

Comando

systemctl list-units
systemctl list-unit-files
systemctl is-enabled <
systemctl is-active <
systemctl daemon-reexec
systemctl daemon-reload
systemctl show <
systemctl show-environment
systemctl set-environment <
systemctl unset-environment <
systemctl mask <
systemctl unmask <
systemctl edit <
systemctl get-default
systemctl set-default <
systemctl isolate <
systemctl list-dependencies <
systemctl list-jobs
systemctl cancel <
systemctl kill <
systemctl log-level [nivel]
systemctl log-target [target]
systemctl default
systemctl poweroff
systemctl reboot
systemctl halt
systemctl suspend
systemctl hibernate
systemctl hybrid-sleep
systemctl emergency
systemctl rescue
systemctl add-wants <
systemctl add-requires <
systemctl list-machines
systemctl enable --now <
systemctl disable --now <
systemctl preset-all
systemctl reenable <
systemctl reload-or-restart <
systemctl condrestart <
systemctl help <
systemctl is-failed <
systemctl list-timers
systemctl show-identity
systemctl machine-id

Descripción

Muestra las unidades actualmente cargadas (servicios, sockets, dispositivos, etc.)

Lista todas las unidades disponibles en disco (incluye .service"

unidad>

unidad>

Recarga completamente el daemon systemd (útil después de cambios en archivos de configuración)

Recarga las configuraciones de las unidades sin reiniciar el daemon

unidad>

Muestra variables de entorno usadas por systemd

VAR=VAL>

VAR>

unidad>

unidad>

unidad>

Muestra el objetivo (target) predeterminado del sistema

target>

target>

unidad>

Muestra trabajos pendientes o en ejecución

job-id>

unidad>

Consulta o establece el nivel de registro del sistema

Consulta o establece el destino de los registros (console, journal, kmsg, null)

Inicia el objetivo predeterminado (usado al arrancar sin argumentos)

Apaga el sistema

Reinicia el sistema

Detiene el sistema sin apagarlo

Suspende el sistema (modo hibernación ligera)

Hiberna el sistema (guarda estado en disco)

Combina suspensión y hibernación

Entra en modo de emergencia (solo root shell)

Entra en modo rescate (entorno mínimo con shell)

target>

target>

Muestra máquinas virtuales o contenedores registrados

unidad>

unidad>

Aplica las configuraciones predeterminadas de todas las unidades (según /etc/systemd/system-preset/)

unidad>

unidad>

unidad>

unidad>

unidad>

Muestra todos los temporizadores activos

Muestra información del host según systemd (UUID, nombre, etc.)

Muestra el ID único del sistema (asociado a /etc/machine-id)

Ejemplo de Uso

```
systemctl list-units --type=service  
.socket
```

```
sudo systemctl daemon-reexec  
sudo systemctl daemon-reload
```

```
systemctl show-environment
```

```
systemctl get-default
```

```
systemctl list-jobs
```

```
sudo systemctl log-level debug  
sudo systemctl log-target journal  
sudo systemctl default  
sudo systemctl poweroff  
sudo systemctl reboot  
sudo systemctl halt  
sudo systemctl suspend  
sudo systemctl hibernate  
sudo systemctl hybrid-sleep  
sudo systemctl emergency  
sudo systemctl rescue  
&lt  
&lt  
systemctl list-machines
```

```
sudo systemctl preset-all
```

```
systemctl list-timers  
systemctl show-identity  
systemctl machine-id
```

.timer

Verifica si una unidad está habilitada para iniciar al arranque

Verifica si una unidad está activa actualmente

Muestra propiedades detalladas de una unidad

Establece variables de entorno globales para systemd

Elimina una variable de entorno establecida previamente

Enmascara una unidad para evitar que se active (crea un symlink a /dev/null)

Desenmascara una unidad previamente enmascarada

Crea o modifica un fragmento de sobrescribir para una unidad (sin editar el archivo original)

Establece el objetivo predeterminado del sistema

Cambia al objetivo especificado inmediatamente (similar a cambiar de runlevel)

Muestra las dependencias de la unidad (qué otras unidades requiere o son requeridas por ella)

Cancela un trabajo específico en cola

Envía una señal a un proceso asociado a una unidad

unidad>

unidad>

Habilita e inicia una unidad en un solo comando

Deshabilita y detiene una unidad en un solo comando

Deshabilita y vuelve a habilitar una unidad (útil tras modificar unit file)

Si es posible, recarga; si no, reinicia la unidad

Reinicia la unidad solo si ya está corriendo

Abre el manual relacionado con la unidad (si existe)

Verifica si una unidad ha fallado

```
etc.)"
systemctl is-enabled ssh.service
systemctl is-active nginx.service

systemctl list-unit-files --type=timer

systemctl show ssh.service

sudo systemctl set-environment DEBUG=1
sudo systemctl unset-environment DEBUG
sudo systemctl mask cups.service
sudo systemctl unmask cups.service
sudo systemctl edit ssh.service

sudo systemctl set-default graphical.target
sudo systemctl isolate multi-user.target
systemctl list-dependencies nginx.service

sudo systemctl cancel 1234
sudo systemctl kill nginx.service
```

Añade una dependencia "	Wants"
Añade una dependencia "	Requires"

```
sudo systemctl enable --now docker.service
sudo systemctl disable --now bluetooth.service

sudo systemctl reenable cron.service
sudo systemctl reload-or-restart nginx.service
sudo systemctl condrestart httpd.service
systemctl help sshd.service
systemctl is-failed apache2.service
```

table-8bff0826-651d-4b48-87ce-4ec712da6633-4

a un objetivo	sudo systemctl add-wants multi-user.target myservice.service
a un objetivo	sudo systemctl add-requires graphical.target myservice.service