

# Systemd: Targets

Laboratorio de Sistemas Operativos II

# Systemd

## Gestor del Sistema y sus Servicios

- Tiene como componentes a las unidades que representan los diferentes recursos del sistema.
- Targets son un tipo de unidad de Systemd

“

Los Targets son importantes para la gestión del estado del sistema, incluyendo el inicio de servicios, el manejo de dependencias entre servicios, y el manejo de los distintos modos de funcionamiento del sistema.

# Concepto de Targets

- ▶ Son unidades que **agrupan** otras unidades (como servicios, dispositivos, etc.) y controlan cuándo se inician o se terminan y su orden para lograr un nivel de ejecución del sistema operativo.
- ▶ Un **nivel de ejecución** del sistema operativo especifica el estado del sistema y define los servicios que se inician.
- ▶ No realizan acciones por sí mismos, sino que coordina otras unidades para alcanzar un estado deseado.
- ▶ Los archivos de los targets llevan el sufijo .target.

# Para qué sirven los Targets

- ▶ **Organizar el arranque del sistema** - > varias unidades deben iniciarse en orden y en grupos específicos.
- ▶ **Controlar el estado del sistema** -> definir cuándo un sistema está en diferentes estados como “mantenimiento”, “sin interfaz gráfica”, etc.
- ▶ **Facilitar tareas de gestión**: iniciar, detener o reiniciar grupos de servicios juntos.

# Archivos de configuración

- ▶ En **/etc/systemd/system** se encuentran los archivos de configuración de las distintas unidades.
- ▶ Los archivos de los targets llevan el sufijo **.target**.
- ▶ Para ver los targets:
  - ▶ `#systemctl list-units --all --type=target`

# Targets comunes en Debian 12

- ▷ **poweroff.target** -> Apaga el sistema.
- ▷ **rescue.target** -> Entra en modo de rescate.
- ▷ **multi-user.target** -> Modo de usuario multiusuario (sin interfaz gráfica).
- ▷ **graphical.target** -> Modo gráfico (con interfaz gráfica).
- ▷ **reboot.target** -> Reinicia el sistema.

# Herramientas para targets

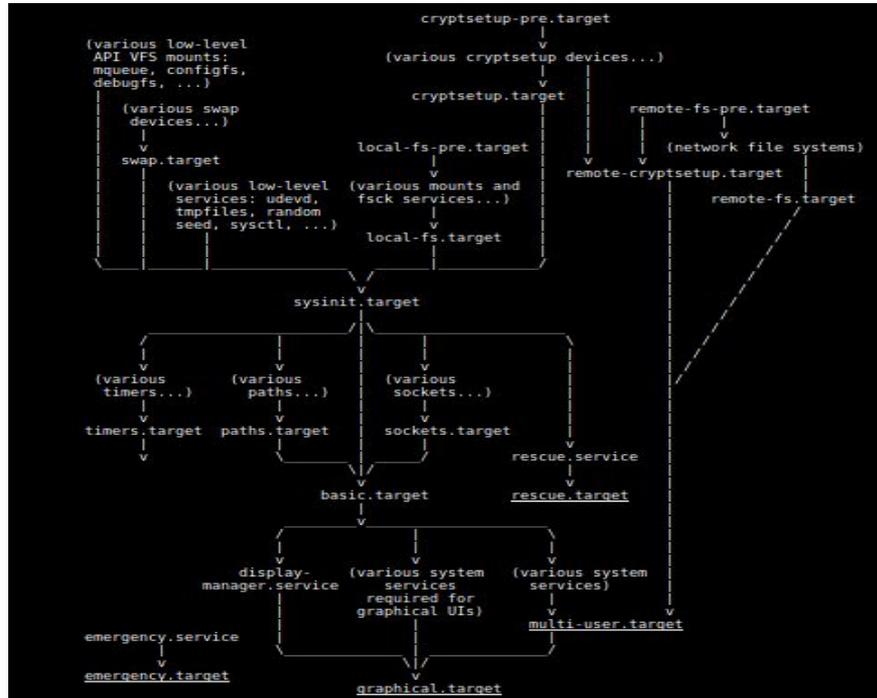
- ▶ Ver target actual (default.target)
  - ▶ `#systemctl get-default`
- ▶ Cambiar target actual (sin afectar el siguiente arranque)
  - ▶ `# systemctl isolate nombre.target`
- ▶ Cambiar target para próximo arranque
  - ▶ `#systemctl set-default nombre.target`

**Operación:** Elimina `/etc/systemd/system/default.target`.  
Crea symlink `/etc/systemd/system/default.target -> /usr/lib/systemd/system/graphical.target`.

- ▶ Cuando systemd inicia el sistema, activa todas las unidades que dependen de default.target (así como, recursivamente, todas las dependencias de estas dependencias).
- ▶ Normalmente, default.target es simplemente un alias de graphic.target o multi-user.target, dependiendo de si el sistema está configurado para una interfaz gráfica o solo para una consola de texto.

# Comando man bootup

Muestra la secuencia general de inicio de los eventos en el arranque del sistema systemd, y los requerimientos básicos de orden para asegurar un inicio limpio y sin complicaciones.



# Dependencia de targets

- ▶ Si el target activo es multi-user.target
- ▶ Para funcionar correctamente necesita que basic.target esté activado. Es decir, antes de entrar en multi-user.target, todos los servicios agrupados en basic.target (inicio del sistema) deben haberse iniciado. A su vez, basic.target requiere sysinit.target.
- ▶ En general: Si en A.target se indica Requires=B.target, hablamos que el target B es una dependencia del A.
- ▶ Si bien systemd tiene como uno de sus objetivos de diseño iniciar servicios en paralelo, hay ciertos procesos y targets que deben ser ejecutados antes que otros por una cuestión de dependencia.

# Wants y After

- ▶ En los archivos .target se puede encontrar líneas que comienzan con las palabras Requires, Wants y After (/lib/systemd/system), controlan cómo se relacionan y en qué orden se inician las unidades.

```
root@nuna:~# cat /lib/systemd/system/basic.target
# This file is part of systemd.
[Unit]
Description=Basic System
Documentation=man:systemd.special(7)
Requires=sysinit.target
Wants=sockets.target timers.target paths.target slices.target
After=sysinit.target sockets.target paths.target slices.target
tmp.mount
```

# Qué significan

- ▶ **Requires** indica los servicios que deben estar activos obligatoriamente cuando se inicia este target.
- ▶ **Wants** es similar a Requires. Se utiliza para indicar las unidades que se deben iniciar preferentemente (no obligatoria) . Si alguna de ellas produce algún error, eso no ocasiona una falla en el funcionamiento del target considerado.
- ▶ **After** se utiliza para indicar un orden de inicio más que una lista de dependencias (también existe un Before, aunque no aparece en nuestro ejemplo actual).
- ▶ Entonces, `basic.target` indica que se deben iniciar `sockets.target`, `timers.target`, etc. Además, `basic.target` se iniciará luego (after) de `sysinit.target`, `sockets.target`, etc.

# Ejemplo

servicio\_base.service (un servicio base imprescindible)

- ▶ servicio\_opcional.service (un servicio opcional)
- ▶ servicio\_final.service

▶ En el Archivo de creación del target: **/etc/systemd/system/mi-target.target**

```
Description=Target Personalizado Agrupando Servicios
Requires=servicio_base.service servicio_final.service
Wants=servicio_opcional.service
After=servicio_base.service servicio_opcional.service
servicio_final.service
```

# ¡Gracias!

## ¿Preguntas?

Referencias:

<https://blog.carreralinux.com.ar/2018/03/targets-en-systemd-introduccion/>

[https://wiki.archlinux.org/index.php/Systemd\\_\(Espa%C3%B1ol\)#Targets](https://wiki.archlinux.org/index.php/Systemd_(Espa%C3%B1ol)#Targets)

<https://geekland.eu/systemctl-administrar-servicios-linux/>

<https://juncotic.com/systemd-ejecutando-un-script-al-inicio-de-gnu-linux/>