Bootup and Initialization

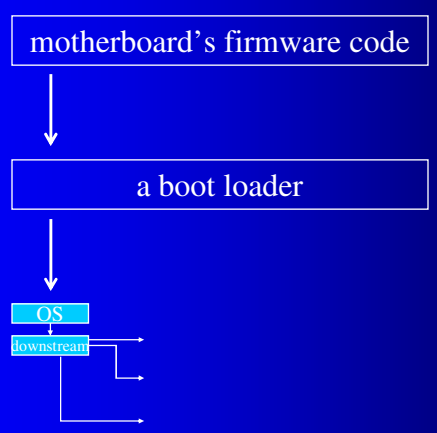
produced and directed by the Linux operating system

all about

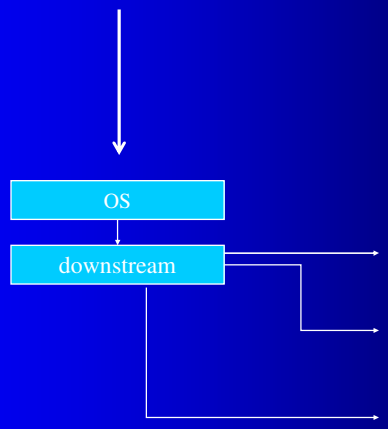
- SysV initializtion method aka SysVinit (venerable, unix)
- systemd (newer, linux ~2010)

David Morgan

General PC Booting sequence

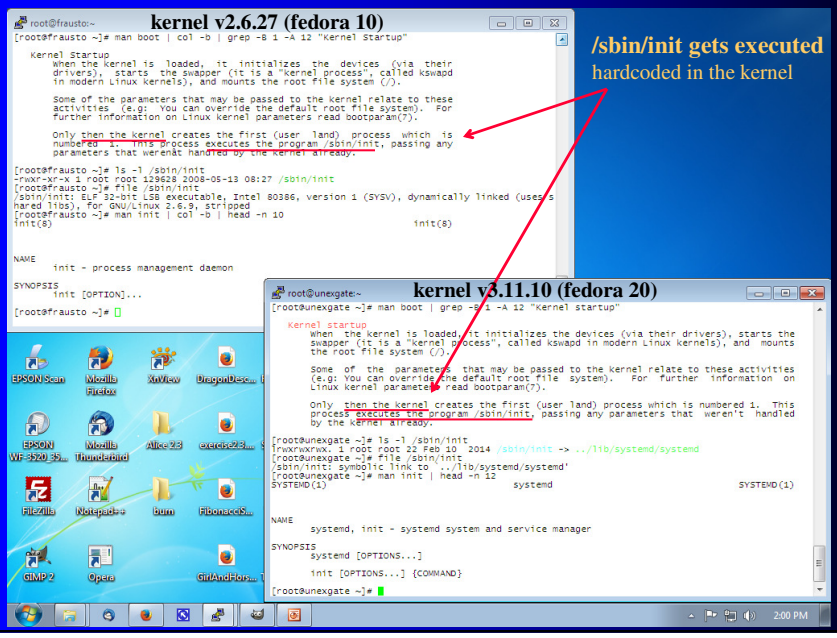


What happens “downstream”?

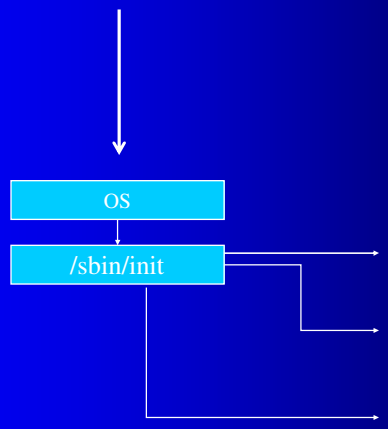


… it depends on the operating system

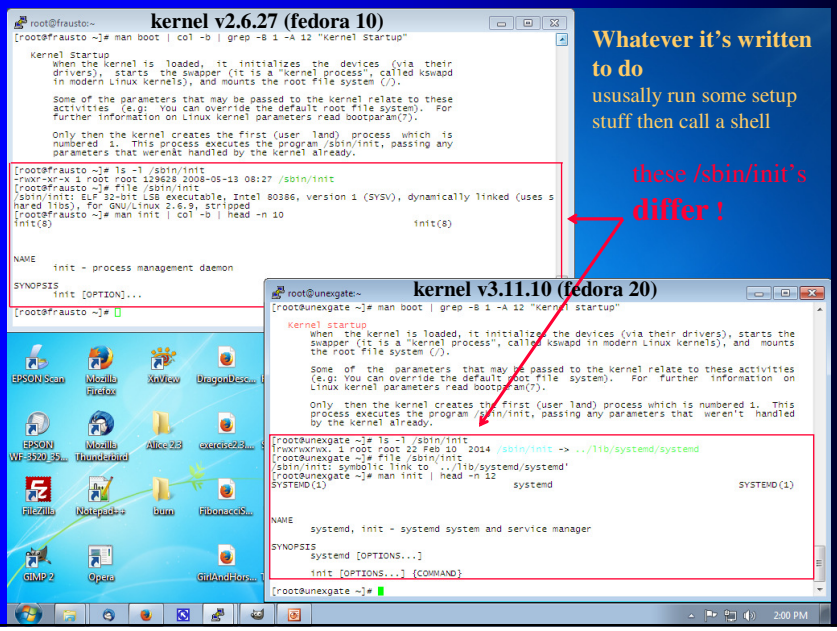
…and what if the OS is linux?



so with linux, “downstream” == /sbin/init



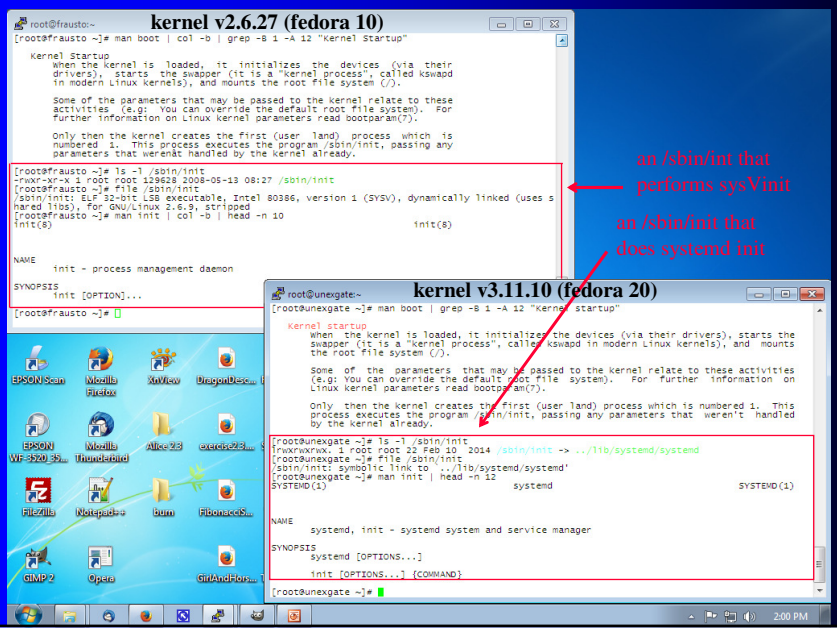
and what does /sbin/init do?



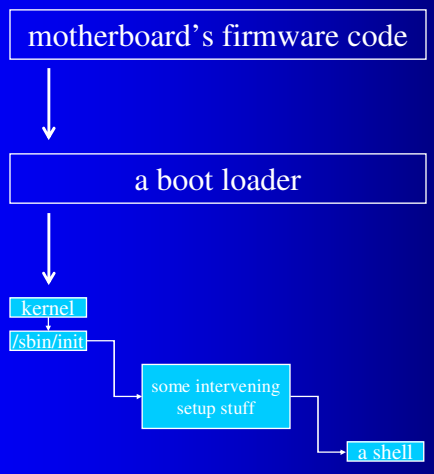
Whatever it’s written to do
ususally run some setup stuff then call a shell

these /sbin/init’s **differ !**

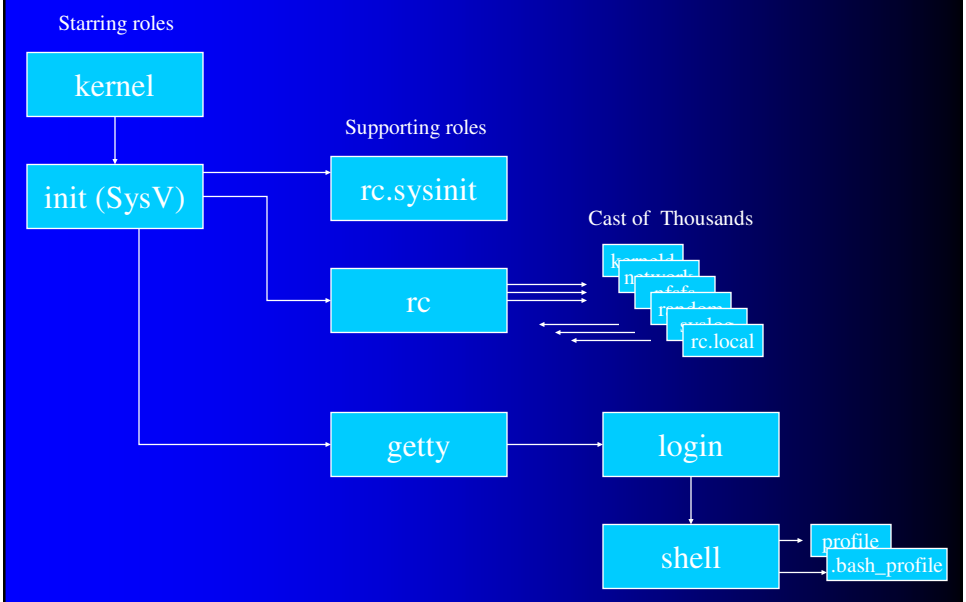
Difference: sysVinit vs systemd init



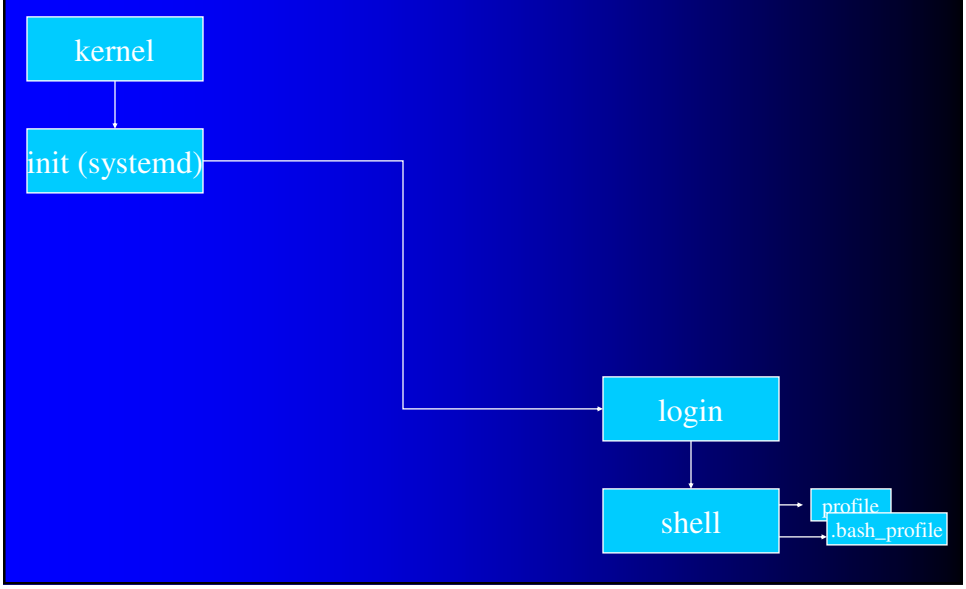
linux booting sequence



A Play in Several Acts - sysvinit



A Play in Several Acts - systemd



Starring role: Kernel

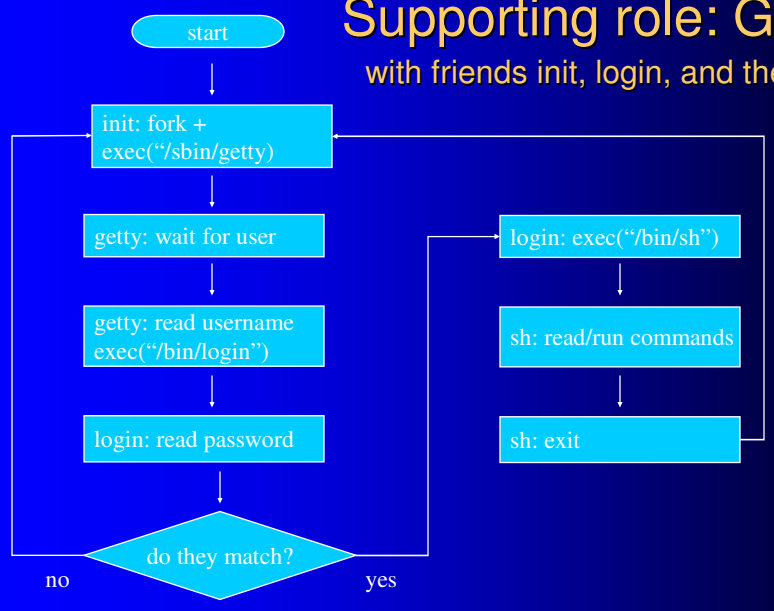
- kernel code loads (from somewhere)
  - from storage device with no filesystem
    - starting from first sector
  - from storage device with filesystem (most common)
    - from a file, e.g., /boot/vmlinuz
  - from a network
    - with PXE boot ("Preboot eXecution Environment)
- identifies/initializes hardware
- displays/stores messages ("dmesg" command shows)
- invokes init process

Starring role: init Process

- father of all processes
  - init is to process structure as root is to file structure
  - always PID number “1”
- creates other processes

Supporting role: Getty

with friends init, login, and the shell



Shell Startup Files

- executed by shell when started by login
- /etc/profile, runs 1st
  - universal settings, all users
- /home/*username*/.bash_profile, runs 2nd
  - settings specific to user "*username*"

Turn services on/off manually

(SysVinit)

- starting
  - /etc/rc.d/init.d/<*script for service*> start *or*
  - **service <*script for service*> start**
- stopping
  - /etc/rc.d/init.d/<*script for service*> stop *or*
  - **service <*script for service*> stop**

- services re-read configuration files when restarted (restart one any time you change its config file)

Set services to boottime auto-on/off

(SysVinit)

- set it to turn on

  chkconfig <*script for service*> on
- set it to not turn on

  chkconfig <*script for service*> off

So much for SysVinit.

Now what about systemd?

systemd

- open source project by Lennart Pottering
- relationship to SysV init system
  - a drop-in replacement (and more)
  - coexists with sysvinit in Fedora 15 (hybrid/transitional)
  - http://fedoraproject.org/wiki/SysVinit_to_Systemd_Cheatsheet
- parallelizes the numerous boot activities
- documentation
  - author's blog: http://0pointer.de/blog
  - see "14 May 2011"entry for links

sysvinit replacement role

Process Identifier 1
On every Unix system there is one process with the special process identifier 1. It is started by the kernel before all other processes and is the parent process for all those other processes that have nobody else to be child of. Due to that it can do a lot of stuff that other processes cannot do. And it is also responsible for some things that other processes are not responsible for, such as bringing up and maintaining userspace during boot.

Historically on Linux the software acting as PID 1 was the venerable sysvinit package, though it had been showing its age for quite a while. Many replacements have been suggested, only one of them really took off: Upstart, which has by now found its way into all major distributions.

As mentioned, the central responsibility of an init system is to bring up userspace. And a good init system does that fast. Unfortunately, the traditional SysV init system was not particularly fast.

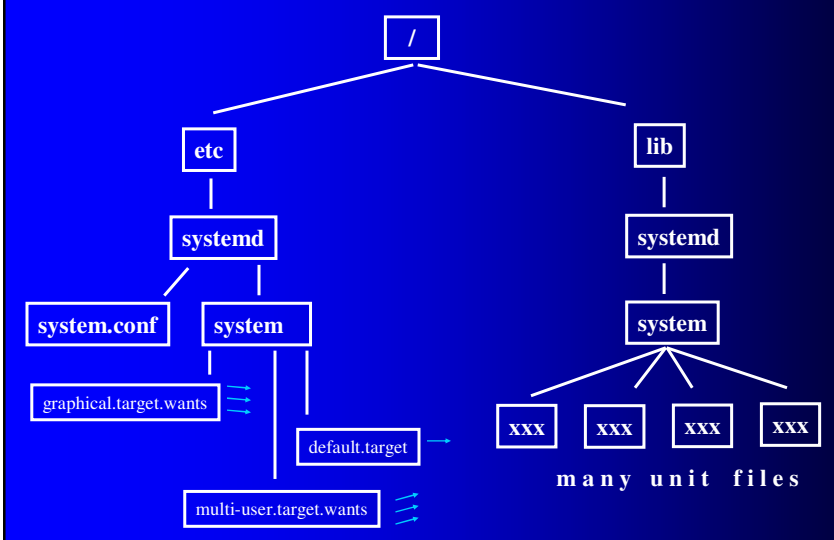
For a fast and efficient boot-up two things are crucial:
To start **less**.
And to start **more** in *parallel*.

What does that mean? Starting less means starting fewer services or deferring the starting of services until they are actually needed. There are some services where we know that they will be required sooner or later (syslog, D-Bus system bus, etc.), but for many others this isn't the case. For example, bluetoothd does not need to be running unless a bluetooth dongle is actually plugged in or an application wants to talk to its D-Bus interfaces. Same for a printing system: unless the machine physically is connected to a printer, or an application wants to print something, there is no need to run a printing daemon such as CUPS. Avahi: if the machine is not connected to a network, there is no need to run Avahi, unless some application wants to use its APIs. And even SSH: as long as nobody wants to contact your machine there is no need to run it, as long as it is then started on the first connection. (And admit it, on most machines where sshd might be listening somebody connects to it only every other month or so.)

Starting more in parallel means that if we have to run something, we should not serialize its start-up (as sysvinit does), but run it all at the same time, so that the available CPU and disk IO bandwidth is maxed out, and hence the overall start-up time minimized.

from Lennart Pottering http://0pointer.de/blog/projects/systemd.html

Default directories and files



a “default.target” unit file (as symlink)
and
a few “wants” subdirs containing symlinks

Default directories and files

```
root@localhost:~
File  Edit  View  Search  Terminal  Help
[root@localhost ~]# ps ax | head -2
  PID TTY      STAT   TIME COMMAND
    1 ?        Ss     0:02 /sbin/init
[root@localhost ~]# ls -l /sbin/init
lrwxrwxrwx. 1 root root 14 Jul 11 17:45 /sbin/init -> ../bin/systemd
[root@localhost ~]# ls -l /etc/systemd/system
total 24
lrwxrwxrwx. 1 root root   40 Jul 11 17:46 dbus-org.freedesktop.Avahi.service -> /lib/systemd/system/avahi-daemon.service
lrwxrwxrwx. 1 root root   42 Jul 11 17:46 dbus-org.freedesktop.NetworkManager.service -> /lib/systemd/system/NetworkManager.service
lrwxrwxrwx. 1 root root   36 Jul 17 13:22 default.target -> /lib/systemd/system/runlevel5.target
drwxr-xr-x. 2 root root 4096 Jul 11 17:42 default.target.wants
drwxr-xr-x. 2 root root 4096 Jul 11 17:42 getty.target.wants
drwxr-xr-x. 2 root root 4096 Jul 11 18:14 graphical.target.wants
drwxr-xr-x. 2 root root 4096 Jul 11 18:14 multi-user.target.wants
drwxr-xr-x. 2 root root 4096 Jul 11 17:46 sockets.target.wants
drwxr-xr-x. 2 root root 4096 Jul 11 17:42 sysinit.target.wants
[root@localhost ~]# ls /lib/systemd/system
abrtd.service                      dev-hugepages.automount      graphical.target                plymouth-halt.service           rsyslog.service          sysinit.target.wants          systemd-readahead-replay.service
accounts-daemon.service            dev-hugepages.mount          graphical.target.wants          plymouth-kexec.service          rtkit-daemon.service     sys-kernel-debug.automount    systemd-remount-api-vfs.service
acpid.service                      dev-mqueue.automount         halt-local.service              plymouth-poweroff.service       runlevel0.target         sys-kernel-debug.mount        systemd-shutdownd.service
atd.service                        dev-mqueue.mount             halt.service                    plymouth-quit.service           runlevel1.target         sys-kernel-security.automount systemd-shutdownd.socket
avahi-daemon.service               display-manager.service      halt.target                     plymouth-quit-wait.service      runlevel1.target.wants   sys-kernel-security.mount     systemd-sysctl.service
avahi-daemon.socket                emergency.service            halt.target.wants               plymouth-read-write.service     runlevel2.target         syslog.socket                 systemd-tmpfiles-clean.service
basic.target                       emergency.target             http-daemon.target              plymouth-reboot.service         runlevel2.target.wants   syslog.target                 systemd-tmpfiles-clean.timer
basic.target.wants                 fedora-autorelabel-mark.service hwclock-load.service         plymouth-start.service          runlevel3.target         syslog.target.wants           systemd-tmpfiles-setup.service
bluetooth.service                  fedora-autorelabel.service   hwclock-save.service            poweroff.service                runlevel3.target.wants   systemd-ask-password-console.path systemd-update-utmp-runlevel.service
bluetooth.target                   fedora-autoswap.service      irqbalance.service              poweroff.target                 runlevel4.target         systemd-ask-password-console.service systemd-update-utmp-shutdown.service
canberra-system-bootup.service     fedora-configure.service     kexec.service                   poweroff.target.wants           runlevel4.target.wants   systemd-ask-password-plymouth.path systemd-user-sessions.service
canberra-system-shutdown-reboot.service fedora-loadmodules.service kexec.target                  prefdm.service                  runlevel5.target         systemd-ask-password-plymouth.service systemd-vconsole-setup.service
canberra-system-shutdown.service   fedora-readonly.service      kexec.target.wants              printer.target                  runlevel5.target.wants   systemd-ask-password-wall.path system-setup-keyboard.service
console-kit-daemon.service         fedora-storage-init-late.service local-fs.target             proc-sys-fs-binfmt_misc.automount runlevel6.target       systemd-ask-password-wall.service time-sync.target
console-kit-log-system-restart.service fedora-storage-init.service local-fs.target.wants         proc-sys-fs-binfmt_misc.mount   serial-getty@.service    systemd-binfmt.service        udev.service
console-kit-log-system-start.service fedora-sysinit-hack.service mail-transfer-agent.target      quotacheck.service              shutdown.target          systemd-hostnamed.service     udev-settle.service
console-kit-log-system-stop.service fedora-sysinit-unhack.service mcelog.service                 quotaon.service                 shutdown.target.wants    systemd-initctl.service       udev.socket
console-shell.service              fedora-wait-storage.service  media.mount                     rc-local.service                sigpwr.target            systemd-initctl.socket        udev-trigger.service
crond.service                      final.target                 messagebus.service              rdisc.service                   single.service           systemd-kmsg-syslogd.service  umount.target
cryptsetup.target                  final.target.wants           multi-user.target               reboot.service                  smartcard.target         systemd-logger.service        var-lock.mount
ctrl-alt-del.target                firstboot-graphical.service  multi-user.target.wants         reboot.target                   smartd.service           systemd-logger.socket         var-run.mount
dbus-org.freedesktop.hostnamed.service firstboot-text.service   NetworkManager.service          reboot.target.wants             sockets.target           systemd-modules-load.service
dbus.service                       fsck-root.service            NetworkManager-wait-online.service remote-fs.target             sockets.target.wants     systemd-random-seed-load.service
dbus.socket                        fsck@.service                network.target                  remount-rootfs.service          sound.target             systemd-random-seed-save.service
dbus.target.wants                  getty@.service               nss-lookup.target               rescue.service                  sssd.service             systemd-readahead-collect.service
default.target                     getty.target                 ntpdate.service                 rescue.target                   swap.target              systemd-readahead-done.service
default.target.wants               gpm.service                  ntpd.service                    rpcbind.target                  sysinit.target           systemd-readahead-done.timer
[root@localhost ~]#
```

PID #1 is /bin/systemd

unit configuration files

Centralized “systemctl” utility

- inspect and control state of systemd
- not to be confused with “sysctl” utility !!

Turn services on/off manually

(set "current" state)

- starting

  systemctl start ***<service's unit file>***
- stopping

  systemctl stop ***<service's unit file>***

- example: turn on/off the logging service

  systemctl start rsyslog.service

  systemctl stop rsyslog.service

Set services to boottime auto-on/off

(set "persistent" state)

- set it to turn on
  - **systemctl enable** ***<service's unit file>***
- set it to not turn on
  - **systemctl disable** ***<service's unit file>***

- example: turn on/off the logging service
  - systemctl enable rsyslog.service
  - systemctl disable rsyslog.service

SysVinit to systemctl cheatsheet

SysVinit to Systemd Cheatsheet - FedoraProject - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://fedoraproject.org/wiki/SysVinit_to_Systemd_Cheatsheet

SysVinit to Systemd Cheatsheet - Fe...

Help

NAVIGATION

- Home
- Get Fedora
- Join Fedora

SUB-PROJECTS

- Ambassadors
- Bug Zappers
- Design
- Documentation
- EPEL
- Infrastructure
- Internationalization
- Localization
- Marketing
- Package Maintainers
- Websites
- Weekly News
- All projects

SEARCH

Go Search

TOOLBOX

Services

sysvinit Command	systemd Command	Notes
service frobozz start	systemctl start frobozz.service	Used to start a service (not reboot persistent)
service frobozz stop	systemctl stop frobozz.service	Used to stop a service (not reboot persistent)
service frobozz restart	systemctl restart frobozz.service	Used to stop and then start a service
service frobozz reload	systemctl reload frobozz.service	When supported, reloads the config file without interrupting pending operations.
service frobozz condrestart	systemctl condrestart frobozz.service	Restarts if the service is already running.
service frobozz status	systemctl status frobozz.service	Tells whether a service is currently running.
ls /etc/rc.d/init.d/	ls /lib/systemd/system/*.service /etc/systemd/system/*.service	Used to list the services that can be started or stopped
chkconfig frobozz on	systemctl enable frobozz.service	Turn the service on, for start at next boot, or other trigger.
chkconfig frobozz off	systemctl disable frobozz.service	Turn the service off for the next reboot, or any other trigger.
chkconfig frobozz	systemctl is-enabled frobozz.service	Used to check whether a service is configured to start or not in the current environment.
chkconfig frobozz --list	ls /etc/systemd/system/*.wants/frobozz.service	Used to list what levels this service is configured on or off
chkconfig frobozz --add		Not needed, no equivalent.

Note that all /sbin/service and /sbin/chkconfig lines listed above continue to work on systemd, and will be translated to native equivalents as necessary. The only exception is chkconfig --list.

Done