

Sistema de arranque linux

Daniel García Moreno (danigm)
dani@danigm.net

Sobre mí (danigm)

- Ex-alumno de la US (2005-2008)
- Ex-sugus
- Usuario de linux desde ~2003
- Desarrollador de GNOME
- Hack Computer OS Engineer, <https://hack-computer.com/>

Arranque en Linux

BIOS	Basic Input/Output System. Instalado en la placa base, un sistema básico que carga el MBR
MBR	Master Boot Record, Registro especial al inicio del disco duro donde suele estar el sistema de arranque (GRUB / LILO)
GRUB	GRand Unified Bootloader, Te permite elegir diferentes configuraciones para arrancar diferentes sistemas operativos
Kernel	Linux, se carga en memoria y lanza el proceso de arranque, /sbin/init con PID=1
Init / Systemd	Lanza diferentes procesos de arranque del sistema operativo

Proceso Init

- Proceso padre, PID=1
- Se ejecuta desde el arranque hasta el apagado
- Si falla al ejecutarse, kernel panic
- Es el padre de todos los procesos de primer nivel. Si un proceso se queda sin padre, este pasa a ser su padre

init (SysVinit)

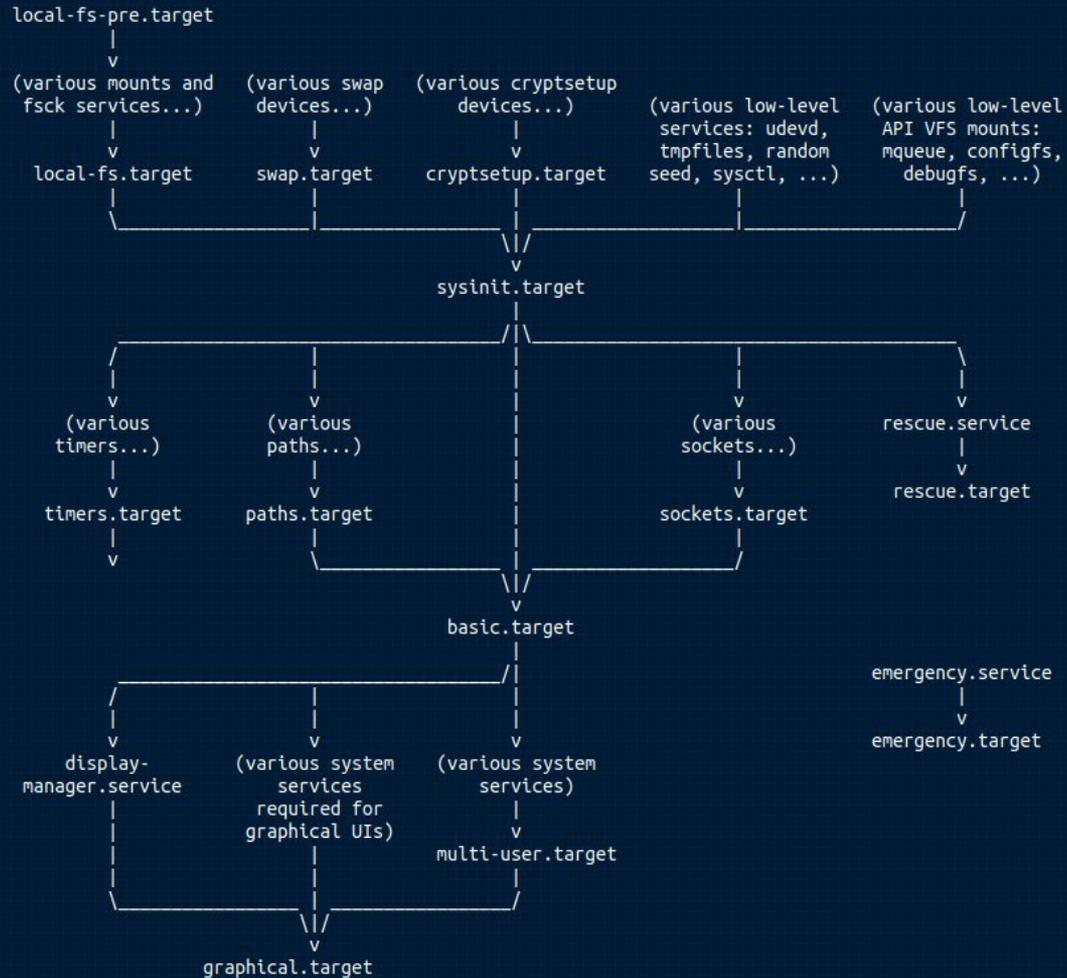
- Sistema simple
- Diferentes runlevels definidos en `/etc/inittab`
- Ficheros de lanzamiento como script bash en `/etc/init.d/`
- Ejecuta todos los scripts para un determinado runlevel de forma secuencial, hasta que un proceso no termina, no se lanza el siguiente

Systemd

- Sistema más complejo, se encarga del arranque, pero también monitoriza los procesos, relanza si fallan, gestiona el log del sistema, etc.
- Arranque de procesos de forma paralela
- Gestión de dependencias de forma automática

Pasos de systemd

1. Monta el sistema de ficheros según `/etc/fstab`
2. Utiliza `/etc/systemd/system/default.target` para saber qué debe hacer
3. Un target define una serie de dependencias que systemd tendrá que levantar, esas dependencias pueden tener a su vez dependencias
- 4.



Systemd hace más cosas aparte del arranque

- journald: Demonio responsable de la gestión de logs, en formato binario
- logind: Demonio para la gestión de sesiones de usuario
- networkd: Demonio para la gestión de la configuración de interfaces de red
- tmpfiles: Gestión de ficheros temporales
- timedated: Demonio para la gestión de la hora del sistema
- udevd: Gestión de dispositivos para linux (/dev)

systemd Utilities

systemctl journalctl notify analyze cgls cgtop loginctl nspawn

systemd Daemons

systemd
journald networkd
logind user-session

systemd Targets

bootmode basic multi-user graphical user-session
dbus telephony display service
shutdown reboot dlog logind user-session
tizen service

systemd Core

manager service timer mount target unit
systemd snapshot path socket swap login
multiseat inhibit namespace log
session pam cgroup dbus

systemd Libraries

dbus-1 libpam libcap libcryptsetup tcpwrapper libaudit libnotify

Linux Kernel

cgroups autofs kdbus

Comparativa

	Init	Systemd
Inicio de servicios en paralelo	No	Sí
Licencia	GPL v2+	LGPL v2.1+
Dependencias	-	D-Bus
Limitar recursos por servicio	No	Sí
Cálculo de dependencias entre servicios	No	Sí

Detractores de Systemd

- Sistema complejo, que quiere hacerlo todo, no sigue los principios de diseño UNIX de hacer una sola cosa y hacerla bien
- Incompatible con scripts shell
- Systemd es sólo para escritorio

<http://0pointer.de/blog/projects/the-biggest-myths.html>

Systemd como supervisor de servicios

```
# systemctl status sshd.service
```

```
# systemctl start/stop/restart sshd.service
```

```
# systemctl list-unit-files
```

```
# systemctl enable/disable service
```

```
# systemctl list-dependencies sshd.service
```

Configuración systemd

Se pueden configurar los diferentes servicios que lanzará systemd y otros detalles a través de ficheros de configuración .ini

- `/etc/systemd/system`, `/usr/lib/systemd/system`
- Ficheros `.service` que definen cómo lanzar o gestionar un servicio

Ejemplo de definición de servicio

```
[danigm@rincewind decide] $ cat decide.service
```

```
[Unit]
```

```
Description=decide
```

```
[Service]
```

```
User=decide
```

```
Type=simple
```

```
PIDFile=/var/run/decide.pid
```

```
WorkingDirectory=/home/decide/decide/decide
```

```
ExecStart=/home/decide/venv/bin/gunicorn -w 5 decide.wsgi --timeout=500 -b 0.0.0.0:8000
```

```
Restart=always
```

```
[Install]
```

```
WantedBy=multi-user.target
```

Journald, log de sistema

- Log en formato binario, información estructurada
- Ocupa menos espacio y no hay necesidad de rotar logs o buscar cadenas en ficheros de texto

```
# journalctl
# journalctl -f
# journalctl -u sshd
# journalctl --since "20 min ago"
```

Fin