

Apuntes SinCara Extras systemd

Consideraciones iniciales

- Se presenta el 30 de abril de 2010, por Lennart Poetterig
- SystemD es un sistema de inicio y administración de servicios.
- Reemplaza los sistemas de inicio tradicionales como SysV init o Upstart (de Canonical / Ubuntu)
- Se enfoca en la eficiencia, la rapidez y la robustez.
- Es ampliamente usado en la comunidad de Linux, aunque ha generado bastante controversia debido a su enfoque radical, a su reemplazo de sistemas de inicio tradicionales y al cambio de paradigma histórico de UNIX.
- Es el estándar de facto para la gestión de servicios en Linux.

Ventajas de SystemD

- Arranque Paralelo: SystemD permite iniciar servicios de manera paralela, lo que reduce significativamente el tiempo de arranque del sistema.
- Dependencias Explícitas: Los servicios en SystemD pueden especificar sus dependencias de manera explícita, lo que facilita la gestión de la secuencia de arranque y evita problemas de dependencia circular.
- Control Dinámico de Servicios: SystemD proporciona herramientas para controlar dinámicamente los servicios en ejecución, como iniciar, detener, reiniciar o consultar el estado de un servicio.
- Integración con cgroups: SystemD integra el control de grupos de procesos (cgroups) para una gestión más eficiente de recursos y aislamiento de procesos.
- Compatibilidad con SysV init: SystemD es compatible con los scripts de inicio de SysV init, lo que facilita la transición para las distribuciones que migran a SystemD.
- https://without-systemd.org/wiki/index_php/Arguments_against_systemd/ - Argumentos contra SystemD

Paquetes RPM

- `rpm -qa | grep systemd # Instalados en Fedora 42`
 - SystemD 256 x86_64
 - `systemd-256`
 - `systemd-container-256`
 - `systemd-libs-256`
 - `systemd-networkd-256`
 - `systemd-pam-256`
 - `systemd-resolved-256`
 - `systemd-udev-256`
 - Otros
 - `rpm-plugin-systemd-inhibit-4.20.0`
 - `python3-systemd-235-11`
 - `libreport-plugin-systemd-journal-2.17.15`
 - noarch
 - `systemd-oomd-defaults-256`
- `yum list systemd # Paquetes no instalados`
 - x86_64
 - `systemd-boot-unsigned`
 - `systemd-bootchart`

- systemd-devel
- systemd-journal-remote
- systemd-standalone-repart
- systemd-standalone-shutdown
- systemd-standalone-sysusers
- systemd-standalone-tmpfiles
- systemd-tests
- uwsgi-logger-systemd.x86_64
- Otros
 - python-systemd-doc-235-11
- noarch
 - systemd-networkd-defaults
 - systemd-rpm-macros
 - systemd-swap
 - systemd-ukify
 - wine-systemd-10.0

Comandos de SystemD

- `relacionados.sh hostnamectl #` Para ver todos los comandos
- `apropos ctl | \grep «ctl ([18])» #` Comandos que terminan en `ctl` instalados, sean de SystemD o no
- Comandos de SystemD
 - `rpm -ql systemd-container systemd-networkd systemd-udev systemd-resolved systemd | grep bin/`
 - `rpm -ql systemd-container systemd-networkd systemd-udev systemd-resolved systemd | grep bin/ | xargs whatis #` Para ver que hace cada uno
- <https://www.linuxfromscratch.org/lfs/view/systemd/chapter08/systemd.html> - Listado de comandos
- Alias de Comandos:
 - `alias sd_servicios='systemctl --no-pager --state=running --type=service'`
 - `alias sd_sockets='systemctl --no-pager --state=running --type=socket'`
 - `alias sd_timers='systemctl --no-pager --state=running --type=timer'`

Componentes principales

- **systemd**: Es el proceso inicial de SystemD y actúa como el proceso principal del sistema. Es responsable de inicializar el sistema y coordinar el arranque de otros servicios y procesos. Es el proceso con el PID número 1.
- **systemd-journald**: Es el servicio de registro del sistema de SystemD. Reemplaza al tradicional `syslog` y proporciona un registro centralizado y estructurado de eventos del sistema y de los servicios.
- **systemd-udev**: Es el administrador de dispositivos de SystemD. Se encarga de la detección y gestión dinámica de dispositivos en el sistema, incluyendo la asignación de nombres de dispositivos y la configuración de `udev`.
- **systemd-networkd**: Es el administrador de red de SystemD. Proporciona una manera de configurar y gestionar la red del sistema, incluyendo la configuración de interfaces de red, enrutamiento y resolución DNS. No suele estar activo, ya que normalmente se utilizan otros servicios:
 - **NetworkManager.service** en Red Hat, Fedora y derivados
 - **networking.service** en Debian, Ubuntu y derivados
- **systemd-resolved**: Es el servicio de resolución de DNS de SystemD. Proporciona resolución de nombres de dominio y reemplaza a las soluciones tradicionales como `resolv.conf`.
- **systemd-logind**: Es el servicio de gestión de sesiones de usuario de SystemD. Se encarga de gestionar las sesiones de usuario, el control de acceso y la suspensión/hibernación del sistema.

- **systemd-userdbd**: Este servicio gestiona bases de datos de usuarios y grupos para la sesión del usuario. Permite almacenar y recuperar información sobre usuarios y grupos dentro del contexto de una sesión de usuario.
- **systemd-timedated**: Es el servicio de gestión de fecha y hora de SystemD. Se encarga de mantener y sincronizar el reloj del sistema con fuentes de tiempo externas.
- **systemd-oomd**: Este servicio, conocido como «Out-Of-Memory Daemon» (Demonio de Falta de Memoria), supervisa la memoria del sistema y gestiona situaciones de falta de memoria. Cuando el sistema se queda sin memoria, systemd-oomd intenta identificar y terminar procesos no esenciales para liberar recursos y evitar el agotamiento total de la memoria.
- **systemd-machined**: Este servicio es responsable de gestionar máquinas virtuales y contenedores en el sistema.
- **systemd-ask-password-wall**: Este servicio se utiliza para mostrar solicitudes de contraseña en la pantalla del sistema. Cuando un servicio necesita autenticación (por ejemplo, al iniciar una unidad cifrada durante el arranque), systemd-ask-password-wall muestra una ventana emergente en el entorno de escritorio para que el usuario introduzca la contraseña.

Información general

- `systemctl --version` # Para ver la versión de SystemD instalada
- El - y el + indican las características habilitadas de SystemD. Cada Distro elige cuales activar o no, pero no se puede cambiar sin reinstalar el sistema.
 - PAM: La integración con PAM (Pluggable Authentication Modules) permite a SystemD interactuar con el sistema de autenticación del sistema operativo. Esto significa que SystemD puede utilizar los mecanismos de autenticación configurados en PAM para servicios que requieren autenticación.
 - AUDIT: La compatibilidad con AUDIT permite a SystemD interactuar con el subsistema de auditoría del kernel Linux (Linux Audit). Esto proporciona capacidades de auditoría avanzadas para registrar eventos del sistema, como accesos a archivos, cambios de configuración, etc.
 - SELINUX: SystemD tiene integración con SELinux (Security-Enhanced Linux), un sistema de seguridad para el control de acceso obligatorio (MAC). Cuando esta característica está habilitada, SystemD puede interactuar de manera adecuada con las políticas de SELinux para aplicar restricciones de seguridad adicionales en el sistema.
 - APPARMOR: SystemD tiene integración con AppArmor, un marco de seguridad para el control de acceso obligatorio (MAC) similar a SELinux. Cuando esta característica está habilitada, SystemD puede interactuar con las políticas de AppArmor para aplicar restricciones de seguridad adicionales en el sistema.
 - IMA (Integrity Measurement Architecture): La compatibilidad con IMA permite a SystemD interactuar con la arquitectura de medición de integridad del kernel Linux. Esto facilita la realización de mediciones de integridad del sistema para detectar cambios no autorizados en los archivos del sistema.
 - SMACK (Simplified Mandatory Access Control Kernel): La compatibilidad con SMACK permite a SystemD interactuar con el marco de control de acceso obligatorio (MAC) Simplified Mandatory Access Control Kernel. Esto proporciona capacidades de control de acceso a nivel de archivo basadas en etiquetas en el sistema.
 - SECCOMP (Secure Computing Mode): La compatibilidad con SECCOMP permite a SystemD utilizar el modo de computación segura del kernel Linux para restringir las llamadas al sistema disponibles para un proceso. Esto ayuda a mejorar la seguridad limitando las capacidades de los procesos.
 - GCRYPT: SystemD puede interactuar con GnuPG Cryptography (GCRYPT), una biblioteca para cifrado y descifrado de datos. Esto puede ser útil para servicios que requieren funciones criptográficas en el sistema.
 - GNUTLS: SystemD tiene integración con GNU Transport Layer Security (GNUTLS), una biblioteca de criptografía y seguridad de red. Esto puede ser útil para servicios que requieren capacidades de cifrado y autenticación segura.
 - OPENSSL: SystemD puede integrarse con OpenSSL, una biblioteca de cifrado y seguridad ampliamente utilizada en sistemas Linux. Esta integración permite que SystemD utilice funciones criptográficas proporcionadas por OpenSSL para tareas como cifrado, descifrado, generación de

claves, etc.

- ACL (Access Control List): La compatibilidad con ACL permite a SystemD interactuar con listas de control de acceso (ACL) en sistemas de archivos. Esto amplía las capacidades de control de acceso a nivel de archivo en el sistema.
- BLKID: SystemD puede interactuar con blkid, una utilidad para identificar dispositivos de bloques (como discos duros, particiones, etc.). Esto puede ser útil para la gestión de dispositivos de almacenamiento en el sistema.
- CURL: SystemD puede integrarse con cURL, una herramienta de línea de comandos y una biblioteca para transferencias de datos con URLs. Esta integración permite que SystemD realice solicitudes HTTP, HTTPS, FTP y otras solicitudes de red utilizando cURL para tareas como descargar archivos, acceder a servicios web, etc.
- ELFUTILS: SystemD tiene integración con ELF Utils, una colección de herramientas para trabajar con archivos ejecutables y objetos ELF (Formato de Archivo Ejecutable y Enlace). Esto puede ser útil para la depuración y análisis de ejecutables en el sistema.
- FIDO2: FIDO2 es un estándar de autenticación en dos factores (2FA) basado en claves públicas que proporciona una forma segura y conveniente de autenticar usuarios en servicios en línea. La compatibilidad con FIDO2 en SystemD significa que puede interactuar con dispositivos de seguridad compatibles con FIDO2 para autenticación de usuario.
- IDN2, IDN (Internationalized Domain Names): Estas características indican el soporte para el manejo de nombres de dominio internacionalizados (IDN) en el sistema. Esto puede ser útil para la interoperabilidad con nombres de dominio que contienen caracteres no ASCII.
- IPTC: IPTC (International Press Telecommunications Council) es un estándar para el intercambio de información entre sistemas de gestión de contenido y aplicaciones relacionadas con la fotografía y el periodismo. La compatibilidad con IPTC en SystemD puede implicar la capacidad de leer, escribir o manipular metadatos IPTC en archivos de medios.
- KMOD: La compatibilidad con KMOD permite a SystemD interactuar con el sistema de carga de módulos del kernel Linux. Esto puede ser útil para la gestión de módulos del kernel y la carga de controladores de dispositivos en el sistema.
- LIBCRYPTSETUP: SystemD tiene integración con libcryptsetup, una biblioteca para configurar y gestionar volúmenes cifrados en el sistema. Esto permite que SystemD maneje servicios y unidades que dependen de volúmenes cifrados de manera adecuada.
- LIBCRYPTSETUP_PLUGINS:
- LIBFDISK: Libfdisk es una biblioteca de utilidades para trabajar con tablas de particiones de disco y manipular particiones en sistemas Linux. La compatibilidad con Libfdisk en SystemD significa que puede utilizar las funciones proporcionadas por esta biblioteca para realizar operaciones relacionadas con el particionado de discos.
- PCRE2 (Perl Compatible Regular Expressions): La compatibilidad con PCRE2 indica que SystemD tiene soporte para expresiones regulares compatibles con Perl. Esto puede ser útil para la búsqueda y manipulación de cadenas de texto en archivos de configuración y otros contextos.
- PWQUALITY: PWQUALITY es un módulo de SystemD que proporciona políticas de calidad de contraseña para el sistema de autenticación. Esto implica que SystemD puede aplicar reglas y restricciones a la creación y el cambio de contraseñas de usuario para garantizar contraseñas seguras y robustas.
- P11KIT: P11Kit es una biblioteca para interactuar con dispositivos de seguridad que admiten el estándar PKCS #11. La compatibilidad con P11Kit en SystemD significa que puede utilizar esta biblioteca para acceder a dispositivos de hardware de seguridad, como tokens USB criptográficos, para tareas de autenticación y cifrado.
- QRENCODE: QREncode es una biblioteca para generar códigos QR, que son códigos de barras bidimensionales que almacenan información en una matriz de puntos. La compatibilidad con QREncode en SystemD podría implicar la capacidad de generar códigos QR para usar en aplicaciones relacionadas con la identificación o la autenticación.
- TPM2 (Trusted Platform Module 2): TPM2 es un estándar para chips de seguridad integrados en placas base que proporcionan funcionalidades como el almacenamiento seguro de claves, la generación de claves criptográficas y la realización de operaciones criptográficas seguras. La compatibilidad con TPM2 en SystemD podría implicar la capacidad de interactuar con chips TPM2

- para tareas de seguridad y autenticación.
- BZIP2, LZ4, XZ, ZLIB, ZSTD: Estas características indican que SystemD tiene soporte para diferentes algoritmos de compresión. Esto puede ser útil para la compresión y descompresión de datos en el sistema, por ejemplo, en archivos de registro.
- BPF_FRAMEWORK (Berkeley Packet Filter Framework): BPF es un marco para programación de filtros de paquetes de red en sistemas Unix. La compatibilidad con BPF en SystemD puede implicar la capacidad de utilizar esta tecnología para filtrar y analizar el tráfico de red en el sistema.
- XKBCOMMON: XKBCOMMON es una biblioteca para manejar la configuración del teclado en sistemas X Window. La compatibilidad con XKBCOMMON en SystemD puede implicar la capacidad de interactuar con esta biblioteca para configurar y gestionar el comportamiento del teclado en sistemas Linux.
- UTMP: SystemD puede interactuar con el archivo utmp (registro de usuarios en el sistema) para registrar la actividad de inicio de sesión y otros eventos relacionados con los usuarios en el sistema.
- SYSVINIT: Esta característica indica que SystemD tiene compatibilidad con los scripts de inicio tradicionales de SysV init. Esto permite que SystemD inicie y controle servicios que aún utilizan el sistema de inicio basado en SysV init.
- default-hierarchy=hybrid: Esta opción indica el tipo de jerarquía de montaje predeterminada utilizada por SystemD para CGroups. En este caso, «hybrid» significa que se utiliza una combinación de la antigua jerarquía de montaje SysV y la nueva jerarquía de montaje de SystemD.
- LIBARCHIVE:
- <https://github.com/systemd/systemd> - Repositorio de SystemD
- `man systemd.index #` Listado de todas las páginas del manual de SystemD

Unidades

- `systemctl -t help #` Para ver los tipos de unidades disponibles
- `man 7 systemd.special #` Para ver las unidades especiales que no se pueden renombrar
- Units o unidades: es como SystemD llama a los diferentes recursos que maneja. Mientras en SysV solo había servicios, en SystemD las unidades se pueden catalogar como:
 - `.service`: son los servicios, equivalentes a los de SysV. Se pueden crear servicios personalizados.
 - `.socket`: se utilizan para configurar y controlar sockets de red o UNIX en el sistema.
 - `.device`: descripción de dispositivos necesarios, como `udev`, `sysfs`, etc. Es decir, los necesarios para montar particiones, acceder a dispositivos, etc.
 - `.mount`: unidad para definir los puntos de montaje del sistema.
 - `.automount`: unidades montadas automáticamente bajo demanda, al acceder al directorio de montaje. Deben tener una unidad `.mount` coincidente para definir los detalles del montaje
 - `.swap`: unidad que describe el espacio SWAP.
 - `.target`: las correspondientes a los niveles de ejecución, para poder sincronizar una serie de sistemas para determinar el estado.
 - `.path`: Las unidades de ruta se utilizan para monitorear rutas de archivos o directorios en el sistema de archivos. Se utilizan para activar servicios o acciones cuando se producen cambios en archivos o directorios específicos, como la generación de eventos cuando se crea un nuevo archivo en un directorio determinado.
 - `.timer`: es similar a las tareas de cron, un temporizador o planificador de tareas.
 - `.snapshot`: una unidad creada automáticamente por `systemctl` con la que poder reconstruir el estado del sistema tras realizar cambios. Pero es temporal, no sobrevive entre sesiones.
 - `.slice`: asociada con nodos Linux Control Group, para asignar o restringir recursos a un proceso dentro del slice.
 - `.scope`: también creada automáticamente por `systemd` a partir de información recibida de las interfaces de bus. Se emplea para la gestión de conjuntos de procesos que se crean externamente.
- <https://es.linux-console.net/?p=5705> - Comprender las unidades y los archivos de unidades de Systemd
- <https://www.digitalocean.com/community/tutorials/how-to-use-systemctl-to-manage-systemd-services-and-units-es> - Manejo de las unidades

Capacidades de las Unidades

- Activación basada en sockets:
 - Los sockets asociados con un servicio se separan mejor del mismo daemon para poder manejarlos por separado.
 - Esto proporciona una serie de ventajas, como retrasar el inicio de un servicio hasta que se acceda por primera vez al socket asociado.
 - Esto también permite que el sistema cree todos los sockets al principio del proceso de inicio, lo que permite iniciar los servicios asociados en paralelo.
- Activación basada en bus (dbus):
 - Las unidades también se pueden activar en la interfaz de bus proporcionada por D-Bus.
 - Se puede iniciar una unidad cuando se publica un bus asociado.
- Activación basada en ruta (path):
 - Una unidad se puede iniciar en función de la actividad o la disponibilidad de ciertas rutas del sistema de archivos.
 - Esto utiliza inotify.
- Activación basada en dispositivos (udev):
 - Las unidades también se pueden iniciar en la primera disponibilidad de hardware asociado aprovechando los eventos de udev.
- Mapeo de dependencias implícitas:
 - La mayor parte del árbol de dependencias entre unidades lo crea SystemD.
 - Se pueden agregar dependencias, pero la mayor parte del trabajo pesado la realiza por debajo SystemD.
- Instancias y plantillas:
 - Los archivos de unidades de plantilla se pueden usar para crear varias instancias de la misma unidad general.
 - Esto permite ligeras variaciones o unidades hermanas que brindan la misma función general.
- Reforzamiento de la seguridad fácil:
 - Las unidades pueden implementar algunas características de seguridad bastante buenas al agregar directivas simples.
 - Por ejemplo, se puede especificar acceso nulo o de solo lectura a una parte del sistema de archivos, limitar las capacidades del kernel y asignar /tmp y acceso a la red privado.
- drop-ins y fragmentos:
 - Las unidades se pueden ampliar fácilmente al proporcionar fragmentos que anularán partes del archivo de unidad del sistema.
 - Esto hace que sea fácil cambiar entre implementaciones de unidades estándar y personalizadas.

From: <https://miguelangel.torresegea.es/wiki/> - miguel angel torres egea

Permanent link: <https://miguelangel.torresegea.es/wiki/info:cursos:pue:ethical-hacker:extras:sincara-systemd?rev=1740651113>

Last update: 27/02/2025 02:11

