06-1 Userspace Initialization - systemd

Chapter 6

Beagle 3.8

bone$ **cat /etc/init.d/README**

```
You are running a systemd-based OS where traditional init scripts have
been replaced by native systemd services files. Service files provide
very similar functionality to init scripts. To make use of service
files simply invoke "systemctl", which will output a list of all
currently running services (and other units). Use "systemctl
list-unit-files" to get a listing of all known unit files, including
stopped, disabled and masked ones. Use "systemctl start
foobar.service" and "systemctl stop foobar.service" to start or stop a
service, respectively. For further details, please refer to
systemctl(1).
```

Beagle 3.8 (cont)

bone$ **cat /etc/init.d/README**

```
Note that traditional init scripts continue to function on a systemd
system. An init script /etc/init.d/foobar is implicitly mapped
into a service unit foobar.service during system initialization.

Thank you!

Further reading:
        man:systemctl(1)
        man:systemd(1)
        http://0pointer.de/blog/projects/systemd-for-admins-3.html

http://www.freedesktop.org/wiki/Software/systemd/Incompatibilities
```

systemd

- init.d is not used on the Bone
- systemd is used for user space initialization
- http://www.freedesktop.org/wiki/Software/systemd/
- Faster boot time by allowing initialization in parallel

Major Linux distributions that adopted systemd

Linux distribution	Date added to software repository[a]	Enabled by default?	Can run without?	Date released as default
Arch Linux	January 2012[52]	Yes	Multiple inits can be installed, but are unsupported[53]	October 2012[54]
CoreOS	July 2013	Yes	?	October 2013 (v94.0.0)[55][56]
Debian	April 2012[57]	Yes	Yes	April 2015 (v8 aka jessie)[58]
Fedora	November 2010 (v14)[59]	Yes	No	May 2011 (v15)
Gentoo Linux[b]	July 2011[62][63][60]	No	Yes	N/A
Mageia	January 2011 (v1.0)[64]	Yes	?	May 2012 (v2.0)[65]
openSUSE	March 2011 (v11.4)[66]	Yes	?	September 2012 (v12.2)[67]
Red Hat Enterprise Linux	June 2014 (v7.0)[68]	Yes	?	June 2014 (v7.0)
Slackware	N/A (not in repository)	N/A	Yes	N/A
SUSE Linux Enterprise Server	October 2014 (v12)	Yes	?	October 2014 (v12)
Ubuntu	April 2013 (v13.04)	Yes	?	April 2015 (v15.04)
Sabayon Linux	August 2013 (v13.08)[61]	Yes	August 2013 (v13.08)	
Ubuntu[e]	April 2013 (v13.04)[63]	Planned[48]	not yet released	

http://en.wikipedia.org/wiki/Systemd

systemd-Outline

- Being an Admin
  - Monitoring boot up
  - cgroup
  - Stopping, starting, etc.
  - Boot time
- Running your own server

Bootup

- Much scrolls by during boot time

```
Starting kernel ...
76
77 Uncompressing Linux... done, booting the kernel.
78 [ 0.000000] Booting Linux on physical CPU 0x0
79 [ 0.000000] Initializing cgroup subsys cpu
80 [ 0.000000] Linux version 3.8.13-bone27 (yoder@ubuntu) (gcc version 4.7.3
20130328
(prerelease) (crosstool-NG linaro-1.13.1-4.7-2013.04-20130415 - Linaro GCC
2013.04) )
#1 SMP Thu Aug 29 19:57:17 EDT 2013
81 [ 0.000000] CPU: ARMv7 Processor [413fc082] revision 2 (ARMv7), cr=10c5387d
82 [ 0.000000] CPU: PIPT / VIPT nonaliasing data cache, VIPT aliasing instruction
cache
83 [ 0.000000] Machine: Generic AM33XX (Flattened Device Tree), model: TI AM335x
BeagleBone
```

- What if you miss something?

systemctl – Seeing what's running

- You can see the status of various processes using systemctl

systemctl

```
beagle $ systemctl
UNIT                          LOAD   ACTIVE SUB       DESCRIPTION
proc-sys...t_misc.automount loaded active waiting   Arbitrary Executable File Formats File System Automount Point
sys-devi...tty-ttyO0.device loaded active plugged   /sys/devices/ocp.2/44e09000.serial/tty/ttyO0
sys-devi...ty-ttyGS0.device loaded active plugged   /sys/devices/ocp.2/47400000.usb/musb-hdrc.0.auto/gadget/tty/ttyGS0
sys-devi...-net-eth0.device loaded active plugged   /sys/devices/ocp.2/4a100000.ethernet/net/eth0
sys-devi...blk0boot0.device loaded active plugged   /sys/devices/ocp.2/mmc.10/mmc_host/mmc1/mmc1:0001/block/mmcblk0/mmcl
sys-devi...blk0boot1.device loaded active plugged   /sys/devices/ocp.2/mmc.10/mmc_host/mmc1/mmc1:0001/block/mmcblk0/mmcl
sys-devi...mmcblk0p1.device loaded active plugged   /sys/devices/ocp.2/mmc.10/mmc_host/mmc1/mmc1:0001/block/mmcblk0/mmcl
sys-devi...mmcblk0p2.device loaded active plugged   /sys/devices/ocp.2/mmc.10/mmc_host/mmc1/mmc1:0001/block/mmcblk0/mmcl
sys-devi...k-mmcblk0.device loaded active plugged   /sys/devices/ocp.2/mmc.10/mmc_host/mmc1/mmc1:0001/block/mmcblk0
sys-devi...tty-ttyS0.device loaded active plugged   /sys/devices/platform/serial8250/tty/ttyS0
sys-devi...tty-ttyS1.device loaded active plugged   /sys/devices/platform/serial8250/tty/ttyS1
sys-devi...tty-ttyS2.device loaded active plugged   /sys/devices/platform/serial8250/tty/ttyS2
sys-devi...tty-ttyS3.device loaded active plugged   /sys/devices/platform/serial8250/tty/ttyS3
sys-module-fuse.device      loaded active plugged   /sys/module/fuse
sys-subs...ices-eth0.device loaded active plugged   /sys/subsystem/net/devices/eth0
-.mount                     loaded active mounted   /
dev-mqueue.mount            loaded active mounted   POSIX Message Queue File System
sys-fs-f...onnections.mount loaded active mounted   FUSE Control File System
sys-kernel-debug.mount      loaded active mounted   Debug File System
tmp.mount                   loaded active mounted   /tmp
systemd-...ord-console.path loaded active waiting   Dispatch Password Requests to Console Directory Watch
```

systemctl

```
beagle $ systemctl
UNIT                          LOAD   ACTIVE SUB       DESCRIPTION
bonescript-autorun.service    loaded active running   Bonescript autorun
bonescript.service            loaded active running   Bonescript server
cloud9.service                loaded active running   Cloud9 IDE
connman.service               loaded active running   Connection service
console-...em-start.service   loaded active exited    Console System Startup Logging
crond.service                 loaded active running   Periodic Command Scheduler
dbus.service                  loaded active running   D-Bus System Message Bus
dropbear....1:42389.service   loaded active running   SSH Per-Connection Server
gateone.service               loaded active running   GateOne daemon
gdm.service                   loaded active running   Gnome Display Manager
getty@tty1.service            loaded active running   Getty on tty1
leds.service                  loaded active exited    Angstrom LED config
mpd.service                   loaded failed failed    Music Player Daemon
ntpdate.service               loaded active exited    Network Time Service (one-shot ntpdate mode)
serial-getty@ttyGS0.service   loaded active running   Serial Getty on ttyGS0
serial-getty@ttyO0.service    loaded active running   Serial Getty on ttyO0
```

Systemctl status

```
bone$ systemctl status mpd.service
mpd.service - Music Player Daemon
               Loaded: loaded (/lib/systemd/system/mpd.service; enabled)
               Active: failed (Result: signal) since Mon 2000-01-03 12:44:01 EST; 13 years 9 months ago
              Process: 125 ExecStart=/usr/bin/mpd --no-daemon (code=killed, signal=ABRT)
               CGroup: name=systemd:/system/mpd.service
```

Systemctl status

bone$ **systemctl status mpd.service**

…

```
Jan 03 12:44:01 yoder-black-bone systemd[1]: mpd.service: main
process exited, code=killed, status=6/ABRT
Jan 03 12:44:01 yoder-black-bone systemd[1]: Unit mpd.service
entered failed state
Jan 03 12:44:10 yoder-black-bone mpd[125]: listen: bind to
'0.0.0.0:6600' failed: Address already in use (continuing
anyway, because binding to '[::]:6600' succeeded)
Jan 03 12:44:10 yoder-black-bone mpd[125]: output: No
"audio_output" defined in config file
Jan 03 12:44:10 yoder-black-bone mpd[125]: output: Attempt to
detect audio output device
Jan 03 12:44:10 yoder-black-bone mpd[125]: output: Attempting
to detect a alsa audio device
Jan 03 12:44:10 yoder-black-bone mpd[125]: ALSA lib
confmisc.c:768:(parse_card) cannot find card '0'
pa_threaded_mainloop_get_api(). Aborting.
```

cgroup - Which Service Owns Which Processes?

- One process can start other processes
- It's hard to tell which process runs what
- *Control groups* (cgroups) are groups of processes
- In systemd every process that is spawned is placed in a control group named after its service
- Makes it easier to track down problems

cgroup

```
bone$ systemd-cgls
└ system
  ├ 1 /lib/systemd/systemd
  ├ bonescript.service
  │ └ 963 /usr/bin/node server.js
  ├ bluetooth.service
  │ └ 933 /usr/sbin/bluetoothd -n
  ├ cloud9.service
  │ ├  918 /usr/bin/node server.js --packed -w /var/lib/cloud9
  │ └ 1009 /usr/bin/nodejs
/opt/cloud9/build/standalonebuild/node_modules/v...
  ├ getty@.service
  │ └ tty1
  │   └ 915 /sbin/agetty tty1 38400
  ├ ifup@.service
  ├ polkitd.service
  │ └ 680 /usr/lib/policykit-1/polkitd --no-debug
```

cgroup

```
bone$ systemd-cgls
…
  ├ serial-getty@.service
  │ ├ ttyGS0
  │ │ └ 1030 /sbin/agetty -s ttyGS0 115200 38400 9600
  │ └ ttyO0
  │   └ 458 /sbin/agetty -s ttyO0 115200 38400 9600
  ├ rsyslog.service
  │ └ 434 /usr/sbin/rsyslogd -n -c5
  ├ upower.service
  │ └ 433 /usr/lib/upower/upowerd
  ├ console-kit-daemon.service
  │ └ 432 /usr/sbin/console-kit-daemon --no-daemon
  ├ systemd-logind.service
  │ └ 431 /lib/systemd/systemd-logind
  ├ wpa_supplicant.service
  │ └ 429 /sbin/wpa_supplicant -u -s -O /var/run/wpa_supplicant
```

cgroup

```
bone$ systemd-cgls
…
 ├ xrdp.service
 │ ├ 675 /usr/sbin/xrdp
 │ └ 691 /usr/sbin/xrdp-sesman
 ├ avahi-daemon.service
 │ ├ 415 avahi-daemon: running [yoder-debian-bone.local]
 │ └ 476 avahi-daemon: chroot helper
 ├ generic-boot-script.service
 │ └ 835 /usr/sbin/udhcpd -S /etc/udhcpd.conf
 ├ apache2.service
 │ ├ 733 /usr/sbin/apache2 -k start
 │ ├ 744 /usr/sbin/apache2 -k start
 │ ├ 750 /usr/sbin/apache2 -k start
 │ └ 751 /usr/sbin/apache2 -k start
 ├ systemd-journald.service
 │ └ 99 /lib/systemd/systemd-journald
 └ udev.service
   ├ 118 /sbin/udevd
   ├ 970 /sbin/udevd
   └ 971 /sbin/udevd
```

Outline

- Being an Admin
  - Monitoring boot up
  - cgroup
  - Stopping, starting, etc.
  - Boot time
- Running your own server

Managing

bone$ **systemctl status systemd-journald.service**

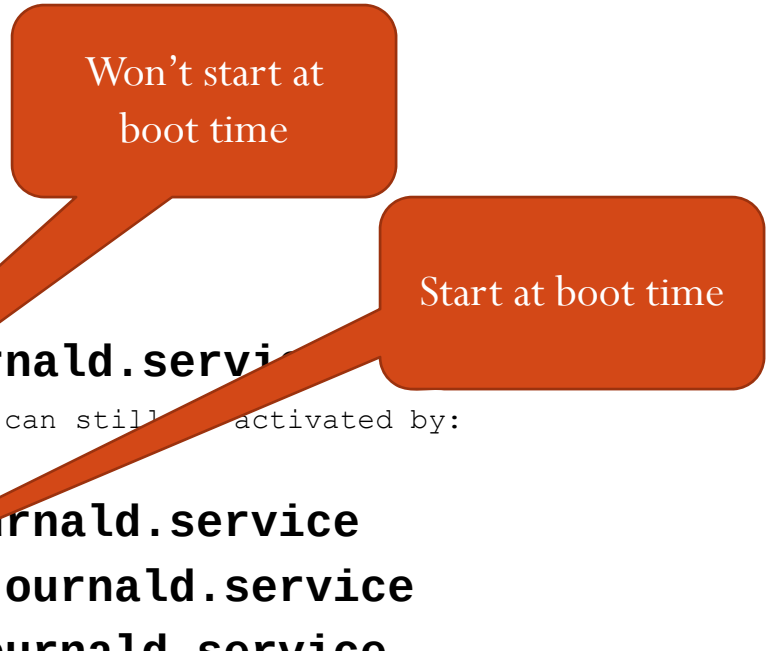
```
systemd-journald.service - Journal Service
           Loaded: loaded (/lib/systemd/system/systemd-journald.service; static)
           Active: mactive (running)since Mon 2000-01-03 12:43:56 EST; 13 years 9
months ago
             Docs: man:systemd-journald.service(8)
                   man:journald.conf(5)
         Main PID: 84 (systemd-journal)
           Status: "Processing requests..."
           CGroup: name=systemd:/system/systemd-journald.service
                     `-84 /lib/systemd/systemd-journald

Jan 03 12:43:56 yoder-black-bone systemd-journal[84]: Allowing runtime journa...
Jan 03 12:43:57 yoder-black-bone systemd-journal[84]: Journal started
Jan 03 12:43:59 yoder-black-bone systemd-journal[84]: Allowing system journal...
Warning: Journal has been rotated since unit was started. Log output is incomplete
or unavailable.
```

Managing

- Stop, start, disable, enable

```
bone$ systemctl stop systemd-journald.servi
Warning: Stopping systemd-journald.servic but it can still  activated by:
  systemd-journald.socket
bone$ systemctl start systemd-journald.service
bone$ systemctl disable systemd-journald.service
bone$ systemctl enable systemd-journald.service
The unit files have no [Install] section. They are not meant to be enabled
using systemctl.
Possible reasons for having this kind of units are:
1) A unit may be statically enabled by being symlinked from another unit's
   .wants/ or .requires/ directory.
2) A unit's purpose may be to act as a helper for some other unit which has
   a requirement dependency on it.
3) A unit may be started when needed via activation (socket, path, timer,
   D-Bus, udev, scripted systemctl call, ...).
```



Outline

- Being an Admin
  - Monitoring boot up
  - cgroup
  - Stopping, starting, etc.
  - Boot time
- Running your own server

Boot performance

```
bone$ systemd-analyze
Startup finished in 1079ms (kernel) + 14107ms (userspace) = 15186ms
bone$ systemd-analyze blame
  9797ms wicd.service
  4742ms apache2.service
  4321ms console-kit-daemon.service
  3525ms xrdp.service
  3479ms bootlogs.service
  3294ms ssh.service
  3037ms cron.service
  2923ms loadcpufreq.service
  2164ms upower.service
  1816ms avahi-daemon.service
  1765ms wpa_supplicant.service
  1736ms systemd-logind.service
  1614ms console-setup.service
  1548ms networking.service
  1348ms lightdm.service
  1297ms polkitd.service
  1262ms capemgr.service
  1250ms generic-boot-script.service
  1009ms rc.local.service 950ms keyboard-setup.service
   923ms udev-trigger.service
   833ms udhcpd.service
   739ms motd.service
   658ms alsa-utils.service
   613ms console-kit-log-system-start.service
   575ms cpufrequtils.service
   562ms udev.service
   510ms kbd.service
   429ms systemd-user-sessions.service
   402ms hostapd.service
   377ms screen-cleanup.service
   330ms saned.service
   327ms systemd-modules-load.service
   249ms systemd-tmpfiles-setup.service
   249ms hdparm.service
   241ms systemd-sysctl.service
   223ms run-lock.mount
```

Outline

- Being an Admin
  - Monitoring boot up
  - cgroup
  - Stopping, starting, etc.
  - Boot time
- Running your own server

Autostarting a server

- For and example, let's use the server in

```
bone$ cd exercises/realtime
bone$ ./boneServer.js
 Listening on 9090
    info  - socket.io started
```

- How do you write your own service script?

Find a working script

```
bone$ systemctl | grep bone
bonescript-autorun.service  loaded active running   Bonescript autorun
bonescript.service          loaded active running   Bonescript server
bonescript.socket           loaded active running   bonescript.socket
```

- I see a couple of bonescript servers that look promising.

```
bone$ systemctl status bonescript
bonescript.service - Bonescript server
   Loaded: loaded (/lib/systemd/system/bonescript.service; static)
   Active: active (running) since Sun 2000-01-09 15:07:55 EST; 13 years 9 months ago
   Main PID: 357 (node)
         CGroup: name=systemd:/system/bonescript.service
            `-357 /usr/bin/node server.js

Jan 09 15:07:55 yoder-black-bone systemd[1]: Starting Bonescript server...
Jan 09 15:08:04 yoder-black-bone bonescript[357]: [35B blob data]
Jan 09 15:08:05 yoder-black-bone bonescript[357]: - - - [Sun, 09 Jan 2000 20:...
```

Copy

```
bone$ cp /lib/systemd/system/bonescript.service boneServer.service
bone$ cat boneServer.service
[Unit]
Description=Bonescript server

[Service]
WorkingDirectory=/usr/lib/node_modules/bonescript
ExecStart=/usr/bin/node server.js
SyslogIdentifier=bonescript

[Install]
WantedBy=multi-user.target
```

Environment Variables

- Node.js also needs

```
bone$ echo $NODE_PATH
/usr/lib/node_modules
```

- You get to figure out how to set it

Install

```
bone$ cp boneServer.service /lib/systemd/system
```

- Start the server

```
bone$ systemctl start boneServer
```

- Point your browser to 192.168.7.2:9090 and see if it works.
- To make it work after rebooting

```
bone$ systemctl enable boneServer
```

```
ln -s '/lib/systemd/system/boneServer.service'
'/etc/systemd/system/multi-user.target.wants/boneServer.service‘
```

- Reboot and see if it worked