



Red Hat Training

A Red Hat training course is available for **RHEL 8**

CHAPTER 15. WORKING WITH SYSTEMD TARGETS

systemd targets are represented by target units. Target units file ends with the `.target` file extension and their only purpose is to group together other systemd units through a chain of dependencies. For example, the `graphical.target` unit, which is used to start a graphical session, starts system services such as the GNOME Display Manager (`gdm.service`) or Accounts Service (`accounts-daemon.service`) and also activates the `multi-user.target` unit. Similarly, the `multi-user.target` unit starts other essential system services such as NetworkManager (`NetworkManager.service`) or D-Bus (`dbus.service`) and activates another target unit named `basic.target`.

This section includes procedures to implement while working with `systemd` targets.

15.1. Difference between SysV runlevels and systemd targets

The previous versions of Red Hat Enterprise Linux were distributed with SysV init or Upstart, and implemented a predefined set of runlevels that represented specific modes of operation. These runlevels were numbered from 0 to 6 and were defined by a selection of system services to be run when a particular runlevel was enabled by the system administrator. Starting with Red Hat Enterprise Linux 7, the concept of runlevels has been replaced with systemd targets.

Red Hat Enterprise Linux 7 was distributed with a number of predefined targets that are more or less similar to the standard set of runlevels from the previous releases. For compatibility reasons, it also provides aliases for these targets that directly maps to the SysV runlevels.

The following table provides a complete list of SysV runlevels and their corresponding systemd targets:

Table 15.1. Comparison of SysV runlevels with systemd targets

Runlevel	Target Units	Description
0	<code>runlevel0.target</code> , <code>poweroff.target</code>	Shut down and power off the system.
1	<code>runlevel1.target</code> , <code>rescue.target</code>	Set up a rescue shell.
2	<code>runlevel2.target</code> , <code>multi-user.target</code>	Set up a non-graphical multi-user system.
3	<code>runlevel3.target</code> , <code>multi-user.target</code>	Set up a non-graphical multi-user system.
4	<code>runlevel4.target</code> , <code>multi-user.target</code>	Set up a non-graphical multi-user system.
5	<code>runlevel5.target</code> , <code>graphical.target</code>	Set up a graphical multi-user system.
6	<code>runlevel6.target</code> , <code>reboot.target</code>	Shut down and reboot the system.

The following table compares the SysV init commands with systemctl. Use the systemctl utility to view, change, or configure systemd targets:

IMPORTANT

The `runlevel` and `telinit` commands are still available in the system and work as expected, but are only included for compatibility reasons and should be avoided.

Table 15.2. Comparison of SysV init commands with systemctl

Old Command	New Command	Description
<code>runlevel</code>	<code>systemctl list-units --type target</code>	Lists currently loaded target units.
<code>telinit <i>runLevel</i></code>	<code>systemctl isolate <i>name.target</i></code>	Changes the current target.

Additional resources

- `man sysv init`
- `man upstart init`
- `man systemctl`

15.2. Viewing the default target

The default target unit is represented by the `/etc/systemd/system/default.target` file.

Procedure

- To determine which target unit is used by default:

```
$ systemctl get-default
graphical.target
```

- To determine the default target using the symbolic link:

```
$ ls -l /usr/lib/systemd/system/default.target
```

= Viewing the target units

By default, the `systemctl list-units` command displays only active units.

Procedure

- To list all loaded units regardless of their state:

```
$ systemctl list-units --type target --all
```

- To list all currently loaded target units:

```
$ systemctl list-units --type target
```

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
basic.target	loaded	active	active	Basic System
cryptsetup.target	loaded	active	active	Encrypted Volumes
getty.target	loaded	active	active	Login Prompts
graphical.target	loaded	active	active	Graphical Interface
local-fs-pre.target	loaded	active	active	Local File Systems (Pre)
local-fs.target	loaded	active	active	Local File Systems
multi-user.target	loaded	active	active	Multi-User System
network.target	loaded	active	active	Network
paths.target	loaded	active	active	Paths
remote-fs.target	loaded	active	active	Remote File Systems
sockets.target	loaded	active	active	Sockets
sound.target	loaded	active	active	Sound Card
spice-vdagentd.target	loaded	active	active	Agent daemon for Spice guests
swap.target	loaded	active	active	Swap
sysinit.target	loaded	active	active	System Initialization
time-sync.target	loaded	active	active	System Time Synchronized
timers.target	loaded	active	active	Timers

LOAD = Reflects whether the unit definition was properly loaded.
 ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
 SUB = The low-level unit activation state, values depend on unit type.

17 loaded units listed.

15.2.1. Changing the default target

The default target unit is represented by the `/etc/systemd/system/default.target` file. The following procedure describes how to change the default target by using the `systemctl` command:

Procedure

1. To determine the default target unit:

```
# systemctl get-default
```

2. To configure the system to use a different target unit by default:

```
# systemctl set-default multi-user.target
rm /etc/systemd/system/default.target
ln -s /usr/lib/systemd/system/multi-user.target /etc/systemd/system/default.target
```

This command replaces the `/etc/systemd/system/default.target` file with a symbolic link to `/usr/lib/systemd/system/name.target`, where `name` is the name of the target unit you want to use. Replace `multi-user` with the name of the target unit you want to use by default.

3. Reboot

```
# reboot
```

15.2.2. Changing the default target using symbolic link

The following procedure describes how to change the default target by creating a symbolic link to the target.

Procedure

1. To determine the default target unit:

```
# ls /usr/lib/systemd/system/default.target -l
```

2. To create a symbolic link:

```
# ln -sf /usr/lib/systemd/system/graphical.target /etc/systemd/system/default.target
```

3. Reboot the system:

```
# reboot
```

Verification steps

- Verify the newly created default.target:

```
$ systemctl get-default
multi-user.target
```

= Changing the current target

This procedure explains how to change the target unit in the current session using the systemctl command.

Procedure

- To change to a different target unit in the current session:

```
# systemctl isolate multi-user.target
```

This command starts the target unit named *multi-user* and all dependent units, and immediately stops all others.

Replace *multi-user* with the name of the target unit you want to use by default.

Verification steps

- Verify the newly created default.target:

```
$ systemctl get-default
multi-user.target
```

15.2.2.1. Booting to rescue mode

Rescue mode provides a convenient single-user environment and allows you to repair your system in situations when it is unable to complete a regular booting process. In rescue mode, the system attempts to mount all local file systems and start some important system services, but it does not activate network interfaces or allow more users to be logged into the system at the same time.

Procedure

- To change the current target and enter rescue mode in the current session:

```
# systemctl rescue

Broadcast message from root@localhost on pts/0 (Fri 2013-10-25 18:23:15 CEST):

The system is going down to rescue mode NOW!
```

NOTE

This command is similar to `systemctl isolate rescue.target`, but it also sends an informative message to all users that are currently logged into the system.

To prevent `systemd` from sending a message, run the following command with the `--no-wall` command-line option: `# systemctl --no-wall rescue`

15.2.2.2. Booting to emergency mode

Emergency mode provides the most minimal environment possible and allows you to repair your system even in situations when the system is unable to enter rescue mode. In emergency mode, the system mounts the root file system only for reading, does not attempt to mount any other local file systems, does not activate network interfaces, and only starts a few essential services.

Procedure

- To change the current target and enter emergency mode:

```
# systemctl emergency
```

NOTE

This command is similar to `systemctl isolate emergency.target`, but it also sends an informative message to all users that are currently logged into the system.

To prevent systemd from sending this message, run the following command with the `--no-wall` command-line option: `# systemctl --no-wall emergency`



Chapter 14. Managing system services with systemctl

Chapter 16. Shutting down, suspending, and hibernating the system

