# Local User Management

**Version 22.7.1, 25 July 2022**

| Registered Address | Support | Sales |
|---|---|---|
| 26, Kingston Terrace, Princeton, New Jersey 08540, United States | | |
| | | +91 80 4850 5445 |
| http://www.rtbrick.com | support@rtbrick.com | sales@rtbrick.com |

# Table of Contents

# 1. Local User Management

Local User Management enables you to create, manage, and secure the Linux local users and groups through the RBFS configuration. This enables you to manage users and groups in the following environments:

- In the RBFS container only for the virtual platform
- In the RBFS container and on the ONL host for the hardware platforms

## 1.1. Supported Platforms

Not all features are necessarily supported on each hardware platform. Refer to the *Platform Guide* for the features and the sub-features that are or are not supported by each platform.

# 2. Configuring Local user management

RBFS allows you to create privileges that are configurable for user-defined and pre-defined roles. RBFS supports a combination of permit and deny regular expressions and a configurable default privilege to support both blacklisting and whitelisting of users. If both permit and deny command regular expressions match, the allow regular expression takes precedence.

## 2.1. Creating Roles

To create a role, you need to configure the following:

- Configure RBAC privilege for the role
- Configure the command privilege for the role

### 2.1.1. Configuring the RBAC Privilege

You need to configure the RBAC privilege for both table and object.

**set system authorization global role** <name> **rbac-permission** ( **object** | **table** ) <resource> <permission-type>

**Command arguments**

| | |
|---|---|
| <name> | Authorization role name |
| <resource> | Represents resources in the RBFS (table/object) |
| <permission-type> | Permissions to create, read and delete. The following are the supported RBAC permission types:<br>-/-/-<br>-/-/delete<br>-/read/-<br>-/read/delete<br>create/-/-<br>create/-/delete<br>create/read/-<br>create/read/delete |

## 2.1.2. Configuring the Command Privilege

**set system authorization global role** <name> **cmd-permission** ( **allow-cmds** <allow-cmds> | **deny-cmds** <deny-cmds> )

| | |
|---|---|
| <role> | Authorization role name |
| <allow-cmds> | List of allow commands regular expression |
| <deny-cmds> | List of deny commands regular expression |

- If you configure a privilege for any of the pre-defined roles (supervisor, operator, reader), then it replaces the default privilege.
- If you delete the configured privilege for a pre-defined role, then it will revert to the default privilege of the role.
- Priority of privilege rules is as follows: explicit deny, explicit permit, default privilege.

The example below shows the new role named "support" which has RBAC permission to read any table and objects. Also, the user is denied everything except the allowed commands (ping, set, show, traceroute, and watch-mode).

```
{
  "ietf-restconf:data": {
    "rtbrick-config:system": {
      "authorization": {
        "global": {
          "role": [
            {
              "name": "support",
              "rbac-permission": [
                {
                  "permission": "-/read/-",
                  "resource-type": "object",
                  "resource": ".*"
                },
                {
                  "permission": "-/read/-",
                  "resource-type": "table",
                  "resource": ".*"
                }
              ],
              "cmd-permission": {
                "allow-cmds": [
                  "ping .*",
                  "set .*",
                  "show .*",
                  "traceroute .*",
                  "watch .*"
                  ],
                "deny-cmds": ".*"
              }
            }
          ]
        }
      }
    }
  }
}
```

### Linux pre-configured users, roles and privileges

| User Name | Role Name | Default Privileges |
|---|---|---|
| supervisor | supervisor | Allow all actions |
| operator | operator | Allow all actions |

| User Name | Role Name | Default Privileges |
|---|---|---|
| reader | reader | All commands will be denied other than the commands which match any of the below regular expression.<br><br>`"color.*",`<br>`"date.*",`<br>`"exit.*",`<br>`"history.*",`<br>`"paging.*",`<br>`"ping.*",`<br>`"show.*",`<br>`"traceroute.*",`<br>`"watch-mode.*"` |

## 2.2. Creating New Users

The new users created through local user management will always have a primary group with the same name and ID of the created user. The new user's ID will be allocated within the range of 3000 and 3999.

You cannot use usernames such as root, wheel, admin, sudo or any of the SMP Linux pre-configured users and groups such as supervisor, operator, reader. Also, a username cannot start with "rtbrick_". If a Linux user with the same username already exists but has an ID outside of the 3000-3999 range then the user creation through the RBFS configuration will fail.

To create a new user, enter the following command:

**set system user** <username>

**Command arguments**

| | |
|---|---|
| <username> | Name of the local user |

## 2.3. Assigning Roles to Users

A "role" is an RBFS RBAC construct and it is mapped to a Linux group. The list of user roles from the RBFS configuration becomes the list of additional Linux groups that the Linux user belongs to. You can create new users and assign "roles" to the new users. The supervisor, operator, and reader are the pre-defined and pre-configured roles both in Linux and RBFS.

When a user is configured in RBFS under “system users”, RBFS/confd validates that the list of user roles only contain roles that are pre-defined or that are configured under “system authorization”.

Do not create role names that start with “rtbrick_”. In addition, "root", "wheel", "admin", and "sudo" are not acceptable role names.

To assign a role to a new user, enter the following command:

**set system user** <username> **role** <role>

### Command arguments

| | |
|---|---|
| <username> | Name of the user |
| <role> | Role of the user (not the primary role) |

Example: Assigning Roles to Users

```
{
  "ietf-restconf:data": {
    "rtbrick-config:system": {
      "user": [
        {
          "username": "bob",
          "role": [
            "reader"
            ],
          "shell": "/usr/local/bin/cli",
          "encrypted-password": "$6$cK31VcnMGYqHV9vnbg/6Ygh2.3uAc5P.0"
        }
      ]
    }
  }
}
```

## 2.4. Configuring Authentication for a New User

You can verify the integrity of your password using password hashing. When a user is present in the configuration but an "encrypted password" is not present, the password authentication is considered disabled for that specific user.

You can also disable password authentication for any of the predefined supervisor, operator and reader users by adding a "system users supervisor" configuration section without any "encrypted password" and thus disabling password authentication for the supervisor user.

SSH public keys can still be configured even if "encrypted password" is not present.

To create an encrypted-password and authenticate the new user, perform the following steps:

1. Generate hash password on any Linux server.
   mkpasswd --method=SHA-512
2. Configure authentication using an encrypted-password and an SSH public key.
   **set system user** <username> **encrypted-password** <encrypted-password>
   **set system user** <username> **ssh-pub-key** <ssh-pub-key>

**Command arguments**

| | |
|---|---|
| <username> | Name of the user. |
| <encrypted-password> | Password string. |
| <ssh-pub-key> | public keys of a user. You can specify multiple ssh-pub-keys. |

3. Log in using username and encrypted password.

- It is possible to change shell, password-hash and ssh-pub-keys for supervisor, reader, and operator roles.
- The password string provided as part of the RBFS configuration needs to be a compatible encrypted password string as defined by the shadow manual page: https://manpages.debian.org/buster/passwd/shadow.5.en.html and by the crypt https://manpages.debian.org/buster/manpages-dev/crypt.3.en.html.

**Example**

```
{
  "ietf-restconf:data": {
    "rtbrick-config:system": {
      "user": [
        {
          "username": "john",
          "shell": "/usr/local/bin/cli",
          "encrypted-password":
"$5$L2DaOYYuddhBV$9RA5MX9RQzLC9fIKJzbnoFBb88w9rkSXl7GVrVJ9PY7",
          "ssh-pub-key": [
            "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQCBAAABAQCubg5sdDycPN5EViNkV6w7rfp2GAfKWuInfaL3xOXyvSNps
maHILYmgrLUU0GKQH9gauPUJpDcvvYaMt0ZBuTbWHVMUc4cvhgbNDkTB2bG2cTZ5QzbicyXff3BlD
WQThVp2LtVBiW2tf7JTTa9SnL4Lnm+CQcXsQ0rxqy2S6bJpsRYlFMyQl/hZ4QEWE153dw0HGvcG8m
jfnPN4wvCc/omfD3ljxx+Gf4oFS0davX6pdphUKLvgL33VVG5xaK71imv2l3897LIJZaHxy7FbB+C
jSYT6QNq1XksX8omrbRjiP3enEQi/bANtzTNnGDnIm1KHf3xuKpoKw+B5fhDZogx"
            ]
        }
      ]
    }
  }
}
```

## 2.5. Setting the User Shell

RBFS validates that the shell is one of the following 3 valid options:

- /usr/sbin/nologin
- /bin/bash
- /usr/local/bin/cli

To configure user shell, enter the following command:

**set system user** <username> **shell** <shell>

### Command arguments

| | |
|---|---|
| <username> | Name of the user |
| <shell> | Name of the shell |

### Example

```
root@rtbrick: cfg> set system user smith shell /usr/local/bin/cli
```

## 2.6. Specifying the Display Name for User Names

The display name allows you to specify a preferred name so that you can easily identify the user. You can change your display name by entering the following command:

**set system user** <username> **display-name** <display_name>

### Command arguments

| | |
|---|---|
| <username> | Name of the user |
| <display_name> | Display name to easily identify the user |

### Example

```
set system user smith display-name primeuser
```

## 2.7. Enabling or disabling CLI access

You can control a user's access to the CLI. By default, users will have access to the CLI.

**set system user** <username> **no-cli-access** < **true** | **false** >

### Command arguments

| | |
|---|---|
| <username> | Name of the user |
| <true \| false> | When the no-cli-access is set to true, the user's access to the CLI is disabled. When the no-cli-access is set to false, the user will be able access the CLI. |

### Example

```
set system user smith no-cli-access false
```

## 2.8. Configuring sudo Without Password

You can configure local system users to log in via passwords or using SSH keys. From a security perspective, it is desirable to allow authentication with SSH keys

only. RBFS provides a configuration knob to disable the requirement for a 'sudo' password so that local users can authenticate with SSH keys only. This knob is configurable only if the user or one of its roles is supervisor.

You can enter the following command to enable or disable the 'sudo' password. By default, this is set to false which ensures that the supervisor must provide a password when using sudo.

**set system user** <user> **no-sudo-password** < **true** | **false** >

### Command arguments

| | |
|---|---|
| <username> | Name of the user |
| <true \| false> | When the no-sudo-password is set to true, it indicates that a 'sudo' password is not required. When it is set to false, it indicates that the supervisor must provide a password when using sudo. |

Example Configuration:

```
{
    "rtbrick-config:user": [
      {
        "username": "smith",
        "role": "supervisor",
        "shell": "/bin/bash",
        "ssh-pub-key": "ssh-rsa AAAAB3Nza<...>",
        "no-sudo-password": "true"
      }
    ]
 }
```

Note: If 'no-sudo-password' is set, you can log in with your SSH key.