

systemd



IES Gonzalo Nazareno
CONSEJERÍA DE EDUCACIÓN

Alberto Molina Coballes



12 de diciembre de 2017

Orígenes de systemd

- Se plantea como sustituto de System V init (PID=1)
- Para obtener un arranque rápido en paralelo de los servicios en el espacio de usuario
- Comienza su desarrollo en 2010
- Compite inicialmente con upstart que comenzó a desarrollarse en 2006
- Englobado en el proyecto freedesktop.org:
<https://www.freedesktop.org/wiki/Software/systemd/>



Systemd ganó

- upstart fue inicialmente adoptado por muchas distros
- Debian: Proposal: switch init system to systemd or upstart y Debate initsystem upstart
- Adopción de systemd:
 - Fedora: Mayo 2011
 - Arch Linux: Octubre 2012
 - RHEL y CentOS: Junio/Julio 2014
 - Ubuntu: Abril 2015
 - Debian: Abril 2015

Mark Shuttleworth: Losing graciously



Funcionalidades iniciales

- Sustituye `/sbin/init`
- Sustituye los niveles de ejecución (*run-levels*)
- Controla los demonios (olvídate de `/etc/init.d/...` o `service`)
- Desaparece `/etc/inittab`
- Sustituye `/etc/fstab`
- Se encarga de borrar los directorios temporales
- Gestiona los servicios en paralelo
- Gestiona los procesos con `cgroups`
- Soporta instantáneas y restauración

<http://0pointer.de/blog/projects/why.html>



Pero ...

Los desarrolladores de systemd pensaron que una vez puestos a sustituir se podría seguir controlando:

- Las sesiones de usuario y las TTYs (systemd-logind)
- Syslog (systemd-journald)
- Dispositivos (systemd-udev)
- Variables del kernel (systemd-sysctl)
- Sesiones de usuario (systemd-user-sessions)
- Puntos de montaje (*.mount)
- Control de eventos como alternativa a cron (*.timer)
- Hora del sistema (systemd-timedated)
- Contenedores (systemd-nspawn)

Además consideran que systemd es un sistema en continuo desarrollo

Y algunos todavía no utilizados por todas las distros

- Interfaces de red (systemd-networkd)
- Chequeo del sistema de ficheros (systemd-fsck)
- Resolución de nombres (systemd-resolved)
- Volcados de memoria (coredumps)
- Cuotas (systemd-quotacheck)
- Y lo que te rondaré morena



Críticas

Systemd ha recibido feroces críticas:

- No solo sustituye init, se ha convertido en un supersistema interno que está reemplazando muchos componentes esenciales
- Rompe con la filosofía UNIX de pequeñas aplicaciones que realizan muy bien una tarea interactuando entre sí
- Incluye cada vez más dependencias que hacen difícil tener un sistema sin systemd
- Es exclusivo de Linux y no utilizable en otros Unices
- Incluso ha habido forks: <https://devuan.org/>

Características principales

- Activación de servicios basados en sockets
- Activación de servicios basados en D-Bus
- Activación de servicios basados en dispositivos
- Activación de servicios basados en ficheros
- Montaje y automontaje
- Compatible con SysV init

Red Hat System Administrator's Guide



Conceptos de systemd. Unidades

Unidad (*unit*) Elemento básico de systemd asociado a un fichero de configuración

La instrucción principal para interactuar con systemd es `systemctl`:

- Información de systemd:

```
systemctl status
```

- Unidades que han fallado:

```
systemctl --failed [--all]
```

- Listar unidades en memoria:

```
systemctl list-units
```

- Listar unidades instaladas:

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



Conceptos de systemd. Unidades

Las unidades se ubican en diferentes directorios:

- `/lib/systemd/system/` Al instalar los diferentes paquetes, estén o no habilitadas
- `/run/systemd/` Creadas durante la ejecución del sistema. Tiene prioridad sobre el anterior
- `/etc/systemd/system/` Unidades habilitadas

Conceptos de systemd. Servicios

Las unidades encargadas de la gestión de servicios tienen la extensión `.service` y se gestionan con `systemctl`:

```
systemctl start foo | foo.service
systemctl stop foo | foo.service
systemctl restart foo | foo.service
systemctl status foo | foo.service
systemctl enable foo | foo.service
systemctl disable foo | foo.service
systemctl mask foo | foo.service
systemctl unmask foo | foo.service
systemctl is-enabled foo | foo.service
systemctl is-active foo | foo.service
systemctl is-failed foo | foo.service
```

Conceptos de systemd. Targets

Target Conjunto de unidades. Los niveles de ejecución se sustituyen en systemd por *targets*, pero se usan con más fines. Cada distro puede definir sus propios *targets*

- *basic.target*: Unidades para el inicio del sistema
- *multi-user.target*: Agrupa la mayoría de los servicios no esenciales. Equivale a los niveles de ejecución 2-4
- *graphical.target*: Agrupa los servicios del entorno gráfico. Equivalente al nivel de ejecución 5

```
systemctl get-default
systemctl list-units --type target [--all]
systemctl set-default nombre.target
systemctl isolate nombre.target
systemctl [--no-wall] rescue
systemctl [--no-wall] emergency
```



Conceptos de systemd. Porciones

Porción (*slice*) Utilizado para gestionar grupos de procesos creando nodos en el árbol de Linux cgroups, por lo que se pueden aplicar límites a los recursos utilizados.

- `system.slice`: Utilizado por defecto por servicios
- `user.slice`: Jerarquía de porciones de recursos de usuarios (los hijos se representan con ‘-’)
- `user-[] .slice`: Sesiones de usuarios gestionadas por systemd-logind

```
systemctl list-units --type slice [--all]
```



Conceptos de systemd. Puntos de montaje

Punto de montaje (*mount*) Define unidades para los puntos de montaje de los sistemas de ficheros en unidades con extensión `.mount`. Sustituye así las entradas de `/etc/fstab`, pero por compatibilidad permite definir allí las entradas que se traducen en tiempo de ejecución a unidades en `/run/systemd`

automount Para montaje automático (requiere una unidad `.mount` y otra `.automount`)

Intercambio (*swap*) Define unidades para los dispositivos o ficheros de *swap*. Al igual que en el caso anterior, pueden seguir definiéndose en `/etc/fstab` y se crean las correspondientes unidades durante la ejecución del sistema.

Conceptos. Temporizadores

Temporizador (*timer*) Como alternativas a la definición de procesos de cron. Hay de dos tipos:

- Definidos conforme a fecha fija, como los *cronjobs*
- Definido por un intervalo de tiempo tras un evento.

```
systemctl list-timers
```

NEXT	LEFT	LAST	PASSED	UNIT	ACTIVATES
Sun ...	3h 35min left	Sun ...	10h ago	apt-daily.timer	apt-daily.service
Mon ...	18h left	Sun ...	4h 47min ago	apt-daily-upgra...	apt-daily-upgrade.service
Mon ...	22h left	Sun ...	1h 29min ago	systemd-tmpfile...	systemd-tmpfiles-clean.ser

Apagar/suspender/hibernar/reiniciar

systemd también controla las funciones de apagado y reinicio, junto con la suspensión (guarda estado en memoria) o la hibernación (guarda el estado en disco) del sistema:

```
systemctl [--no-wall] [-f] halt
systemctl [--no-wall] [-f] poweroff
systemctl [--no-wall] [-f] reboot
systemctl suspend
systemctl hibernate
systemctl hybrid-sleep
```

Siguen existiendo las instrucciones clásicas poweroff, shutdown, etc. dentro del paquete systemd-sysv pero realizan sus funciones a través de systemd.



timedatectl

Instrucción utilizada para consultar y fijar la hora del sistema.

```
timedatectl
    Local time: dom 2017-12-10 12:14:57 CET
    Universal time: dom 2017-12-10 11:14:57 UTC
    RTC time: dom 2017-12-10 11:14:57
    Time zone: Europe/Madrid (CET, +0100)
    Network time on: yes
    NTP synchronized: yes
    RTC in local TZ: no
```

Incluye ntp integrado a través de `systemd-timesyncd.service`

loginctl

Puesto (*seat*) Hardware asociado a un puesto de trabajo

Sesión Definida por el tiempo entre el ingreso y la salida del usuario

systemd-logind permite definir sistemas multisesión y/o multipuesto.

loginctl es la instrucción para controlar las sesiones, puestos y demás aspectos relacionados:

```
loginctl list-users
loginctl list-seats
loginctl list-sessions
loginctl user-status [USER]
...
```

