



Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'systemd.slice.5' command

\$ man systemd.slice.5

SYSTEMD.SLICE(5) systemd.slice SYSTEMD.SLICE(5)

NAME

systemd.slice - Slice unit configuration

SYNOPSIS

slice.slice

DESCRIPTION

A unit configuration file whose name ends in ".slice" encodes information about a slice unit. A slice unit is a concept for hierarchically managing resources of a group of processes. This management is performed by creating a node in the Linux Control Group (cgroup) tree. Units that manage processes (primarily scope and service units) may be assigned to a specific slice. For each slice, certain resource limits may be set that apply to all processes of all units contained in that slice. Slices are organized hierarchically in a tree. The name of the slice encodes the location in the tree. The name consists of a dash-separated series of names, which describes the path to the slice from the root slice. The root slice is named -.slice. Example: foo-bar.slice is a slice that is located within foo.slice, which in turn is located in the root slice -.slice. Note that slice units cannot be templated, nor is possible to add multiple names to a slice unit by creating additional symlinks to its unit file.

By default, service and scope units are placed in system.slice, virtual

machines and containers registered with `systemd-machined(8)` are found in `machine.slice`, and user sessions handled by `systemd-logind(8)` in `user.slice`. See `systemd.special(7)` for more information.

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. The slice specific configuration options are configured in the `[Slice]` section. Currently, only generic resource control settings as described in `systemd.resource-control(5)` are allowed.

See the [New Control Group Interfaces\[1\]](#) for an introduction on how to make use of slice units from programs.

AUTOMATIC DEPENDENCIES

Implicit Dependencies

The following dependencies are implicitly added:

- ? Slice units automatically gain dependencies of type `After=` and `Requires=` on their immediate parent slice unit.

Default Dependencies

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

- ? Slice units will automatically have dependencies of type `Conflicts=` and `Before=` on `shutdown.target`. These ensure that slice units are removed prior to system shutdown. Only slice units involved with late system shutdown should disable `DefaultDependencies=` option.

OPTIONS

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

SEE ALSO

`systemd(1)`, `systemd.unit(5)`, `systemd.resource-control(5)`,
`systemd.service(5)`, `systemd.scope(5)`, `systemd.special(7)`,
`systemd.directives(7)`

NOTES

<https://www.freedesktop.org/wiki/Software/systemd/ControlGroupInterface>

systemd 252

SYSTEMD.SLICE(5)