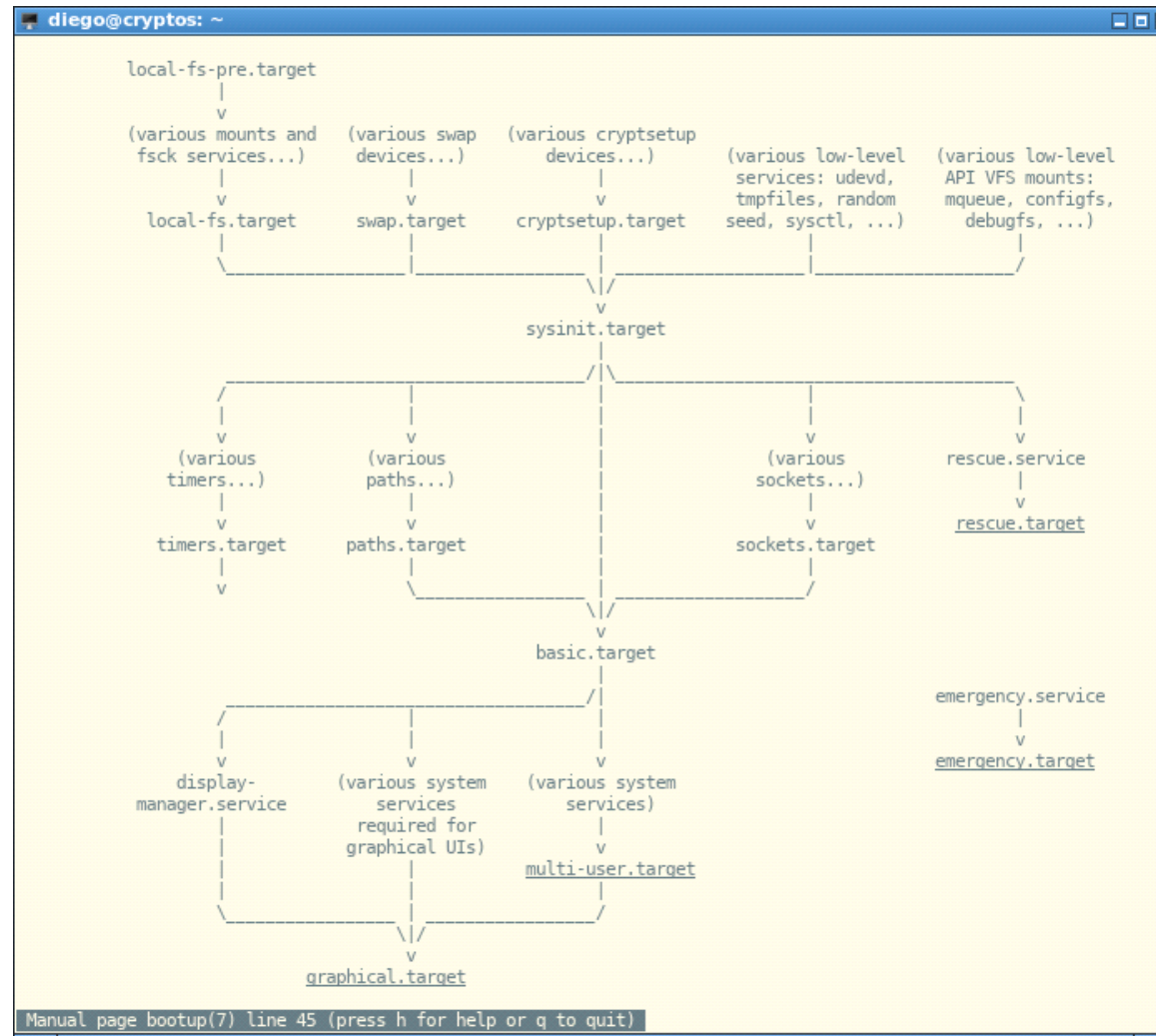


# MANEJO DE LA CONFIGURACIÓN DEL SISTEMA SERVICIO SYSTEMD

# Arranque del sistema

1. Arranque de la BIOS
2. Cargador de arranque (MBR)
3. GRUB2 arranca
  - Carga la imagen del núcleo
    - Gestión de la memoria: supervisa cuánta memoria se utiliza para almacenar qué tipo de elementos, así como el lugar en que los guarda.
    - Gestión de los procesos: determina qué procesos pueden usar la unidad central de procesamiento (CPU), cuándo y durante cuánto tiempo.
    - Controladores de dispositivos: actúa como mediador o intérprete entre el hardware y los procesos.
    - Seguridad y llamadas al sistema: recibe solicitudes de servicio por parte de los procesos.
  - Extrae el contenido de initramfs (contiene las herramientas y guiones necesarios para montar el sistema de archivos antes de que se lance el binario **init** en el sistema de archivos raíz real)

4. El núcleo carga los módulos de los drivers necesarios desde initrd.
5. Se ejecuta systemd.



# Modos de ejecución del sistema

Descripción	Denominación: <ul style="list-style-type: none"><li>- Según SysV init system</li><li>- Según systemd (utilizado por CentOS 7)</li></ul>	Orden para que el sistema se ejecute en un modo específico: <ul style="list-style-type: none"><li>- Según SysV init system</li><li>- Según systemd (utilizado por CentOS 7)</li></ul>
Modo apagado del sistema (halt)	<ul style="list-style-type: none"><li>- Level 0</li><li>- <b>poweroff.target</b></li></ul>	<pre># init 0 # systemctl isolate poweroff.target</pre>
Modo mantenimiento (implica modo monousuario)	<ul style="list-style-type: none"><li>- Level 1</li><li>- <b>rescue.target</b></li></ul>	<pre># init 1 # systemctl isolate rescue.target</pre>
Modo multiusuario sin servicios de red	<ul style="list-style-type: none"><li>- Level 2</li><li>- <b>No definido específicamente</b></li></ul>	
Modo multiusuario con servicios de red	<ul style="list-style-type: none"><li>- Level 3</li><li>- <b>multi-user.target</b></li></ul>	<pre># init 3 # systemctl isolate multi-user.target</pre>
Modo multiusuario completo con entorno gráfico (X11)	<ul style="list-style-type: none"><li>- Level 5</li><li>- <b>graphical.target</b></li></ul>	<pre># init 5 # systemctl isolate graphical.target</pre>
Modo reinicio del sistema (apagado inmediato y seguidamente inicio)	<ul style="list-style-type: none"><li>- Level 6</li><li>- <b>reboot.target</b></li></ul>	<pre># init 6 # systemctl isolate reboot.target</pre>

# ¿QUÉ PAPEL JUEGA **systemd** EN EL SISTEMA?

- **systemd** es un conjunto de bloques básicos de compilación para un sistema Linux. Proporciona un gestor de sistemas y servicios que se ejecuta como PID 1 e inicia el resto del sistema.
- Sistemas que lo utilizan:
  - Debian
  - Fedora
  - Mandriva
  - Mageia
  - Ubuntu
  - **CentOS 7, 8**
  - RHEL 7.0
  - OpenSUSE
  - Arch Linux

# CARACTERÍSTICAS DE **systemd**

- Es compatible con SysV (init).
- Con respecto a **SysV**, aumenta la velocidad de inicialización del sistema.
- Asume que cualquier dispositivo puede conectarse/desconectarse en cualquier momento (**hotplug**).
- Utiliza la activación de demonios vía socket, permitiendo así la paralelización en la ejecución de éstos.
- Todos los servicios y procesos que maneja se registran (**journal**) mediante Linux Control Groups (**cgroups**).
- Los servicios se agrupan (**subsistemas**), posibilitando su administración conjunta.

# LAS UNIDADES (UNITS) EN `systemd`

- En `systemd` una unidad se refiere a cualquier recurso que el sistema sabe como operar y gestionar.
- Las unidades se encuentran en:
  - `"/usr/lib/systemd/system"`: archivos de configuración por defecto del sistema
  - `"/etc/systemd/system"`: archivos de configuración del administrador del sistema que anulan los archivos de configuración predeterminados del sistema operativo
  - `"/run/systemd/system"`: archivos de configuración generados durante la ejecución que anulan los archivos de configuración instalados.

# TIPOS DE UNIDADES DE SERVICIO (UNITS) DE `systemd`

- **Tipo service.** Es un demonio que puede ser arrancado, parado, reiniciado o recargado.
- **Tipo socket.** Un socket de tipo fichero o de tipo TCP/IP. Cada de este tipo debe tener una unidad de servicio correspondiente.
- **Tipo device.** Un archivo de tipo dispositivo.
- **Tipo mount.** Un punto de montaje..
- **Tipo automount.** Un punto de montaje automático. Cada unidad automount tiene una unidad mount correspondiente, que se inicia al acceder al directorio de automontaje.
- **Tipo target.** Agrupación lógica de unidades. Integra a otras unidades, que pueden ser controladas conjuntamente.
- **Tipo snapshot.** Agrupación de unidades utilizadas para obtener instantáneas del sistema.



# Servicios en systemd

Cuando arranca systemd, éste arranca aquellos servicios incluidos en los subdirectorios .wants de los grupos de arranque correspondientes.

Por ejemplo

- En el directorio [root@localhost graphical.target.wants]# ls -l /usr/lib/systemd/system/graphical.target.wants
  - total 0
  - lrwxrwxrwx. 1 root root 39 ago 10 11:21 systemd-update-utmp-runlevel.service -> ../systemd-update-utmp-runlevel.service
- 
- [root@localhost graphical.target.wants]# ls -l /etc/systemd/system/graphical.target.wants
  - total 4
  - lrwxrwxrwx. 1 root root 47 jun 12 12:59 accounts-daemon.service -> /usr/lib/systemd/system/accounts-daemon.service
  - lrwxrwxrwx. 1 root root 61 jun 12 13:02 initial-setup-reconfiguration.service -> /usr/lib/systemd/system/initial-setup-reconfiguration.service
  - lrwxrwxrwx. 1 root root 44 jun 12 13:00 rtkit-daemon.service -> /usr/lib/systemd/system/rtkit-daemon.service
  - lrwxrwxrwx. 1 root root 39 jun 12 13:02 udisks2.service -> /usr/lib/systemd/system/udisks2.service

# Servicios en systemd

Cuando arranca systemd, éste arranca aquellos servicios incluidos en los subdirectorios .wants de los grupos de arranque correspondientes.

Por ejemplo

- En el directorio [root@localhost graphical.target.wants]# ls -l /usr/lib/systemd/system/graphical.target.wants
  - total 0
  - lrwxrwxrwx. 1 root root 39 ago 10 11:21 systemd-update-utmp-runlevel.service -> ../systemd-update-utmp-runlevel.service
- 
- [root@localhost graphical.target.wants]# ls -l /etc/systemd/system/graphical.target.wants
  - total 4
  - lrwxrwxrwx. 1 root root 47 jun 12 12:59 accounts-daemon.service -> /usr/lib/systemd/system/accounts-daemon.service
  - lrwxrwxrwx. 1 root root 61 jun 12 13:02 initial-setup-reconfiguration.service -> /usr/lib/systemd/system/initial-setup-reconfiguration.service
  - lrwxrwxrwx. 1 root root 44 jun 12 13:00 rtkit-daemon.service -> /usr/lib/systemd/system/rtkit-daemon.service
  - lrwxrwxrwx. 1 root root 39 jun 12 13:02 udisks2.service -> /usr/lib/systemd/system/udisks2.service

# MANEJANDO **systemd** (I)

- La orden para administrar los servicios controlados por **systemd** es **systemctl**

Obtener los servicios que están instalados en el sistema

```
# systemctl -t service list-unit-files -all
```

Obtener los servicios que se están ejecutando

```
# systemctl -t service list-units -all
```

# MANEJANDO `systemd` (II)

Sintaxis general:

**# `systemctl` *acción* [*opcion*] *Nombre\_Servicio***

Opciones communes del argumento Acción:

`start` (para iniciar)

`stop` (para detener)

`restart` (para reiniciar)

`reload` (para recargar)

`status` (mostrar información de estado del servicio)

`enable` (configurar que el servicio debe iniciarse en el arranque del sistema)

`disable` (el servicio deja de iniciarse en el arranque del sistema)

# MANEJANDO `systemd` (III)

Obtener los “targets” del sistema

```
# systemctl list-units -type=target
```

Cambiar de modo de ejecución (target)

```
# systemctl isolate graphical.target
```

# UN EJEMPLO: INSTALACIÓN DEL SERVICIO COCKPIT (I)

- El servicio cockpit es una herramienta para la administración remota de un sistema.
- Permite:
  - Monitorizar recursos del sistema remoto.
  - Manejar el entorno de seguridad proporcionado por Selinux del sistema remoto.
  - Interactuar con el sistema remoto a través de un a consola de órdenes.
- Principios de funcionamiento:
  - La comunicación es mediante un socket TCP por el puerto 9090.
  - Se trata de un servicio que utiliza el protocolo http (web service) para conectarse con el usuario remoto.
  - El usuario remoto sólo requiere de un navegador para utilizar este servicio. Navegadores compatibles:
    - Mozilla Firefox
    - Google Chrome
    - Microsoft Edge
    - Apple Safari
    - Opera

# UN EJEMPLO: INSTALACIÓN DEL SERVICIO COLCKPIT (II)

- Instalación del servicio en el sistema a administrar remotamente:

Paso 1: # yum install cockpit

Paso 2: # systemctl start cockpit

Paso 3: # systemctl enable cockpit.socket

Paso 4: # firewall-cmd --permanent --zone=public --add-service=cockpit

Paso 5: # firewall-cmd -reload

- En el sistema en el que opera el usuario
  - En el navegador, introducir la siguiente url: `https://SERVER_IP:9090`
  - Autenticarse con un usuario con privilegios de administrador en el equipo a administrar.