
Systemd

18 de diciembre de 2016

Resumen

En el mundo de *UNIX* donde el kernel está público y cada vez es más grande la comunidad que desarrolla nuevas tecnologías para mejorar el sistema *UNIX*, en nuestro caso vamos a analizar una tecnología que rápidamente ha sido adaptada por las principales distribuciones de *UNIX*, *systemd*. Se explicará en que consiste esta tecnología y sus ventajas e inconvenientes. Además examinaremos la implementación de *systemd* para saber un poco más sobre como funciona.

1. Introducción

En los sistemas de tipo *UNIX*, concedidos al principio de la decada de los 70, como se indica en la historia de The Open Group [4], seideo de forma gratuita y libre. Fomentando la idea de software libre y de código abierto y creando la comunidad tan grande y amplia que conocemos hoy en día. Gracias a esta comunidad se ha seguido desarrollando y avanzando el sistema *UNIX* hasta el kernel de *Linux* que conocemos y utilizamos a diario. Constantemente se están desarrollando nuevas tecnologías y procesos para mejorar el sistema.

Nosotros nos vamos a centrar en comparar una nueva versión de unos de los procesos principales del sistema con su predecesora que curiosamente se ha mantenido casi intacto desde la década de los noventa, es decir, desde hace más de 20 años no ha sido modificada, echo sorprendente pensando en la velocidad que avanza siempre la tecnología. El proceso en cuestión es el encargado del inicio del sistema, el proceso con PID (Proccess ID) 1 el arranque del sistema funciona con un proceso que va ejecutando todos los demás procesos como los drivers del adaptador de red o del ratón, los controladores de pantalla, etc.

Este proceso era *init.d* que utilizaban todas las distribuciones basadas en Linux. Pero Lennart Poettering, como redacta en su artículo [1] y Kay Sievers decidieron desarrollar uno proceso nuevo de arranque del sistema, desarrollando así *systemd* el encargado de iniciar practicamente todas las distribuciones de Linux actuales.

Aunque ambos sistemas son daemons (demonios) hay multitud de diferencias entre ellos, que se explicarán mas adelante y finalmente una conclusión de ambos de sistemas de arranque para el mundo de los servidores.

2. Init

El proceso Init se caracterizaba por su simpleza y facilidad de uso. El funcionamiento de Init consiste en ir iniciando los procesos listados en un archivo de configuración, es decir, inicia el primer proceso de la listado y cuando este se ha iniciado inicia el siguiente y así sucesivamente. De esta forma tan simple se iniciaba el sistema. Además utiliza la filosofía del software libre de ser transparente para el usuario y permitirle poder modificar por completo su implementación, ya que bastaba con modificar el fichero de configuración. De esta forma el proceso Init siempre es el padre o antecesor de todo proceso y además adopta cualquier proceso que pudiera quedarse sin un padre por el motivo que fuera.

Pero tenia unos claros inconvenientes derivados de su propia naturaleza simple y sencilla, primero las dependencias deben controlarse por parte del programador que tiene que saber que procesos van han de iniciarse primero y cuales después, por ejemplo si en el listado de procesos aparece un proceso apache que depende de la configuración de red este proceso se iniciará con errores o no se iniciará puesto que el controlador de red todavía no se ha iniciado, aun cuando apache este perfectamente configurado y el controlador de red también. El otro problema principal era la sobrecarga al inicio del sistema, ya que se iniciaban los procesos de uno en uno y hasta que uno no acabase de arrancar no se iniciaba el siguiente.

3. Systemd

El proceso systemd al como la definen los autores en la web oficial de systemd [3] es una suite o conjunto de herramientas diseñadas para ofrecer una funcionalidad específica, en este caso para facilitar y mejorar el arranque del sistema operativo *UNIX*. Específicamente es un conjunto de demonios de *UNIX*, un demonio [2] es un proceso que funciona en segundo plano en el sistema a la espera de eventos o llamadas que se producen en el sistema y cuando son despertados realizan una tarea concreta.

4. Historia

4.1. Funcionamiento

4.2. Implementación

5. Conclusiones

6. Conclusiones

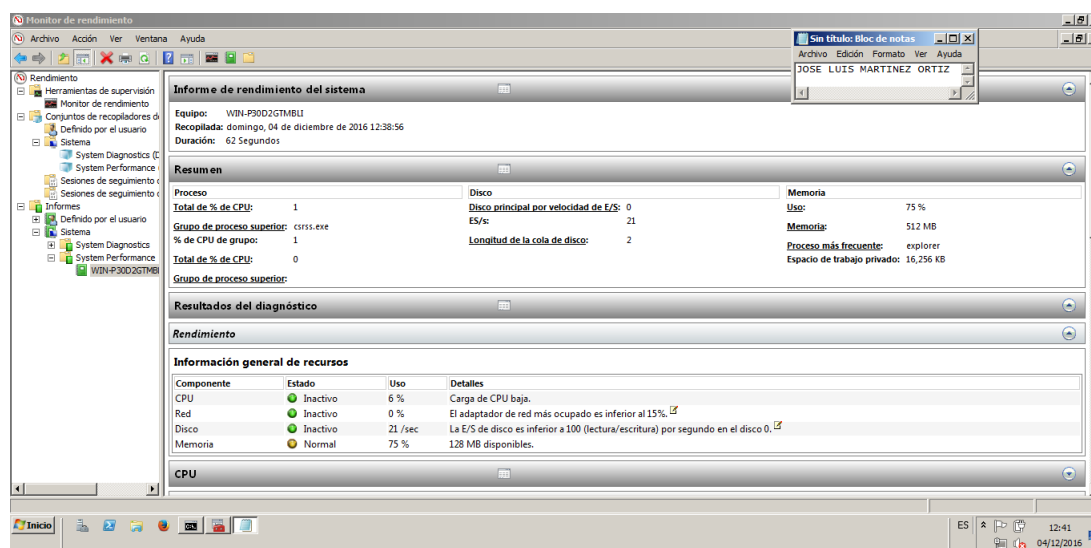


Figura 6.1: Informe del monitor de rendimiento - resumen

Referencias

- [1] <http://0pointer.de/blog/projects/systemd.html#faqs>, consultado el 14 de Diciembre de 2016. Blog de Lennart.
- [2] <https://wiki.archlinux.org/index.php/daemons>, consultado el 14 de Diciembre de 2016. ArchWiki, apartado sobre daemons.
- [3] <https://www.freedesktop.org/wiki/Software/systemd/>, consultado el 14 de Diciembre de 2016. Web oficial de systemd.
- [4] http://www.unix.org/what_is_unix/history_timeline.html, consultado el 14 de Diciembre de 2016. The Open Group, UNIX.