Back in Fedora 15, Fedora introduced a piece of core software called systemd. systemd is a collection of tools for a range of different tasks. Its primary purpose is initializing, managing and tracking system services and daemons in Fedora, both during startup and while the system is running. In previous versions of Fedora, these jobs were handled by SysVinit (System V init, or just init).

To understand the primary purpose of systemd — initializing the Fedora operating system — first we need to understand what an init system actually does.

### Booting and initializing

Your computer’s startup or boot process begins with the BIOS (Basic Input / Output System) software on the motherboard. After completing hardware initialization and checks, the BIOS starts up the bootloader (GRUB in Fedora’s case). The bootloader accesses the Master Boot Record (MBR) on your storage device, usually a hard drive. It uses the data there to locate and start the Linux kernel.
The init system is the first process started on Fedora after the kernel starts. In fact, the init system always gets the Process ID (PID) of 1 on a system. This process is always executed by the Linux kernel after the early stages of bootup are completed by the BIOS and bootloader (GRUB).

**Daemons and other processes**

The init system then must start all the necessary daemons, background processes and services required to make the operating system work. Examples of such processes include starting up a boot screen, various parts of the networking stack, and hardware in your machine.

Often these processes continue to run in the background after they start, and are called daemons. These daemons manage many parts of your system, such as logging information, watching for devices you insert or remove, and managing user login.

To see all the processes and daemons that the systemd init system is managing on your Fedora system, use the command `systemctl` in the terminal to list them.

After initialization, the init system stays running, for as long as your system is running, and it manages the processes already started, and allows others to be started, stopped and restarted while the OS is running.

**Controlling with systemd**

The init system can also be used to start other services and daemons too. For example, let's say you want your machine to start a webserver on boot. You can tell systemd to start the httpd daemon during boot:

```
sudo systemctl enable httpd
```

If you want the webserver to start right away, that's easy as well:

```
sudo systemctl start httpd
```

Once you have started the service, you can check the status of the service with

```
systemctl status httpd
```

These are only simple examples of how to control specific services with systemd. Managing your Fedora system with systemd is not difficult, though. The Fedora
Introducing the systemd series

This is the beginning of a series of articles delving into systemd usage. The series will assume that you've read the systemd overview page. Once you've done that, this series will show you other interesting ways to use systemd, and explain how it works. Get ready for a fun ride as you learn more about managing Fedora!

Share:

Tumblr Facebook Google Reddit Twitter

Ryan Lerch
Ryan is a designer that works on stuff for Fedora. He uses Fedora Workstation as his primary desktop, along with the best tools from the Libre Graphics world, notably, the vector graphics editor, Inkscape.

Eduardo Silva
October 21, 2015 at 12:22
Fedora Project community

5 Comments

ADD YOURS

Eduardo Silva
October 21, 2015 at 12:22

There's a typo in the last two commands, should be httpd not http
Ryan Lerch
October 21, 2015 at 20:22

Thanks Eduardo!
I have fixed it up now
thanks again

Bubba
October 21, 2015 at 22:05

You look like Mick Foley.

Jose Gonzalez
October 23, 2015 at 13:07

I thought BIOS looked at MBR for bootloader then the bootloader(GRUB) looked at its config for the kernel.

mate
October 31, 2015 at 18:46

BIOS looks in to MBR and loads a small program which knows where GRUB is located then this small program loads GRUB and eventually GRUB loads a kernel.

Leave a Reply
Your email address will not be published.

Name