |
    RtBrick Host system ufi13.q2c.u19.r5.nbg.rtbrick.net:
        Date:       Wed Jan  7 05:51:30 AM UTC 2026
        Uptime:     up 0 minutes

    Image metadata:
        UUID:       bc4c74eb-db7c-4183-af87-1d31ae2abee3
        Version:    25.4.0-g4internal.20260106120631+BDO2078.C1fcf7e8e
        Role:       accessleaf
        Platform:   q2c
        Model:      s9600-72xc
        Format:     host-installer
        Build date: 2026-01-06 12:06:31 UTC
        Based on:   Debian GNU/Linux 12 (bookworm)
|                                                                        |
+------------------------------------------------------------------------+
```

3.4. RBFS-Based Installations

You can perform the subsequent installation tasks (after the initial RBFS installation) such as software upgrades and recovery operations. The commands, available on the RtBrick host system and RBFS CLI, can be used for downloading, installing on the alternate partition, deleting images, and maintaining multiple software versions and so on with minimal impact on the running software. These CLIs include:

- RtBrick Host System Commands
- RBFS CLI Commands

3.4.1. RtBrick Host System Commands for Installation Operations

You can manage the RBFS image on the host system. The following commands allow you to perform various actions such as rebooting an image and install another version of the software in the other partition of the image. This operation is supported only if RBFS is already present on the system.

rtb-reboot

This command allows you to reboot the host installer with either RTB image A or image B. This is a temporary reboot of the selected image (it restarts the system using the specific image).

Syntax:

rtb-reboot -p (A | B)

rtb-reboot --no-reboot (A | B)
rtb-reboot (A | B) (install | update | rescue | uninstall)

The command options and description are provided in the following table:

Attribute	Description
-p	Sets the selected boot target permanently. It indicates every time the system reboots in the future, it will use this boot target unless you change it again.
--no-reboot	Prevents the system from rebooting immediately after setting the boot target. It is useful if you want to change the boot option but delay the reboot.
A	Boots the system using RTB Image-A Kernel (one of the two software image partitions).
B	Boots the system using RTB Image-B Kernel (the alternate image partition).
install	Boots into ONIE in Install mode to load a new operating system once.
update	Boots ONIE in 'Update' mode to update the ONIE software itself.
rescue	Boots ONIE in 'Rescue' mode for recovery and troubleshooting.
uninstall	Boots ONIE in uninstall mode, which removes the installed operating system.

rtb-reboot A

If the host installer is running image B, use the command to `rtb-reboot A` to reboot the device and directly boots image A. This is a temporary reboot of the selected image (restarts the system using the specific image).

This does not require any further boot selection input. The following example shows the output of the `rtb-reboot A` command:

```
supervisor@host:~ $ sudo rtb-reboot A
Using GRUB partition: /dev/sda2
Setting one-time boot to: RTB: Image-A Kernel
System will reboot into A in 10 seconds.
Press Enter to reboot immediately or Ctrl+C to cancel.
Broadcast message from root@host on pts/0 (Fri 2025-09-12 12:41:48 UTC):
The system will reboot now!
         Stopping session-26.scope- Session 26 of User supervisor...
[  OK  ] Removed slice system-modpr···lice - Slice /system/modprobe.
```

rtb-reboot B

If the host installer is running image A, use the command to `rtb-reboot B` to reboot the device and directly boots image B. This is a temporary reboot of the selected image (restarts the system using the specific image).

This does not require any further boot selection input. The following example shows the output of the `rtb-reboot B` command:

```
supervisor@host:~ $ sudo rtb-reboot B
Using GRUB partition: /dev/sda2
Setting one-time boot to: RTB: Image-B Kernel
System will reboot into B in 10 seconds.
Press Enter to reboot immediately or Ctrl+C to cancel.
Broadcast message from root@host on pts/0 (Fri 2025-09-12 12:51:03 UTC):
The system will reboot now!
         Stopping session-2.scopeession 2 of User supervisor...
[  OK  ] Removed slice system-modpr···lice - Slice /system/modprobe.
```

rtb-install

Image partition enables the installation of two different software versions, one on Image A and another on Image B. If you installed the software on Image A, you can install a different version on Image B.

The `rtb-install` command allows you to install another version of the software on Image B (if you have already installed RBFS on Image A) using a specified file path or URL as the source location.

For example, if RTB image A is already in use, you can target RTB image B for installation using a URL as the source.

The following output is used to figure out which image partition is currently running, either Image A or Image B. This output shows image B that indicates currently running image partition is Image B.

```
supervisor@host:/ $ ls -la /imag*
-rw------- 1 root root 2 Sep 19 05:59 /image-B
```

When you execute the `rtb-install` command, it downloads the image from the given URL, installs it automatically, and provides an acknowledgment message once the installation is complete. Afterward, the system prompts for a reboot for the new installation to take effect.

Syntax:

rtb-install -i (**A** | **B**) (**-f** <filename> | **-u** <url>)

The command options and description are provided in the following table:

Attribute	Description
-i	Specifies the installation target image partition.
A	Installs using RTB Image-A Kernel (one of the two software image partitions).
B	Installs using RTB Image-B Kernel (the alternate image partition).
-f	Specify the file name. Installs the image from the specified file name.
-u	Specify the URL. Installs the image by downloading it from the specified URL location.

The following example shows the output of the `rtb-install` command:

```
supervisor@host:/ $  sudo rtb-install -i A -u
http://server.example.com/_/images/latest/rtbrick-host-installer/bookworm-installer-
multiservice-edge-q2a-s9510-28dc-Downloading file from
http://server.example.com/_/images/latest/rtbrick-host-installer/bookworm-installer-
multiservice-edge-q2a-s9510-28dc-25.3.1-candidate.1...
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 1421M  100 1421M    0     0  7360k      0  0:03:17  0:03:17 --:--:-- 9467k
Starting the downloaded file with IMAGE=A...
Verifying archive integrity...  100%   SHA256 checksums are OK. All good.
Uncompressing bookworm-installer-multiservice-edge-q2a-s9510-28dc-25.3.1-candidate.1
100%

installer started from within RTB-System
skipping partitioning...
creating ext4 filesystem on partition RTB-IMAGE-A
mke2fs 1.47.0 (5-Feb-2023)
/dev/sda6 contains a ext4 file system labelled 'RTB-IMAGE-A'
    last mounted on / on Fri Sep 12 13:22:37 2025
Discarding device blocks: done
Creating filesystem with 4194304 4k blocks and 1048576 inodes
Filesystem UUID: 65b409b3-4c51-41ba-9ff5-51ce6dfbf16e
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

tune2fs 1.47.0 (5-Feb-2023)
installation to Image-A partition
[ 3345.852798] EXT4-fs (sda6): mounted filesystem with ordered data mode. Quota mode:
none.
```

```
extracting image to /mnt/tmp.72RKIWi8Bj...
...finished
DEBUG: A
A: setting 65b409b3-4c51-41ba-9ff5-51ce6dfbf16e in /mnt/tmp.72RKIWi8Bj/etc/fstab
mount: (hint) your fstab has been modified, but systemd still uses
       the old version; use 'systemctl daemon-reload' to reload.
found ma1 interface, creating udev rule
installing grub bootloader...
unmounting partitions...
[ 3412.562356] EXT4-fs (sda6): unmounting filesystem.
finished installation!
please reboot the system
```

The installation is now complete. You can proceed with a reboot to apply the changes. The following example shows the output of the `reboot` command:

```
supervisor@host:/ $ sudo reboot
Broadcast message from root@host on pts/0 (Fri 2025-09-12 14:40:37 UTC):
The system will reboot now!

supervisor@host:/ $ [  OK  ] Stopped   OK  ] Removed slice system-modpr···lice - Slice
/system/modprobe.
[  OK  ] Stopped target graphical.target - Graphical Interface.
[  OK  ] Stopped target rpcbind.target - RPC Port Mapper.

After the reboot, then the login prompt will be shown as below,

[  OK  ] Reached target graphical.target - Graphical Interface.
         Starting systemd-update-ut··· Record Runlevel Change in UTMP...
[  OK  ] Finished systemd-update-ut··· - Record Runlevel Change in UTMP.
Debian GNU/Linux 12 host ttyS0
host login: [   37.241798] device ma1 entered promiscuous mode
[   37.398983] device ma1 left promiscuous mode

host login:
```

You can log in and check the image version.

```
ufi13 login:
ufi13 login: supervisor
Password:

+------------------------------------------------------------------+
|                                                                  |
    RtBrick Host system ufi13.q2c.u19.r5.nbg.rtbrick.net:
        Date:       Wed Jan  7 05:51:30 AM UTC 2026
        Uptime:     up 0 minutes

    Image metadata:
```

```
        UUID:       bc4c74eb-db7c-4183-af87-1d31ae2abee3
        Version:    25.4.0-g4internal.20260106120631+BDO2078.C1fcf7e8e
        Role:       accessleaf
        Platform:   q2c
        Model:      s9600-72xc
        Format:     host-installer
        Build date: 2026-01-06 12:06:31 UTC
        Based on:   Debian GNU/Linux 12 (bookworm)
|                                                                    |
+--------------------------------------------------------------------+
```

3.4.2. RBFS Commands for Installation Operations

Once RBFS is installed and operational, RBFS CLI commands are available for subseuent installation operations. You can perform operations such as image downloading, installing, activating, and deleting while the router is operational. You can use these commands to perform these operations through both in-band and out-of-band (OOB) interfaces. It allows you to perform software installation and upgrade tasks through the management plane without requiring a dedicated OOB connection.

Software can be downloaded and installed without any downtime. Only a brief downtime occurs during the activation.

The following sections provide information about the commands:

Download Software Image

The following command downloads the software image from the specified HTTPS URL and saves it to the configuration partition using the provided filename.

The URL corresponds to the local directory where the software image was downloaded. The file name is derived from the downloaded image in the directory. For information about the URL and file name of the downloaded image, refer to the section Location of the Downloaded Image.

Syntax:

request software download <url> <file-name>

Option	Description
<url>	Specify HTTPS URL location.
<file-name>	Specify the file name.

The following example shows the executed show command and its corresponding output that includes the generated job UUID.

```
supervisor@rtbrick.net: cfg> request software download
```

```
https://server.example.com/_/images/latest/rtbrick-host-installer/bookworm-installer-
multiservice-edge-q2c-s9600-72xc-25.4
.1.1 rbfs-q2c-img
Task successful with job_uuid: 4aeab4e4-5cb9-49c9-af5e-1281e7446242
```

Delete Software Image

This command deletes the specified software image file from the partition:

Syntax:

request software delete <file-name>

Option	Description
<file-name>	Specify the file name. NOTE: Autocompletion is supported for specifying software image file names.

The following example shows the executed show command and its corresponding output.

```
supervisor@rtbrick.net: cfg> request software delete rbfs-q2c-img
Task successful
```

Install Software Image

This command starts the installation of the software image into the specified partition.

You must always specify a partition and if a mounted (active) partition is selected, then it throws an error.

Syntax:

request software install (A | B) <file-name>

Option	Description
A | B	Specify the target image partition.
<file-name>	Specify the file name. NOTE: Autocompletion is supported for specifying software image file names.

The following example shows the executed show command and its corresponding output that includes the generated job UUID.

```
supervisor@rtbrick.net: cfg> request software install partition-b rbfs-q2c-img
Task successful with job_uuid: 5d596d33-3b26-4cfb-9948-ec2b0ae07800
```

Set Active Partition Flag

You can set a reboot flag for a specific software partition (for example, Partition A or Partition B) as the active partition for the all future reboots. The device will boot from that partition every time it restarts, until changed again.

Syntax:

request software activate (A | B) <option>

Option	Description
A | B	Specify the desired partition.
temporary	A temporary reboot activates the specified partition only for the next boot. After the device restarts again, it automatically reverts to the original active partition.
reboot	Reboots the specified partition.

The following example shows the executed show command and its corresponding output.

```
supervisor@rtbrick.net: cfg> request software activate partition-b reboot
Task successful
```

3.4.3. RBFS APIs for Installation Tasks

RBFS provides the following API endpoints to perform reboot operations. You can use these API endpoints after the initial RBFS installation on your device.

```
* POST /api/v1/ctrld/system/_reboot : Reboot the switch.

* POST /api/v1/ctrld/system/_reinstall : Reboot the switch in onie install mode.

* POST /api/v1/ctrld/system/_rescue :Reboot the switch in onie rescue mode.

* POST /api/v1/ctrld/system/_uninstall :Reboot the switch in onie uninstall mode.

* POST /api/v1/ctrld/system/_update :Reboot the switch in onie update mode.
```

For more information about RBFS APIs for installation and software management tasks, see RBFS CTRLD APIs.

3.5. Operational Commands

3.5.1. software partition

Using these show commands, you can view the image partition related information in the system.

Syntax:

show software partition

This command displays all software partitions on the system along with the installed software images. If no software is installed in a partition, the command lists any software image files that have been downloaded to that partition.

The active partition (the one currently running) is clearly shown in the output. The partition marked with the 'reboot' tag shows which partition will become active after the next reboot.

Example Command:

```
show software partition
```

Example: The output shows software image details for partitions A and B, including image IDs, versions, and platform information. Partition B is currently set as both the active and permanent boot image. Temporary boot option is not set.

```
supervisor@rtbrick.net: op> show software partition
System Image Information:
  Image A Details:
    Image ID            : 64ad2c9a-e72a-41f6-be80-de8e9325cc3b
    Image Type          : host-installer
    Platform Chipset    : q2a
    Model               : s9510-28dc
    Element Role        : multiservice-edge
    Image Version       : 2504.0.0-g6daily.20251028034032+Bdevelopment.Cd73d28d2
    RTB Image Version   : 25.4.0-g6daily.20251028034032+Bdevelopment.Cd73d28d2
  Image B Details:
    Image ID            : ebbacee2-9cf2-4ec4-bd81-05ed23b40250
    Image Type          : host-installer
    Platform Chipset    : q2a
    Model               : s9510-28dc
    Element Role        : multiservice-edge
    Image Version       : 2504.0.0-g6daily.20251030035506+Bdevelopment.Cd73d28d2
    RTB Image Version   : 25.4.0-g6daily.20251030035506+Bdevelopment.Cd73d28d2
  Boot Options:
    Active Image        : B
    Permanent Boot      : B
    Temporary Boot      : Not Set
  Installer Files: Not available
```

Example: The output shows that partition B is the active and permanent boot image and partition A is set as the temporary boot image. The system will boot from partition A on the next reboot but will revert to partition B for future boots.

```
supervisor@rtbrick.net: op> show software partition
System Image Information:
  Image A Details:
```

```
    Image ID            : 64ad2c9a-e72a-41f6-be80-de8e9325cc3b
    Image Type          : host-installer
    Platform Chipset    : q2a
    Model               : s9510-28dc
    Element Role        : multiservice-edge
    Image Version       : 2504.0.0-g6daily.20251028034032+Bdevelopment.Cd73d28d2
    RTB Image Version   : 25.4.0-g6daily.20251028034032+Bdevelopment.Cd73d28d2
  Image B Details:
    Image ID            : ebbacee2-9cf2-4ec4-bd81-05ed23b40250
    Image Type          : host-installer
    Platform Chipset    : q2a
    Model               : s9510-28dc
    Element Role        : multiservice-edge
    Image Version       : 2504.0.0-g6daily.20251030035506+Bdevelopment.Cd73d28d2
    RTB Image Version   : 25.4.0-g6daily.20251030035506+Bdevelopment.Cd73d28d2
  Boot Options:
    Active Image        : B
    Permanent Boot      : B
    Temporary Boot      : A
  Installer Files: Not available
```

Example: The output shows two software images installed (Image A and Image B). Image B is currently active, while Image A remains the permanent boot image.

```
supervisor@rtbrick.net: op> request software activate partition-a
Task successful
supervisor@rtbrick.net: op> show software partition
System Image Information:
  Image A Details:
    Image ID            : 64ad2c9a-e72a-41f6-be80-de8e9325cc3b
    Image Type          : host-installer
    Platform Chipset    : q2a
    Model               : s9510-28dc
    Element Role        : multiservice-edge
    Image Version       : 2504.0.0-g6daily.20251028034032+Bdevelopment.Cd73d28d2
    RTB Image Version   : 25.4.0-g6daily.20251028034032+Bdevelopment.Cd73d28d2
  Image B Details:
    Image ID            : ebbacee2-9cf2-4ec4-bd81-05ed23b40250
    Image Type          : host-installer
    Platform Chipset    : q2a
    Model               : s9510-28dc
    Element Role        : multiservice-edge
    Image Version       : 2504.0.0-g6daily.20251030035506+Bdevelopment.Cd73d28d2
    RTB Image Version   : 25.4.0-g6daily.20251030035506+Bdevelopment.Cd73d28d2
  Boot Options:
    Active Image        : B
    Permanent Boot      : A
    Temporary Boot      : Not Set
  Installer Files: Not available
```

3.5.2. Job Data

Using the job data show commands, you can track the real-time activity related to the software download or installation.

Syntax:

show software job <job_uuid> data

This command is used to view the real-time activity and log output of a software job. For example, tracking the progress of a software image download or installation.

A job UUID is generated and displayed when an image download or installation is initiated. The following example shows a 'job-uuid' is printed on the screen.

```
Task successful with job_uuid: 4aeab4e4-5cb9-49c9-af5e-1281e7446242
```

Example command:

```
show software job 2f3d3570-69d2-4307-9c97-305155f8016b data
```

Example: The output shows the specific software job that shows the system initiated a download. The download is progressing at a rate of around 110 MB/s.

```
supervisor@rtbrick.net: op> show software job 2f3d3570-69d2-4307-9c97-305155f8016b
data
--2025-10-14 04:56:38--  https://server.example.com/_/images/latest/rtbrick-host-
installer/bookworm-installer-spine-q2c-s9600-32x-25.4.0-
g6daily.20251013163757+Bdevelopment.C3dae1a64
Resolving server.example.com (server.example.com)... 10.200.137.175
Connecting to server.example.com (server.example.com)|10.200.137.175|:443...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 1450171722 (1.3G) [text/plain]
Saving to: □/var/config/installer/image-1-dev□

     0K ........ ........ ........ ........  2% 96.8M 14s
 32768K ........ ........ ........ ........  4%  112M 13s
 65536K ........ ........ ........ ........  6%  112M 12s
 98304K ........ ........ ........ ........  9%  112M 12s
131072K ........ ........ ........ ........ 11%  111M 11s
163840K ........ ........ ........ ........ 13%  112M 11s
196608K ........ ........ ........ ........ 16%  112M 11s
229376K ........ ........ ........ ........ 18%  112M 10s
262144K ........ ........ ........ ........ 20%  111M 10s
294912K ........ ........ ........ ........ 23%  113M 10s
327680K ........ ........ ........ ........ 25%  112M 9s
360448K ........ ........ ........ ........ 27%  112M 9s
```

```
393216K ........ ........ ........ ........ 30%  112M 9s
425984K ........ ........ ........ ........ 32%  112M 8s
458752K ........ ........ ........ ........ 34%  105M 8s
491520K ........ ........ ........ ........ 37%  112M 8s
<...>
2025-10-14 04:56:50 (111 MB/s) - /var/config/installer/image-1-dev saved
[1450171722/1450171722]
```

3.5.3. Job Status

This show command is used to view the final state or result of the software job after execution.

Syntax:

show software job <job_uuid> status

The following example shows the executed show command. The corresponding output shows the software job status that has completed and its state is marked as done with an exit code of 0.

```
supervisor@rtbrick.net: op> show software job 2f3d3570-69d2-4307-9c97-305155f8016b
status
State      : done
Exit Code  : 0
```

4. RBFS Automated Installation (ZTP)

4.1. Overview

Zero Touch Provisioning (ZTP) automates the tasks of installing software images. It is a method for setting up and configuring devices automatically. ZTP installs or upgrades the RBFS software image on your hardware platforms without any manual intervention.

ZTP automatically provisions routers newly installed in the network and it is very useful in deploying routers in a large-scale environment as it eliminates much of the manual intervention. ZTP is also used to automate the software upgrade process and help with a high level of network automation.

4.2. ZTP Workflow

A new hardware platform comes pre-installed with the ONIE (Open Network Installation Environment). ONIE is an open-source installation environment that acts as an enhanced boot loader utilizing capabilities in a Linux or BusyBox environment. ONIE allows users and channel partners to install the Network Operating System as part of provisioning.

ONIE requires a management LAN to obtain the configuration and software image information through the management interface. ONIE can access only the management interface. It starts a Dynamic Host Configuration Protocol (DHCP) based discovery process to obtain basic configuration information, such as the management IP address and the URL of the image to install on the bare-metal switch.

Then ONIE pulls the image and boots it.

Even after ONIE boots the image, the switch is not configured. This leads to questions about how to configure the switch.

The RtBrick images come with some pre-installed daemons. The pre-installed Control Daemon (CtrlD) is responsible for the management of the switch, and takes over after the image is activated.

The Control Daemon is responsible for configuring the switch. To do this, the hardware platform must be connected to the DHCP server and the management server through a management LAN.

The management server is responsible for providing the image binaries and the configuration of each device.

In the ZTP, ONIE performs the role of discovering, downloading and activating the image from the image registry.

In essence, the following is the high-level workflow of ZTP process:

ONIE performs the following tasks:

- DHCP discovery

- Image download
- Image activation

Control Daemon performs the following tasks:

- DHCP discovery
- Switch configuration

ONIE allows to automate the firmware update. The image request to the management server is slightly different, and the management server needs to provide the firmware update image that the device vendor provides.

This section provides information about the NOS installation and firmware (FW) update.

4.2.1. ZTP Process

This section provides information about ZTP process. Figure. 1 illustrates the ZTP process at a high level.

The ZTP process is divided into two main parts:

Software Image Discovery and Installation

The ONIE in the device uses information that you have defined on the Dynamic Host Configuration Protocol (DHCP) server to locate the IP address and image download URL.

- ONIE uses different ways to pull the image from the repository for downloading. In the ZTP process, HTTP is used to pull the image because ONIE conveys the serial number as the HTTP header. This serial number allows the image registry to identify the switch and select the appropriate image.

Along with the serial number, ONIE also sends the `onie-operation` that allows to distinguish between an `os-install` and `onie-update`, and select the correct image for either NOS install or firmware upgrade.

- See the ONIE image discovery for further information (/ONIE/)
- CtrlD configuration discovery and application.
- CtrlD sends DHCPINFORM to request all options required for configuration discovery.
- The configurations are downloaded from the management server (`httpd`) and applied.

Image Partition

RBFS supports multiple partitions, each containing one image (e.g., Image-A and Image-B). Each partition provides an isolated environment for installing and running software without affecting the active system. This enables safe upgrades; if a new image fails, the system rolls back to the previous working version.

Each partition is independently bootable, allowing the system to switch between them as needed.

During the ZTP installation, the system defaults to Image Partition A. If Partition A is already in use, it automatically selects Partition B. It ensures an unused partition is always chosen for installation.

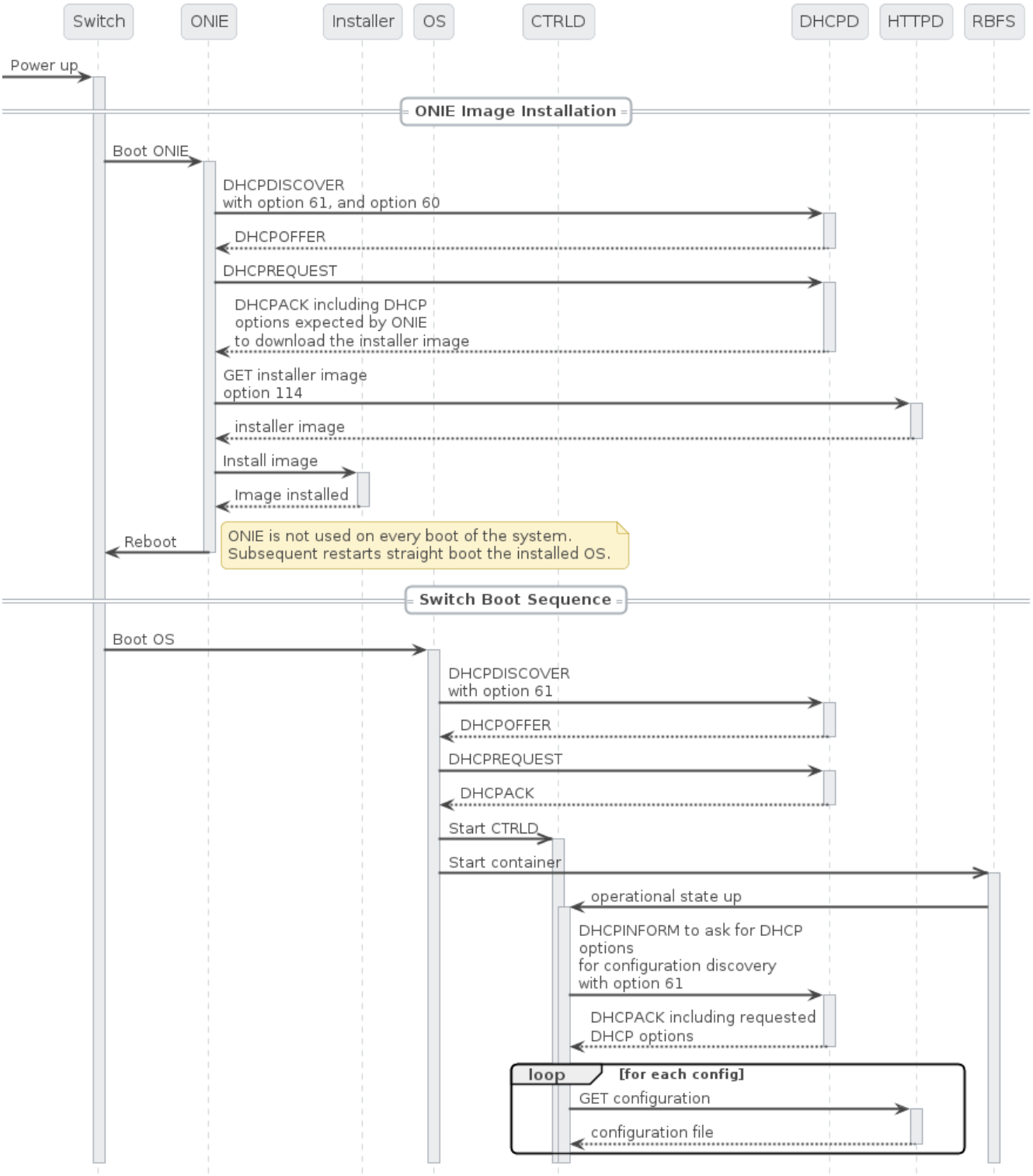


Figure 5. The ZTP Process

Figure 2. depicts the relationship between the fabric, the DHCP server, and the management server.

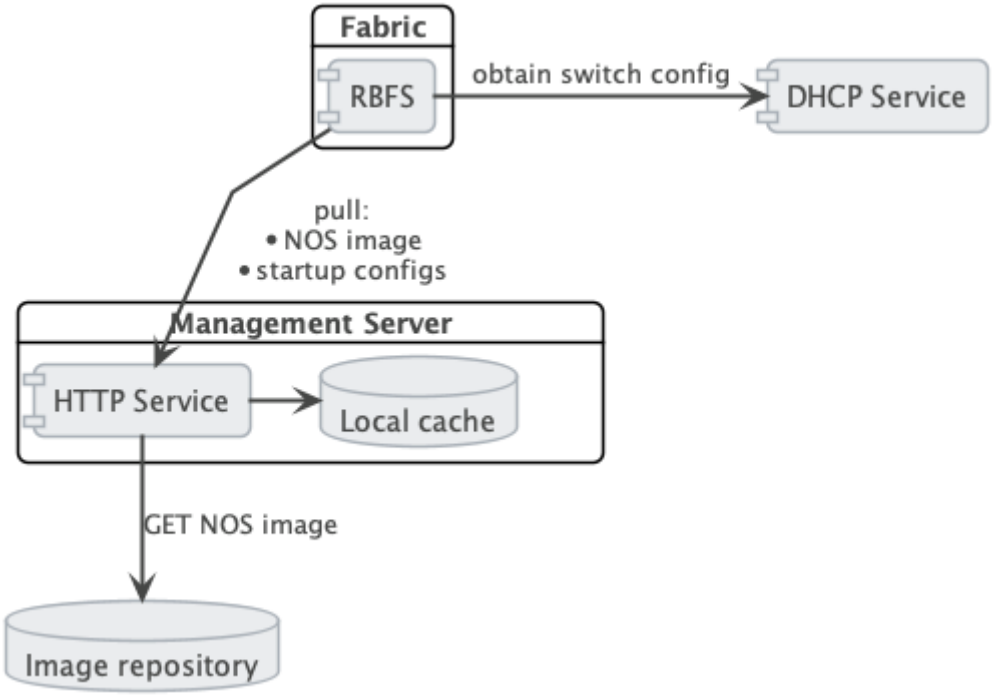


Figure 6. The Management Server Architecture

4.3. DHCP Service

Because of its low set of requirements, the default DHCP server shipped with Debian 12, isc-dhcp, is used to run the DHCP service.

The following code shows an example configuration of a DHCP server and hardware box (**dhcp.conf**).

```
authoritative;
default-lease-time 600;
max-lease-time 72----

# This is only needed if the version is lower than 4.4
option loader-pathprefix code 210 = text;


subnet 10.0.0.0 netmask 255.255.255.0 {
  range 10.0.0.200 10.0.0.250;
  option routers 10.0.0.138;
  option domain-name-servers 10.0.0.210;
  option domain-name "local";
  host LEAF01 {
    # Identify client by MAC address.
    hardware ethernet 48:65:ee:11:da:85;
    # Identify client by serial number
    option dhcp-client-identifier "\000WLC1C27L00003P2";
    fixed-address 10.0.0.250;
    option host-name LEAF01;
    # Set DHCP option 114 (default-url) to set the installer image URL.
```

```
    # ONIE loads the installer image from the specified URL.
    option default-url "http://managementserver/ztp/image";
    # Set DHCP option 210 (path prefix) to set the configuration base URL.
    # CTRLD loads all configuration files from this base URL.
    option loader-pathprefix "http://managementserver";
  }
}
```

Most of the used options are already predefined in the ISC-DHCP server. You can see the reference under `/ISCKB/`, the `loader-pathprefix` is defined since DHCP 4.4, so if you use an older one, define it as described above.

4.4. HTTP Service (Management Server)

The HTTP daemon (`httpd`) is responsible for providing the NOS installer and the configuration files.

Therefore, a self-implemented Golang HTTP server is used, which reads the `ONIE_SERIAL_NUMBER` and `ONIE-OPERATION` HTTP header and maps them to the NOS/FW installer image download path, and maps the serial number to the ZTP configuration files. For more details about the configuration files, see the following section.

The `ONIE-OPERATION` header can have the following values:

- install nos: `os-install`
- update firmware: `onie-update`

The following sections provide information about the installation and configuration of the server.

4.4.1. ZTP installation

For the installation, you can choose any one of the following two options:

ZTP Installation with the Debian Package

You must perform the following steps for ZTP installation using the Debian package.

- Ensure that you have added the `rtrbick` repository to your `apt.sources` list and updated the cache.
- Ensure that the port `80` is available and not in use on your device.
- Install the package `rtbrick-fabric-ztp`.
- The package installs a `systemd` service named `rtbrick-fabric-ztp`.
- Ensure that the service is running with `sudo systemctl status rtbrick-fabric-ztp`.
- The default location for the ZTP configuration files is `/var/rtbrick/ztp/configs/` where you need to copy your configuration files.

If you want to override server settings, perform the following:

- Edit the service configuration file `/etc/systemd/system/rtbrick-fabric-ztp.service` and add parameters to the `ExecStart` command.
- Parameter `--addr`: the listen address of the server, default is `0.0.0.0:80`.
- Parameter `--requestTimeout`: the request timeout server in seconds, default is `600`, must possibly be increased depending on the connection speed and image file sizes.
- Parameter `--filePath`: the location for the ZTP configuration files, the default location is `/var/rtbrick/ztp/configs/`.

ZTP Installation as Docker Container

You must perform the following steps for ZTP installation as a docker container.

- Ensure that you have access to the `rtbrick` docker registry.
- Ensure that the port `80` is available and not in use on your device.
- Create a compose file `docker-compose.yml`. The following is a sample compose file.

```
version: '3.3'
services:
  ztp:
    image: 'docker.rtbrick.com/rbms-fabric-ztp:latest'
    container_name: rbms-fabric-ztp
    restart: unless-stopped
    ports:
      - '80:80'
    volumes:
      - './configs:/var/rtbrick/ztp/configs'
```

- The compose setup uses a 'bind mound' method for the ZTP configuration folder. Therefore, the `docker-compose.yml` must be placed in the same location together with the `./configs` folder for the ZTP configurations. To know the details of the configuration files, see the following sections.
- If required, adapt the compose file for a different image version, port binding or different configuration folder location.
- Start the container using the `docker-compose up -d` command.

4.4.2. ZTP configuration

The HTTP service matches the `ONIE-SERIAL-NUMBER` header to the configuration files. Therefore, the configuration folder should contain a JSON file for the serial number (`<serial_number>.json`) for each supported serial number.

This file contains settings for locations of all additional configuration files that have to be delivered for the specific device and settings for the NOS installer image and the firmware update image.

Example `sample.json` file for a serial number 'sample':

```
{
  "description": "192.168.202.116",
  "ctrld": "ctrld.json",
  "ctrldrbac": "ctrldrbac.json",
  "startup": "sample_startup.json",
  "accessjwks": "sample_accessjwks.json",
  "apigwd": "sample_apigwd.json",
  "tls": "sample_tls.pem",
  "image": "http://server.example.com/_/images/latest/rtbrick-onl-installer/rtbrick-
onl-installer-accessleaf-qmx-20.4.0-g8daily.20200415051734+Bmaster.C059a09ea",
  "update_image": "http://pgk.rtbrick.net/firmwares/onie-firmware-x86_64-
ufispace_s9600_32x_ufispace_s9600_64x-r0_v0.3.0.updater"
}
```

Image Location Configuration

For the configuration entries `"image"` and `"update_image"` you have three possibilities:

- Redirect URL: Configuration value must start with `http`, the server redirects the request to download the image from the URL. For example, "http://server.example.com/_/images/latest/rtbrick-host-installer/rtbrick-host-installer-accessleaf-qmx-20.4.0-g8daily.20200415051734+Bmaster.C059a09ea"
- Absolute File Location: config value must start with `/`, can point to any file on the local disk, example `/usr/share/images/rtbrick-host-installer.img`.
- Relative File Location: config value must be a filename and not start with `/`, points to any file in the `<ztppath>/configs/images/` folder, example "rtbrick-host-installer.img"

4.5. Control Daemon

Once the RBFS image is activated by ONIE, Control Daemon (CtrlD) is responsible for executing the remaining tasks and configuring the switch. CtrlD acts as a post-ZTP daemon, it runs after the image is activated.

There are various configuration files that CtrlD can load from a management server and apply to the system.

- **CtrlD config**: This is the base configuration for CtrlD. There the Graylog can be specified, but also the authentication and authorization mechanism can be controlled.
- **CtrlD rbac policy**: The Role Based Access Control (RBAC) policy of CtrlD is defined in this configuration file.
- **Startup Config**: This is the file for RBFS switch configuration.
- **TLS pem file**: This file is intended for the API Gateway (ApiGwD). The file is an X509 public/private key file in PEM format defined in the RFC7468.
- **Access JWKS file**: This file is intended for the ApiGwD. The JSON Web Key Set (JWKS) is described in the RFC 7517.

4.5.1. Trigger the ZTP process

The ZTP process in CtrlD is triggered for a specific container (LXC) on the switch. This can be triggered in the following ways.

- By the switch (RBFS Linux container) itself by sending the *operational state up* to CtrlD.
- By sending a REST request to trigger the ZTP process to CtrlD (/api/v1/ctrld/ztp/_run).

If 'load-last-config' option is set to true, ZTP is in the disabled state. ZTP is enabled if load-last-config is false.

By default, 'load-last-config' is false and ZTP is enabled. You must set to 'load-last-config' true to disable ZTP.

4.5.2. Trigger the reinstall

The reinstall of a switch can be triggered by sending a POST request to CtrlD (/api/v1/ctrld/system/_reinstall)

4.5.3. Trigger Firmware Update

The firmware update of a switch can be triggered by sending a POST request to CtrlD (/api/v1/ctrld/system/_update)

4.5.4. Management Server URL Discovery

CtrlD has to discover the management server URL to download the configuration files from the management server. Therefore, a management interface, that allows sending an DHCPINFORM request to the DHCP server, is defined.

The request contains **DHCP option 60**, that conveys the vendor class identifier "rtbrick", which informs the DHCP server about the vendor information.

The request contains the **DHCP option 61** that conveys the client identifier. The client identifier is either omitted or contains the serial number. The serial number is gathered from the ONIE file system information file `/lib/platform-config/current/host/onie-info.json`. If that does not result in a valuable result the following command is executed `dmidecode -s system-serial-number` (see /RFC2131/ and /RFC2132/ for further information).

There are at least two DHCP options requested, **DHCP option 54** that conveys the IP address of the DHCP server (see /RFC2132/ for further information), and **DHCP option 210** that conveys the path prefix for all configuration files (see /RFC5071/ for further information).

If the DHCP option 210 is not returned, CtrlD attempts to read the configurations from the IP address of the ZTP server. Otherwise, CtrlD attempts to read the configurations from the base URL specified in DHCP option 210.

4.5.5. Request configurations

The request to the Management server contains the following HTTP headers:

- ONIE-SERIAL-NUMBER: This serial number is either the onie serial number or empty string.
- CONTAINER-NAME: Container that triggered the ZTP process.
- ELEMENT-NAME: Element name that triggered the ZTP process.
- HOST-NAME: Host name of the device that triggered the ZTP process.

All this information can be used to select the right configurations for the container. This also allows the use of ZTP Configuration Process for virtual environments.

The requested URL:

- CtrlD Config: <management server url>/ztp/config/ctrld
- CtrlD rbac policy: <management server url>/ztp/config/ctrldrbac
- Startup Config: <management server url>/ztp/config/startup
- TLS pem file: <management server url>/ztp/config/tls
- Access JWKS file: <management server url>/ztp/config/accessjwks

If any of the file is not found, the process still goes forward.

4.5.6. Business Events

During the ZTP Process log messages are sent to the configured `ztp` graylog endpoint.

For more information, see the switch API documentation.

4.5.7. Overall Process Flow

The following two figures show the CtrlD ZTP process flow.

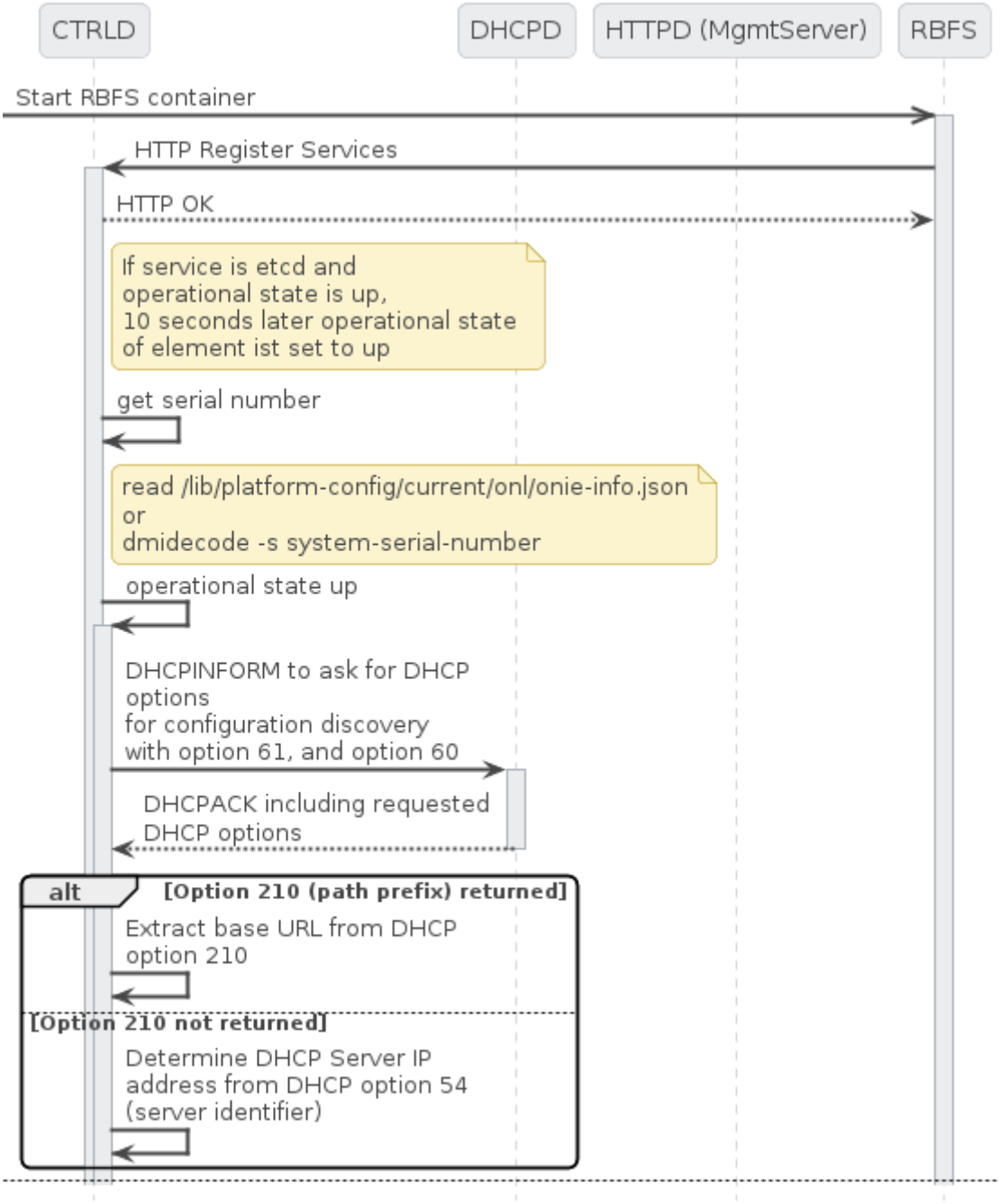


Figure 7. CTRLD ZTP process flow (Part 1/2)

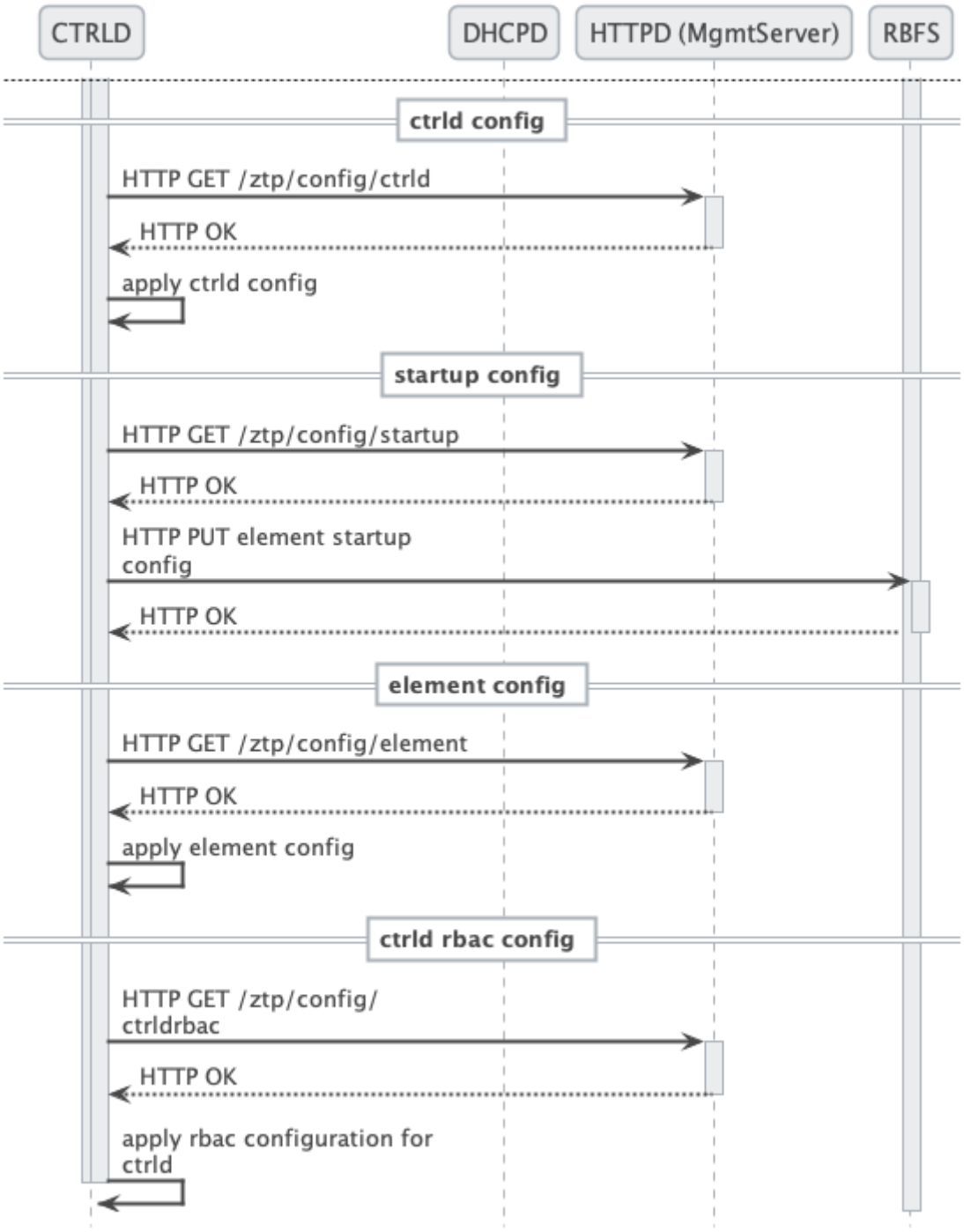


Figure 8. CTRLD ZTP process flow (Part 2/2)

4.6. References

References

ONIE	Open Network Installation Environment Image Discovery
RFC2131	RFC2131 - Dynamic Host Configuration Protocol
RFC2132	RFC2132 - DHCP Options and BOOTP Vendor Extensions https://tools.ietf.org/html/rfc2132
RFC5071	RFC5071 - Dynamic Host Configuration Protocol Options Used by PXELINUX
ISCKB	ISC Default DHCP Options

5. RBFS Licensing

5.1. Overview

RBFS Licensing allows you to access the full functionality of your RtBrick FullStack (RBFS) installation. Rtbrick provides a 28-day evaluation license on request. It is not allowed to be used in production. Use a permanent or subscription license that has been purchased through RtBrick Sales. If you want to extend the evaluation period and get additional licenses, contact RtBrick Support.

Without any license installed on your system, you can evaluate RBFS for 7 days. You need to get an evaluation license or purchase an actual license within 7 days to use the full functionality of RBFS.

5.2. Obtaining or Extending Licenses

To obtain new RBFS licenses or extend the existing licenses, go to https://portal.rtbrick.com/, click **Licenses** in the left-side menu, and then select the **Request license** link. For details, see the Managing Licenses via Self-Service Portal section below.

5.3. Managing Licenses via Self-Service Portal

The RtBrick Self-Service portal enables users to view existing license keys, request new licenses, and renew licenses that are about to expire.

5.3.1. Accessing the license key

To access the license key, click on **Licenses** in the left-side menu. This page lists your available licenses. Select the license you want to view.

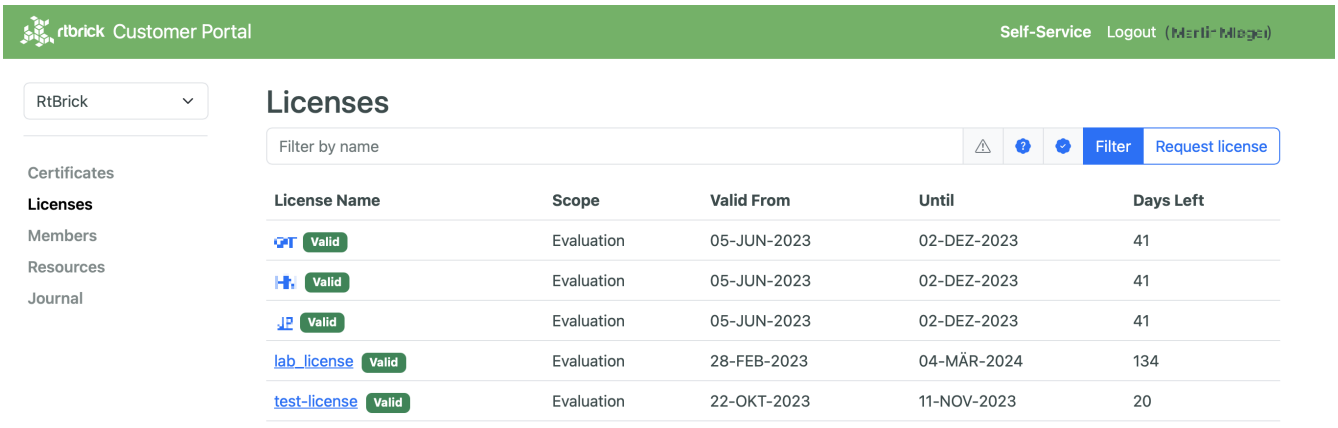


Figure 9. RtBrick License List

The detail view shows the license details including the license key.

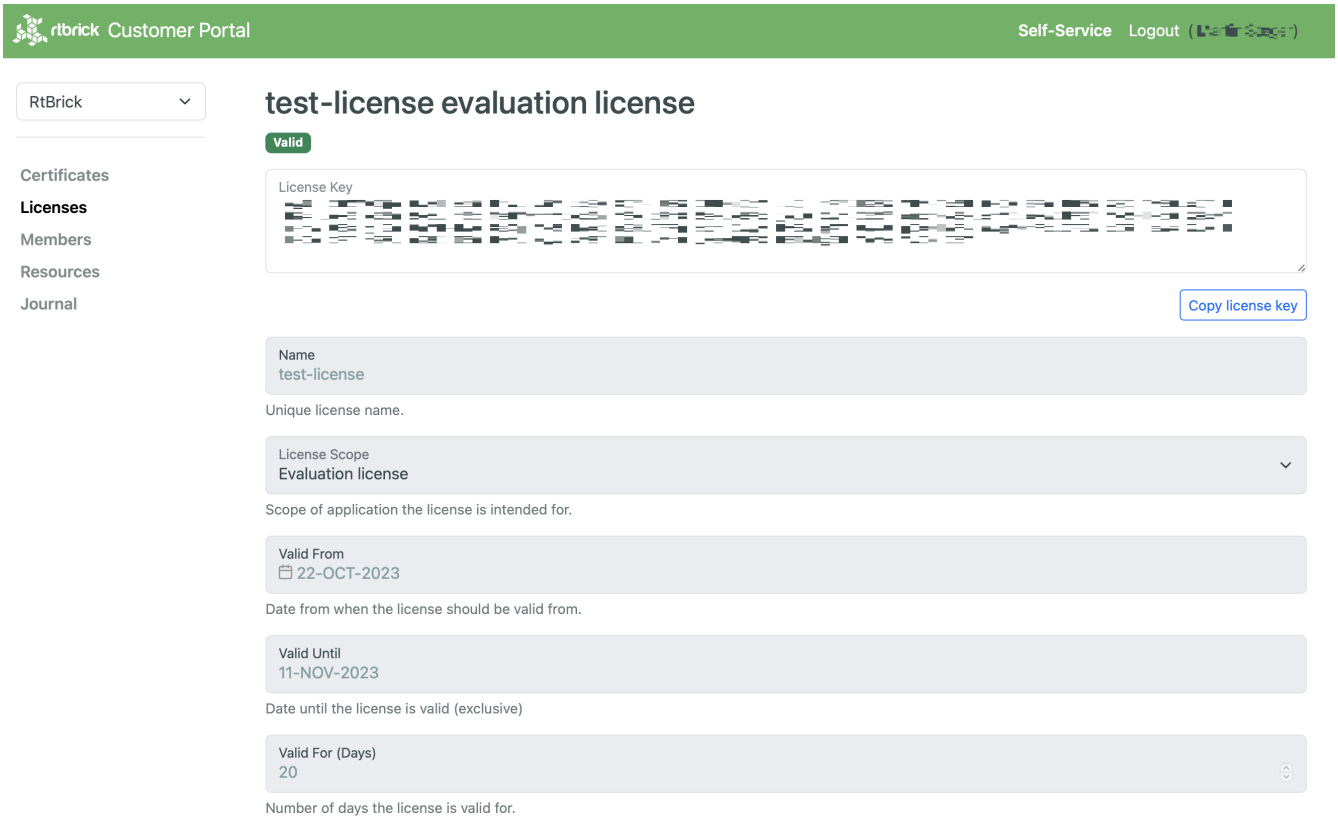


Figure 10. Approved License

Click **Copy license key** to add the license key to the clipboard.

5.3.2. Working with the license list view

The license list view allows filtering licenses by their names and lifecycle status.

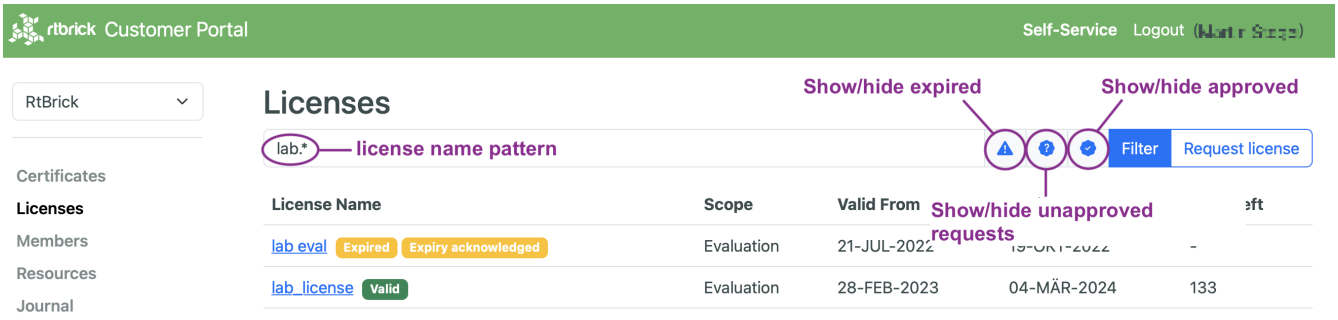


Figure 11. License Filtering

The name filter is a regular expression. The icons next to the filter allow including or excluding expired licenses, unapproved license requests and approved licenses from the license list.

5.3.3. Requesting a new license

To request a new license, proceed to licenses on the left-side menu and click the **Request license** button to request a new license.

Figure 12. RtBrick License List

Fill the license request form with all relevant data.

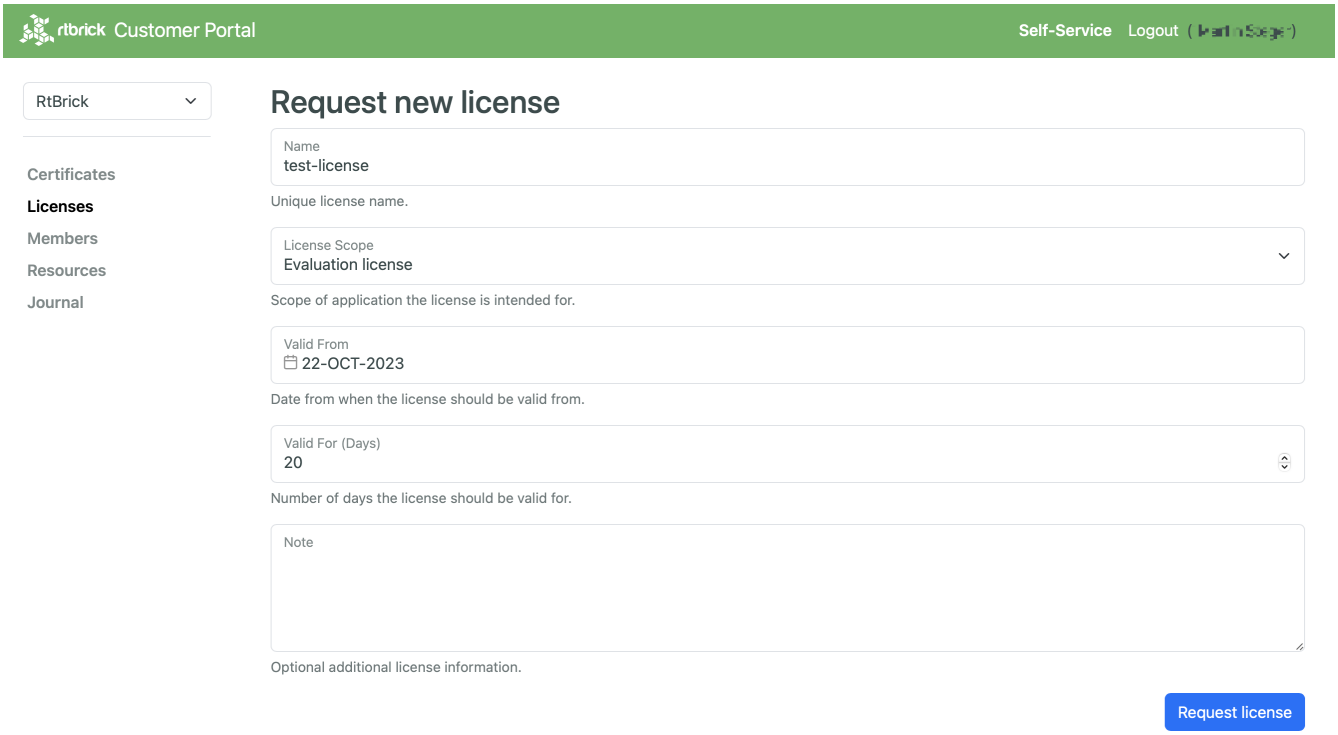


Figure 13. License Request

Click the **Request license** button to submit the license request.

5.3.4. Renewing an existing license

The portal reports when a license is about to expire. Click the **Renew** button to create a license request from the current license and copy all relevant data from the license to the license request. Once the license request has been approved, the new license and the license about to expire are both valid to give some time for deploying the new license key to the RBFS instances.

Click the **No renew** button if a license is supposed to expire and shall not be included in the expiry notifications anymore.

5.4. Installing a License

You can install a license by using the RBFS CLI or via the RESTCONF API. You should get a license encrypted string from Rtbrick and configure the same via CLI.

When you upgrade your RBFS installation, the existing license should either get restored via saved configuration or it needs to be installed again.

To install a license, enter the following command:

Syntax

set system license <license_key>

When setting the license key, ensure no other configuration changes are included. Perform the license installation as a standalone commit.

Example

```
supervisor@rtbrick: cfg> set system license
eyJzdGFydF9kYXRlIjogMTYxNTg3MTE3MCwgImVuZF9kYXRlIjogMTYxNTk1NzU3MH0=.Yx/XiFDFRzAtXPUO
aIoh5GqiXa+kOJBWp3LgDeJooVrl88mpPs2ZRMPC+k5HvoZDXvsreqRrqoFR3vk7S2PlqmLxYf0bNBly4d1hrl
oBwwFkFuJaiU/M+ZGPExgILdVyXumI88VYx8m/Z5SxEj0bFQGUy8UHRUYW/Ay8fhPfYejWuSgpv3OrIThH9CVj
lDmrp/k4yOuHyTz5gLgq4A0h33vB5O99aOIJW5UX4XDKvQqmqX5kytRlR1SseWuAbWKjUdVOkf2Mk36IbF9/xA
Kier++LzXESpLMI+MT63AybSDHOBZydoMjLH9C6cPEfGHzWTIBNtT3679Tokf25EK1Jw==
```

The following example shows the running configuration.

```
supervisor@rtbrick: cfg> show config system
{
    "rtbrick-config:system": {
        "license": [
            {
                "license-key":
"eyJzdGFydF9kYXRlIjogMTYxNTg3MTE3MCwgImVuZF9kYXRlIjogMTYxNTk1NzU3MH0=.Yx/XiFDFRzAtXPUO
aIoh5GqiXa+kOJBWp3LgDeJooVrl88mpPs2ZRMPC+k5HvoZDXvsreqRrqoFR3vk7S2PlqmLxYf0bNBly4d1hrl
oBwwFkFuJaiU/M+ZGPExgILdVyXumI88VYx8m/Z5SxEj0bFQGUy8UHRUYW/Ay8fhPfYejWuSgpv3OrIThH9CVj
lDmrp/k4yOuHyTz5gLgq4A0h33vB5O99aOIJW5UX4XDKvQqmqX5kytRlR1SseWuAbWKjUdVOkf2Mk36IbF9/xA
Kier++LzXESpLMI+MT63AybSDHOBZydoMjLH9C6cPEfGHzWTIBNtT3679Tokf25EK1Jw=="
            }
        ]
    }
}
```

5.5. Installing Multiple Licenses

You can install multiple licenses. Additional licenses can be installed even when you have existing license(s). The license with the maximum evaluation period will be prioritised over others. When you have multiple evaluation licenses installed, the one that expires later takes higher priority compared to the other licenses.

5.6. Viewing the installed license

Syntax

```
show system license
```

Example

```
root@rtbrick: cfg> show system license
License Validity:
  License 1:
    Start date : Tue Mar 16 05:06:10 GMT +0000 2021
    End date   : Wed Mar 17 05:06:10 GMT +0000 2021
root@rtbrick: cfg>
```

After verifying the validity of the license, the license file will be installed at the following location:

```
/etc/rtbrick/license/rtbrick-license
```

Syntax

Use the `show system license active` command to view the active licenses running.

```
show system license active
```

Example:

```
supervisor@rtbrick.net: cfg> show system license active
Active License Rtbrick_Internal
  Start date: Fri Mar 06 00:00:00 GMT +0000 2026
  End date: Sat Mar 06 23:59:00 GMT +0000 2027
  Days left: 338
```

To access the Operational State API that corresponds to this CLI, click here.

5.7. Deleting a License

To delete a license, enter the following command:

Syntax

```
delete system license <license_key>
```

Example

```
supervisor@rtbrick: cfg> delete system license
eyJzdGFydF9kYXRlIjogMTYxNTg3MTE3MCwgImVuZF9kYXRlIjogMTYxNTk1NzU3MH0=.Yx/XiFDFRzAtXPUO
aIoh5GqiXa+kOJBWp3LgDeJooVrl88mpPs2ZRMPC+k5HvoZDXvsreqRrqoFR3vk7S2PlqmLxYf0bNBly4d1hrl
oBwwFkFuJaiU/M+ZGPExgILdVyXumI88VYx8m/Z5SxEj0bFQGUy8UHRUYW/Ay8fhPfYejWuSgpv3OrIThH9CVj
lDmrp/k4yOuHyTz5gLgq4A0h33vB5O99aOIJW5UX4XDKvQqmqX5kytRlR1SseWuAbWKjUdVOkf2Mk36IbF9/xA
Kier++LzXESpLMI+MT63AybSDHOBZydoMjLH9C6cPEfGHzWTIBNtT3679Tokf25EK1Jw==
```

5.8. License Expiry

When a license expires, you will not be able see the operational state of the system via CLI or BDS API.

5.8.1. Restricted Configuration Commits after License Expiry

RBFS restricts system configuration when a valid license is absent, expired, or beyond the grace period. You can run only the `set system license` command, and all other `set system` commands are blocked. Also, the `set system license` command must be committed independently and cannot be combined with other configuration changes. These restrictions are applied to RESTCONF operations too.

5.8.2. License Validation

The process of verifying the validity of the software license is known as license validation. If no license is installed, a 7-day evaluation period will be provided. During this time, there will be no license validation. After the evaluation period ends, the system will check perform license validation every 12 hours. If a valid license is not found, access to the operational state of the system via CLI or BDS API will not be available.

Once a license is installed on the device, it will be validated every 12 hours. If a license is installed within 7 days of evaluation, it is considered the end of the evaluation period, and the license validation will start from that point onward.

Relevant warning or error messages will be generated based on the license validation:

- A warning is generated if the license validity is less than seven days.
- An error message is generated if the license validity is less than one day.

- A critical message is generated if the license has already expired.

Both BDS and file logs are generated for license expiry, and if the Graylog plugin is configured, they are sent to the Graylog. For a list the logs related to license expiry, refer to the section License Log Messages.

To find out the details about the license installed on your system, run the “show system license” command as explained in the section Viewing the installed license.

5.8.3. Operational Commands to View Active Licenses

You can use the show command `show system license active` to view details of the currently active license(s). The command enables you to quickly verify the license currently in use.

show system license active

Example:

The following example output displays information about the currently active license, including the license name, validity period (start date and end date), and the number of days remaining.

```
supervisor@rtbrick.net: cfg> show system license active
Active License RtBrick_Lab_v1
  Start date: Thu Nov 27 00:00:00 GMT +0000 2025
  End date: Fri Nov 27 23:59:00 GMT +0000 2026
  Days left: 312
```

Registered Address	Support	Sales
40268, Dolerita Avenue Fremont CA 94539		
+1-650-351-2251		+91 80 4850 5445
http://www.rtbrick.com	support@rtbrick.com	sales@rtbrick.com

©Copyright 2024 RtBrick, Inc. All rights reserved. The information contained herein is subject to change without notice. The trademarks, logos and service marks ("Marks") displayed in this documentation are the property of RtBrick in the United States and other countries. Use of the Marks are subject to RtBrick's Term of Use Policy, available at https://www.rtbrick.com/privacy. Use of marks belonging to other parties is for informational purposes only.