



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA

**Sistemas Operativos
(840)**

Material de Exposición

Sistemas de Inicio en Sistemas Unix

Alumno

Nava Escobar Jose Alfredo

Profesor

Gunnar Eyal Wolf Iszaevich



Martes 26 de enero de 2021

Sistemas de inicio en sistemas tipo Unix

¿Qué es un Sistema de Inicio?

La inicialización es un procedimiento crucial que se encuentra en el corazón de cualquier sistema. Un sistema de inicio es el primer proceso que inicia el sistema operativo después de que se inicia el kernel, este proceso se encarga de ejecutar una serie de tareas como cargar los drivers para finalmente dar paso al inicio de la interfaz gráfica de nuestro sistema operativo, en pocas palabras se encarga de conseguir que todo funcione como debe ser al iniciar el funcionamiento.

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

Filosofías de arranque en sistemas tipo Unix

- Tradicionales:
 - SysV
 - BSD
- Nuevos
 - Systemd
 - Upstart

System V

Abreviado como SysV es un antiguo sistema de inicialización de sistema que por compatibilidad gran parte de sus características están incorporadas en otros sistemas de arranque, tiene su origen en una versión Unix de AT&T Bell Labs.

Funcionamiento de SysV

- Una vez que se inicia el kernel, el script de inicio ejecutara el programa `/sbin/init` el cual lee el archivo `/etc/inittab/` que a su vez ejecutará más scripts que complementaran la inicialización del sistema.
- El primer script que ejecuta init es `/etc/rc.d/rc.sysinit`. Este script realiza varias tareas de inicialización.
- Lo siguiente que hace init (nuevamente impulsado por entradas en `/ etc / inittab`) es ejecutar `/etc/rc.d/rc`, pasándole un número conocido como nivel de ejecución.

Cuando SysV se utiliza, el estado de la maquina se define mediante niveles de ejecución, estos niveles son los encargados de arrancar o detener servicios, todo controlado por el comando init, los niveles de ejecución son los siguientes:

0. Shutdown
1. Single User Mode
2. Multiuser mode without networking
3. Multiuser mode with networking
4. Unused
5. Multiuser mode with networking and GUI
6. Reboot
- 7.

El nivel de ejecución inicial se especifica en `/ etc / inittab`.

El administrador del sistema también puede volver a ejecutar el procesamiento en este paso una vez finalizado el proceso de arranque para cambiar el nivel de ejecución del nivel establecido previamente a un nuevo nivel de ejecución. El administrador lo hace ejecutando el comando `/sbin / init X` (donde X es el nivel de ejecución como arriba)

- El primer paso es ejecutar los scripts en el directorio `/rc.d/rcX.d` (donde X es el nivel de ejecución anterior) cuyo nombre comienza con "K". A cada uno de estos scripts se le pasa el parámetro "stop". Estos son en realidad enlaces simbólicos a archivos en `/etc/rc.d/init.d.`) Esto detiene los programas que se estaban ejecutando en el nivel de ejecución anterior.
- A continuación, se ejecutan los scripts del directorio `/rc.d/rcX.d` (donde X es el nuevo nivel de ejecución) cuyo nombre comienza con "S". A cada uno de estos scripts se le pasa el parámetro "start". Estos archivos "S" también son enlaces simbólicos a archivos en `/etc/rc.d/init.d.` Esto inicia los programas que se ejecutarán en el nuevo nivel de ejecución.
- Una vez que se ejecutan los scripts RC, el proceso de inicio lee entradas adicionales en `/etc/inittab` y las usa para iniciar varios mecanismos de inicio de sesión locales.

Conocido como BSD, este sistema operativo nace a partir de aportes realizados por la Universidad De Berkeley (de ahí su nombre) tienes varios sistemas operativos descendientes como SunOS, FreeBSD, NetBSD, OpenBSD y Mac OS X.

La filosofía de BSD es más sencilla ya que consiste en ejecutar el programa `/etc/rc` el cual se encarga de realizar todas las tareas del área de usuario como requisito previo a la operación del sistema, las ideas básicas de BSD son la modularidad fina y la reutilización de código, la primera significa que cada servicio básico obtiene su propio script capaz de iniciar el servicio, detenerlo verificar su estado, etc. La reutilización de código significa que las operaciones básicas que deben realizar la mayoría de los scripts se implementan como funciones.

Upstart

Es el sistema de inicio basado en eventos que surgió como remplazo a SysV y fue programado por Scott James Remnant, este sistema de arranque trabaja de forma asíncrona ya que supervisa las tareas mientras el sistema va iniciado, se encarga del monitoreo y procesamiento de las tareas de inicio cuando el sistema arranca y los detiene cuando este se apaga.

Este sistema tiene compatibilidad con SysV por lo que es capaz de ejecutar scripts de este sin la necesidad de modificaciones, esto es una gran ventaja ya que permite trabajar con un entorno mixto formado por sistemas de arranque “tradicionales” y nuevos.

Systemd

Su nombre proviene de “system daemon” y es un sistema de inicio que surge como sustituto para SysV y BSD, permite hacer más trabajo paralelamente al arranque del sistema y evitar la sobrecarga del Shell. Hoy en día systemd es el sistema de arranque predeterminado para las principales distribuciones de Linux, pero es compatible con los scripts de System V.

Pero systemd no solo administra y realiza la inicialización del sistema, además monitoriza los procesos y los relanza si es que estos fallan, gestiona el log del sistema, y gestiona las dependencias de forma automática.

Systemd administra unidades que son representaciones de los recursos y servicios del sistema, las unidades que administra este sistema de arranque son:

- service
- socket
- device
- mount
- automount
- swap
- target
- path
- timer
- snapshot

- slice
- scope

OpenRC

OpenRC es un sistema de inicio basado en dependencias para sistemas operativos de computadoras similares a Unix. Fue creado por Roy Marples, un desarrollador de NetBSD que también participó activamente en el proyecto Gentoo. Se adoptó más ampliamente como un sistema de inicio fuera de Gentoo luego de la decisión de algunas distribuciones de Linux de no adoptar systemd.

OpenRC proporciona una serie de características promocionadas como innovadoras por sistemas de inicio como systemd o Upstart, como:

- soporte de cgroups ,
- supervisión de procesos,
- puesta en marcha paralela de servicios, y
- Se ejecutan los scripts iniciados por hardware.

Lo hace sin requerir grandes cambios de diseño para adaptarse a diseños y dependencias radicalmente diferentes

Comparaciones:

- SysV - BSD

Si bien se consideran filosofías diferentes SysV y BSD con el tiempo los dos tipos se han mezclado significativamente y los sistemas operativos modernos de tipo Unix tienden a tener características de ambos.

En un inicio y debido a sus orígenes BSD se consideró un “Unix Universitario”, mientras que System V era más “comercial”

- SysV – Systemd & Upstart & OpenRC

El sistema de arranque SysV comenzaba a operar en serie, por ende una tarea solo comenzaba después de que el inicio de la anterior fuera correcto, lo cual provocaba un tiempo de arranque mayor, con systemd y upstart esto cambió ya que fueron diseñado para hacer las cosas de manera simultánea y paralela por lo cual se evitan los retrasos.

- Upstart – Systemd

La principal diferencia es que Upstart está esperando eventos y systemd está coordinando las dependencias. Ambos sistemas pueden ejecutar scripts regulares y ambos intentan iniciarse en paralelo, sin embargo con el paso del tiempo la mayoría de los sistemas tipo Unix han optado por usar systemd debido a que su forma de operar resulta ser más óptima.

Upstart lo que hace es iniciar y detener servicios, otras tareas se delegan a otros subsistemas especializados en ellas, por otro lado systemd hace mucho más que eso pues además administra

contraseñas, inicios de sesión, terminales, energía, procesamiento de registros, puntos de montaje del sistema de archivos, redes y más.

- Systemd = Controversia

Las funciones que involucra systemd han sido objeto de críticas muy duras esto debido a que su filosofía va en contra de lo que representa un sistema Unix, este sistema de arranque ha presentado un crecimiento orientado a hacerse cargo de más y más.

Mientras que la filosofía Unix va orientada a “hacer una cosa y hacerla bien”, Systemd es un sistema mucho más complejo, pero limita la libertad del usuario y añade puntos débiles que por sí mismos pueden colapsar el sistema o son fáciles de atacar externamente.

Journal es un sistema de registro de incidencias que permite solucionar fácilmente fallos del sistema, systemd ha modificado la forma en que trabaja respecto a versiones anteriores, haciéndola más compleja, manejando journal mediante código binario fácilmente corruptible, además de otras problemáticas.

SystemD busca tener un control unificado sobre el ecosistema GNU/Linux con el objetivo de eliminar “diferencias sin sentido entre distribuciones”.

Pero también existe la otra cara de la moneda, existen usuarios de sistemas Unix que prefieren este sistema de arranque por encima de los tradicionales y Upstart, argumentando que las características de systemd ofrecen una ventaja en el arranque y funcionamiento de los sistemas, además de proporcionar herramientas para gran parte de la configuración de estos.

Otro de los argumentos es que las computadoras modernas son bestias complejas, por lo tanto el sistema operativo que se ejecuta en ellas también tendrá que ser complejo.

Fuentes de Consulta

Linux System Process Initialization (SysV)

<http://glennastory.net/boot/init.html>

A practical guide to BSD rc.d scripting

http://beefchunk.com/documentation/sys/bsd/rc.d_scripting/article.html

Upstart

<https://www.ediciones-eni.com/open/mediabook.aspx?idR=a7bd42c053beaef985e4ce2939d9bafc>

Linux System Process Initialization Using Upstart

<http://glennastory.net/boot/upstart.html>

Sistema de arranque Linux

<https://1984.lsi.us.es/wiki-ssoo/images/ssoo/2/21/Sistema-de-arranque-linux.pdf>

Linux System Process Initialization Using Systemd

<http://glennastory.net/boot/systemd.html>

Understanding and administering systemd

<https://docs.fedoraproject.org/en-US/quick-docs/understanding-and-administering-systemd/index.html>

La polémica de SystemD

<https://lignux.com/la-polemica-systemd/>

The biggest Myths

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

OpenRC

<https://wiki.gentoo.org/wiki/OpenRC>

Inicio del sistema

<http://gwolf.sistop.org/laminas/X2-inicio-sistema.pdf>