NAME

systemd.special - Special systemd units

SYNOPSIS

basic.target, bluetooth.target, ctrl-alt-del.target, cryptsetup.target, cryptsetup-pre.target, dbus.service, dbus.socket, default.target, display-manager.service, emergency.target, exit.target, final.target, getty.target, graphical.target, halt.target, hibernate.target, hybrid-sleep.target, initrd-fs.target, kbrequest.target, kexec.target, local-fs.target, local-fs-pre.target, multi-user.target, network.target, network-online.target, network-pre.target, nss-lookup.target, nss-user-lookup.target, paths.target, poweroff.target, printer.target, reboot.target, remote-fs.target, remote-fs-pre.target, rescue.target, initrd-root-device.target, initrd-root-fs.target, rpcbind.target, runlevel2.target, runlevel3.target, runlevel4.target, runlevel5.target, shutdown.target, sigpwr.target, sleep.target, slices.target, smartcard.target, sockets.target, sound.target, suspend.target, swap.target, sysinit.target, syslog.socket, system-update.target, time-sync.target, timers.target, umount.target, -.slice, system.slice, user.slice, machine.slice

DESCRIPTION

A few units are treated specially by systemd. They have special internal semantics and cannot be renamed.

SPECIAL SYSTEM UNITS

basic.target

A special target unit covering basic boot-up.

systemd automatically adds dependency of the type *After=* for this target unit to all services (except for those with *DefaultDependencies=no*).

Usually, this should pull-in all local mount points plus /var, /tmp and /var/tmp, swap devices, sockets, timers, path units and other basic initialization necessary for general purpose daemons. The mentioned mount points are special cased to allow them to be remote.

This target usually does not pull in any non-target units directly, but rather does so indirectly via other early boot targets. It is instead meant as a synchronization point for late boot services. Refer to **bootup(7)** for details on the targets involved.

ctrl-alt-del.target

systemd starts this target whenever Control+Alt+Del is pressed on the console. Usually, this should be aliased (symlinked) to reboot.target.

cryptsetup.target

A target that pulls in setup services for all encrypted block devices.

dbus.service

A special unit for the D-Bus bus daemon. As soon as this service is fully started up systemd will connect to it and register its service.

dbus.socket

A special unit for the D-Bus system bus socket. All units with *Type=dbus* automatically gain a dependency on this unit.

default.target

The default unit systemd starts at bootup. Usually, this should be aliased (symlinked) to multi-user.target or graphical.target.

The default unit systemd starts at bootup can be overridden with the *systemd.unit=* kernel command line option.

display-manager.service

The display manager service. Usually, this should be aliased (symlinked) to gdm.service or a similar display manager service.

emergency.target

A special target unit that starts an emergency shell on the main console. This target does not pull in any services or mounts. It is the most minimal version of starting the system in order to acquire an

interactive shell; the only processes running are usually just the system manager (PID 1) and the shell process. This unit is supposed to be used with the kernel command line option *systemd.unit=*; it is also used when a file system check on a required file system fails, and boot-up cannot continue. Compare with rescue.target, which serves a similar purpose, but also starts the most basic services and mounts all file systems.

Use the "systemd.unit=emergency.target" kernel command line option to boot into this mode. A short alias for this kernel command line option is "emergency", for compatibility with SysV.

In many ways booting into emergency.target is similar to the effect of booting with "init=/bin/sh" on the kernel command line, except that emergency mode provides you with the full system and service manager, and allows starting individual units in order to continue the boot process in steps.

exit.target
    A special service unit for shutting down the system or user service manager. It is equivalent to poweroff.target on non-container systems, and also works in containers.

    systemd will start this unit when it receives a request to shut down over D-Bus or a **SIGTERM** or **SIGINT** signal when running as user service daemon.

    Normally, this (indirectly) pulls in shutdown.target, which in turn should be conflicted by all units that want to be scheduled for shutdown when the service manager starts to exit.

final.target
    A special target unit that is used during the shutdown logic and may be used to pull in late services after all normal services are already terminated and all mounts unmounted.

getty.target
    A special target unit that pulls in statically configured local TTY getty instances.

graphical.target
    A special target unit for setting up a graphical login screen. This pulls in multi-user.target.

    Units that are needed for graphical logins shall add *Wants=* dependencies for their unit to this unit (or multi-user.target) during installation. This is best configured via *WantedBy=graphical.target* in the unit's "[Install]" section.

hibernate.target
    A special target unit for hibernating the system. This pulls in sleep.target.

hybrid-sleep.target
    A special target unit for hibernating and suspending the system at the same time. This pulls in sleep.target.

halt.target
    A special target unit for shutting down and halting the system. Note that this target is distinct from poweroff.target in that it generally really just halts the system rather than powering it down.

    Applications wanting to halt the system should start this unit.

initrd-fs.target
    systemd-fstab-generator(3) automatically adds dependencies of type *Before=* to sysroot-usr.mount and all mount points found in /etc/fstab that have **x-initrd.mount** and not have **noauto** mount options set.

kbrequest.target
    systemd starts this target whenever Alt+ArrowUp is pressed on the console. This is a good candidate to be aliased (symlinked) to rescue.target.

kexec.target
    A special target unit for shutting down and rebooting the system via kexec.

    Applications wanting to reboot the system with kexec should start this unit.

local-fs.target

systemd-fstab-generator(3) automatically adds dependencies of type *Before=* to all mount units that refer to local mount points for this target unit. In addition, it adds dependencies of type *Wants=* to this target unit for those mounts listed in /etc/fstab that have the **auto** mount option set.

multi-user.target
> A special target unit for setting up a multi-user system (non-graphical). This is pulled in by graphical.target.
>
> Units that are needed for a multi-user system shall add *Wants=* dependencies for their unit to this unit during installation. This is best configured via *WantedBy=multi-user.target* in the unit's "[Install]" section.

network-online.target
> Units that strictly require a configured network connection should pull in network-online.target (via a *Wants=* type dependency) and order themselves after it. This target unit is intended to pull in a service that delays further execution until the network is sufficiently set up. What precisely this requires is left to the implementation of the network managing service.
>
> Note the distinction between this unit and network.target. This unit is an active unit (i.e. pulled in by the consumer rather than the provider of this functionality) and pulls in a service which possibly adds substantial delays to further execution. In contrast, network.target is a passive unit (i.e. pulled in by the provider of the functionality, rather than the consumer) that usually does not delay execution much. Usually, network.target is part of the boot of most systems, while network-online.target is not, except when at least one unit requires it. Also see **Running Services After the Network is up**[1] for more information.
>
> All mount units for remote network file systems automatically pull in this unit, and order themselves after it. Note that networking daemons that simply provide functionality to other hosts generally do not need to pull this in.

paths.target
> A special target unit that sets up all path units (see **systemd.path(5)** for details) that shall be active after boot.
>
> It is recommended that path units installed by applications get pulled in via *Wants=* dependencies from this unit. This is best configured via a *WantedBy=paths.target* in the path unit's "[Install]" section.

poweroff.target
> A special target unit for shutting down and powering off the system.
>
> Applications wanting to power off the system should start this unit.
>
> runlevel0.target is an alias for this target unit, for compatibility with SysV.

reboot.target
> A special target unit for shutting down and rebooting the system.
>
> Applications wanting to reboot the system should start this unit.
>
> runlevel6.target is an alias for this target unit, for compatibility with SysV.

remote-fs.target
> Similar to local-fs.target, but for remote mount points.
>
> systemd automatically adds dependencies of type *After=* for this target unit to all SysV init script service units with an LSB header referring to the "$remote_fs" facility.

rescue.target
> A special target unit that pulls in the base system (including system mounts) and spawns a rescue shell. Isolate to this target in order to administer the system in single-user mode with all file systems mounted but with no services running, except for the most basic. Compare with emergency.target, which is much more reduced and does not provide the file systems or most basic services.

runlevel1.target is an alias for this target unit, for compatibility with SysV.

Use the "systemd.unit=rescue.target" kernel command line option to boot into this mode. A short alias for this kernel command line option is "1", for compatibility with SysV.

initrd-root-device.target
A special initrd target unit that is reached when the root filesystem device is available, but before it has been mounted. **systemd-fstab-generator(3)** and **systemd-gpt-auto-generator(3)** automatically setup the appropriate dependencies to make this happen.

initrd-root-fs.target
systemd-fstab-generator(3) automatically adds dependencies of type *Before=* to the sysroot.mount unit, which is generated from the kernel command line.

runlevel2.target, runlevel3.target, runlevel4.target, runlevel5.target
These are targets that are called whenever the SysV compatibility code asks for runlevel 2, 3, 4, 5, respectively. It is a good idea to make this an alias for (i.e. symlink to) graphical.target (for runlevel 5) or multi-user.target (the others).

shutdown.target
A special target unit that terminates the services on system shutdown.

Services that shall be terminated on system shutdown shall add *Conflicts=* and *Before=* dependencies to this unit for their service unit, which is implicitly done when *DefaultDependencies=yes* is set (the default).

sigpwr.target
A special target that is started when systemd receives the SIGPWR process signal, which is normally sent by the kernel or UPS daemons when power fails.

sleep.target
A special target unit that is pulled in by suspend.target, hibernate.target and hybrid-sleep.target and may be used to hook units into the sleep state logic.

slices.target
A special target unit that sets up all slice units (see **systemd.slice(5)** for details) that shall be active after boot. By default the generic user.slice, system.slice, machines.slice slice units, as well as the root slice unit -.slice are pulled in and ordered before this unit (see below).

It's a good idea to add *WantedBy=slices.target* lines to the "[Install]" section of all slices units that may be installed dynamically.

sockets.target
A special target unit that sets up all socket units (see **systemd.socket(5)** for details) that shall be active after boot.

Services that can be socket-activated shall add *Wants=* dependencies to this unit for their socket unit during installation. This is best configured via a *WantedBy=sockets.target* in the socket unit's "[Install]" section.

suspend.target
A special target unit for suspending the system. This pulls in sleep.target.

swap.target
Similar to local-fs.target, but for swap partitions and swap files.

sysinit.target
systemd automatically adds dependencies of the types *Requires=* and *After=* for this target unit to all services (except for those with *DefaultDependencies=no*).

This target pulls in the services required for system initialization. System services pulled in by this target should declare *DefaultDependencies=no* and specify all their dependencies manually, including access to anything more than a read only root filesystem. For details on the dependencies of this target, refer to **bootup(7)**.

syslog.socket

> The socket unit syslog implementations should listen on. All userspace log messages will be made available on this socket. For more information about syslog integration, please consult the **Syslog Interface**[2] document.

system-update.target

> A special target unit that is used for off-line system updates. **systemd-system-update-generator(8)** will redirect the boot process to this target if /system-update exists. For more information see the **System Updates Specification**[3].

timers.target

> A special target unit that sets up all timer units (see **systemd.timer(5)** for details) that shall be active after boot.

> It is recommended that timer units installed by applications get pulled in via *Wants=* dependencies from this unit. This is best configured via *WantedBy=timers.target* in the timer unit's "[Install]" section.

umount.target

> A special target unit that unmounts all mount and automount points on system shutdown.

> Mounts that shall be unmounted on system shutdown shall add Conflicts dependencies to this unit for their mount unit, which is implicitly done when *DefaultDependencies=yes* is set (the default).

SPECIAL SYSTEM UNITS FOR DEVICES

Some target units are automatically pulled in as devices of certain kinds show up in the system. These may be used to automatically activate various services based on the specific type of the available hardware.

bluetooth.target

> This target is started automatically as soon as a Bluetooth controller is plugged in or becomes available at boot.

> This may be used to pull in Bluetooth management daemons dynamically when Bluetooth hardware is found.

printer.target

> This target is started automatically as soon as a printer is plugged in or becomes available at boot.

> This may be used to pull in printer management daemons dynamically when printer hardware is found.

smartcard.target

> This target is started automatically as soon as a smartcard controller is plugged in or becomes available at boot.

> This may be used to pull in smartcard management daemons dynamically when smartcard hardware is found.

sound.target

> This target is started automatically as soon as a sound card is plugged in or becomes available at boot.

> This may be used to pull in audio management daemons dynamically when audio hardware is found.

SPECIAL PASSIVE SYSTEM UNITS

A number of special system targets are defined that can be used to properly order boot-up of optional services. These targets are generally not part of the initial boot transaction, unless they are explicitly pulled in by one of the implementing services. Note specifically that these *passive* target units are generally not pulled in by the consumer of a service, but by the provider of the service. This means: a consuming service should order itself after these targets (as appropriate), but not pull it in. A providing service should order itself before these targets (as appropriate) and pull it in (via a *Wants=* type dependency).

Note that these passive units cannot be started manually, i.e. "systemctl start time-sync.target" will fail with an error. They can only be pulled in by dependency. This is enforced since they exist for ordering purposes

only and thus are not useful as only unit within a transaction.

cryptsetup-pre.target
>   This passive target unit may be pulled in by services that want to run before any encrypted block
>   device is set up. All encrypted block devices are set up after this target has been reached. Since the
>   shutdown order is implicitly the reverse start-up order between units, this target is particularly useful
>   to ensure that a service is shut down only after all encrypted block devices are fully stopped.

local-fs-pre.target
>   This target unit is automatically ordered before all local mount points marked with **auto** (see above). It
>   can be used to execute certain units before all local mounts.

network.target
>   This unit is supposed to indicate when network functionality is available, but it is only very weakly
>   defined what that is supposed to mean, with one exception: at shutdown, a unit that is ordered after
>   network.target will be stopped before the network — to whatever level it might be set up then — is
>   shut down. It is hence useful when writing service files that require network access on shutdown,
>   which should order themselves after this target, but not pull it in. Also see **Running Services After
>   the Network is up**[1] for more information. Also see network-online.target described above.
>
>   systemd automatically adds dependencies of type *After=* for this target unit to all SysV init script
>   service units with an LSB header referring to the "$network" facility.

network-pre.target
>   This passive target unit may be pulled in by services that want to run before any network is set up, for
>   example for the purpose of setting up a firewall. All network management software orders itself after
>   this target, but does not pull it in.

nss-lookup.target
>   A target that should be used as synchronization point for all host/network name service lookups. Note
>   that this is independent of user/group name lookups for which nss-user-lookup.target should be used.
>   All services for which the availability of full host/network name resolution is essential should be
>   ordered after this target, but not pull it in. systemd automatically adds dependencies of type *After=* for
>   this target unit to all SysV init script service units with an LSB header referring to the "$named"
>   facility.

nss-user-lookup.target
>   A target that should be used as synchronization point for all user/group name service lookups. Note
>   that this is independent of host/network name lookups for which nss-lookup.target should be used. All
>   services for which the availability of the full user/group database is essential should be ordered after
>   this target, but not pull it in. Note that system users are always resolvable, and hence do not require
>   any special ordering against this target.

remote-fs-pre.target
>   This target unit is automatically ordered before all remote mount point units (see above). It can be
>   used to run certain units before the remote mounts are established. Note that this unit is generally not
>   part of the initial transaction, unless the unit that wants to be ordered before all remote mounts pulls it
>   in via a *Wants=* type dependency. If the unit wants to be pulled in by the first remote mount showing
>   up, it should use network-online.target (see above).

rpcbind.target
>   The portmapper/rpcbind pulls in this target and orders itself before it, to indicate its availability.
>   systemd automatically adds dependencies of type *After=* for this target unit to all SysV init script
>   service units with an LSB header referring to the "$portmap" facility.

time-sync.target
>   Services responsible for synchronizing the system clock from a remote source (such as NTP client
>   implementations) should pull in this target and order themselves before it. All services where correct
>   time is essential should be ordered after this unit, but not pull it in. systemd automatically adds
>   dependencies of type *After=* for this target unit to all SysV init script service units with an LSB header

referring to the "$time" facility.

SPECIAL USER UNITS

When systemd runs as a user instance, the following special units are available, which have similar definitions as their system counterparts: exit.target, default.target, shutdown.target, sockets.target, timers.target, paths.target, bluetooth.target, printer.target, smartcard.target, sound.target.

SPECIAL PASSIVE USER UNITS

graphical−session.target

This target is active whenever any graphical session is running. It is used to stop user services which only apply to a graphical (X, Wayland, etc.) session when the session is terminated. Such services should have "PartOf=graphical-session.target" in their "[Unit]" section. A target for a particular session (e. g. gnome-session.target) starts and stops "graphical-session.target" with "BindsTo=graphical-session.target".

Which services are started by a session target is determined by the "Wants=" and "Requires=" dependencies. For services that can be enabled independently, symlinks in ".wants/" and ".requires/" should be used, see **systemd.unit(5)**. Those symlinks should either be shipped in packages, or should be added dynamically after installation, for example using "systemctl add-wants", see **systemctl(1)**.

Example 1. Nautilus as part of a GNOME session

"gnome-session.target" pulls in Nautilus as top-level service:

[Unit]
Description=User systemd services for GNOME graphical session
Wants=nautilus.service
BindsTo=graphical-session.target

"nautilus.service" gets stopped when the session stops:

[Unit]
Description=Render the desktop icons with Nautilus
PartOf=graphical-session.target

[Service]
...

graphical−session−pre.target

This target contains services which set up the environment or global configuration of a graphical session, such as SSH/GPG agents (which need to export an environment variable into all desktop processes) or migration of obsolete d-conf keys after an OS upgrade (which needs to happen before starting any process that might use them). This target must be started before starting a graphical session like gnome-session.target.

SPECIAL SLICE UNITS

There are four ".slice" units which form the basis of the hierarchy for assignment of resources for services, users, and virtual machines or containers.

-.slice
    The root slice is the root of the hierarchy. It usually does not contain units directly, but may be used to set defaults for the whole tree.

system.slice
    By default, all system services started by **systemd** are found in this slice.

user.slice
    By default, all user processes and services started on behalf of the user, including the per-user systemd instance are found in this slice.

machine.slice
    By default, all virtual machines and containers registered with **systemd-machined** are found in this

slice.

SEE ALSO

systemd(1), **systemd.unit(5)**, **systemd.service(5)**, **systemd.socket(5)**, **systemd.target(5)**, **systemd.slice(5)**, **bootup(7)**, **systemd-fstab-generator(8)**

NOTES

1. Running Services After the Network is up
   http://www.freedesktop.org/wiki/Software/systemd/NetworkTarget

2. Syslog Interface
   http://www.freedesktop.org/wiki/Software/systemd/syslog

3. System Updates Specification
   http://freedesktop.org/wiki/Software/systemd/SystemUpdates