Name

systemd.target — Target unit configuration

Synopsis

target.target

Description

A unit configuration file whose name ends in ".target" encodes information about a target unit of systemd, which is used for grouping units and as well-known synchronization points during start-up.

This unit type has no specific options. See systemd.unit(5) for the common options of all unit configuration files. The common configuration items are configured in the generic [Unit] and [Install] sections. A separate [Target] section does not exist, since no target-specific options may be configured.

Target units do not offer any additional functionality on top of the generic functionality provided by units. They exist merely to group units via dependencies (useful as boot targets), and to establish standardized names for synchronization points used in dependencies between units. Among other things, target units are a more flexible replacement for SysV runlevels in the classic SysV init system. (And for compatibility reasons special target units such as runlevel3.target exist which are used by the SysV runlevel compatibility code in systemd. See systemd.special(7) for details).

Automatic Dependencies

Implicit Dependencies

There are no implicit dependencies for target units.

Default Dependencies

The following dependencies are added unless DefaultDependencies=no is set:

- Target units will automatically complement all configured dependencies of type Wants= or Requires= with dependencies of type After= unless DefaultDependencies=no is set in the specified units. Note that Wants= or Requires= must be defined in the target unit itself — if you for example define Wants=some.target in some.service, the automatic ordering will not be added.

- Target units automatically gain Conflicts= and Before= dependencies against shutdown.target.

Options

Target unit files may include [Unit] and [Install] sections, which are described in systemd.unit(5). No options specific to this file type are supported.

Example

Example 1. Simple standalone target

```
# emergency-net.target

[Unit]
Description=Emergency Mode with Networking
Requires=emergency.target systemd-networkd.service
After=emergency.target systemd-networkd.service
AllowIsolate=yes
```

When adding dependencies to other units, it's important to check if they set `DefaultDependencies=`. Service units, unless they set `DefaultDependencies=no`, automatically get a dependency on `sysinit.target`. In this case, both `emergency.target` and `systemd-networkd.service` have `DefaultDependencies=no`, so they are suitable for use in this target, and do not pull in `sysinit.target`.

You can now switch into this emergency mode by running `systemctl isolate emergency-net.target` or by passing the option `systemd.unit=emergency-net.target` on the kernel command line.

Other units can have `WantedBy=emergency-net.target` in the `[Install]` section. After they are enabled using **systemctl enable**, they will be started before `emergency-net.target` is started. It is also possible to add arbitrary units as dependencies of `emergency.target` without modifying them by using **systemctl add-wants**.

See Also

systemd(1), systemctl(1), systemd.unit(5), systemd.special(7), systemd.directives(7)