systemd as a Container Manager

Seth Jennings
sjenning@redhat.com
Texas Linux Fest 2015
8/21/2015

Agenda

- Very quick overview of systemd
- What is a Linux Container
- systemd as a Container Manager
- Live Demo! Because I like to punish myself!

Disclaimer

What is systemd?

- systemd is a suite of system management daemons, libraries, and utilities designed as a central management and configuration platform for the Linux operating system.

How Big Is This “Suite”

- systemd - init process, pid 1
- journald
- logind
- udevd
- hostnamed
- machined
- importd
- networkd
- resolved
- localed
- timedated
- timesyncd
- and more!

Don't Leave!

- No deep dive on all of these
- Focus on using systemd for container management
  - Spoiler alert: many of the systemd commands you already use work on containers managed by systemd too!

What is a Linux Container

- What it is not
  - Magic
    - conjured only from the mystical language of Go
  - Virtualization (hardware emulation)
  - A completely new concept never before conceived of by man since time began
  - An image format
  - An image distribution mechanism
  - Only usable by modular (microservice) applications at scale

What is a Linux Container

- A resource-constrained, namespaced environment, initialized by a container manager and enforced by the kernel, where processes can run
  - kernel cgroups limits hardware resources
    - cpus, memory, i/o
    - special cgroup filesystem /sys/fs/cgroup
  - kernel namespacing limits resource visibility
    - mount, PID, user, network, UTS, IPC
    - syscalls clone(), setns(), unshare()

What is a Linux Container

- The set of processes in the container is rooted in a process that has pid 1 inside the pid namespace of the container
- The filesystem inside the container can be as complex as a docker image or as simple as a subdirectory on the host (think chroot).

What is a Container Manager

- A userspace program that issues syscalls to the kernel to start the container's pid 1 process in the container environment
  - docker, lxc, **systemd-nspawn**
  - rkt uses systemd-nspawn as default stage1
- The kernel runs the container process and enforces the container constraints, not the container manager
- The container manager is not a hypervisor. It is the parent process of the container's pid 1 process

What Kind of Container?

- Single process or multiple processes
  - microservice vs machine-like
- Opaque or transparent images
  - immutable versioned blob vs subdirectory on host
- Composed images or update in-place
  - package manager inside the container
- Ephemeral or persistent

What Kind of Container?

- Scheduled by cluster manager or locally managed lifecycle
- Host is just a container execution platform (CoreOS, Atomic, Snappy, etc) or a traditional Linux box

systemd as a Container Manager

- systemd has a number of coupled components that make managing containers easy
- machined
  - manage containers
- systemd-nspawn
  - starts pid 1 in container environments
- importd
  - retrieve container images

systemd as a Container Manager

- networkd
  - host and container network configuration
- resolved
  - host and container name resolution

Create a Container Filesystem

```
$ dnf -y \
--releasever=23 \
--installroot=/tmp/f23 \
install fedora-release @standard \
passwd systemd dnf
```

Start a Container

```
$ systemd-nspawn -D /tmp/f23

Spawning container f23 on /tmp/f23.

Press ^] three times within 1s to kill
container.

[root@f23 ~]# ps

  PID TTY          TIME CMD

    1 ?        00:00:00 bash

   18 ?        00:00:00 ps

[root@f23 ~]# passwd <-- set root passwd
```

Boot a Container

```
$ systemd-nspawn -bD /tmp/f23
Spawning container f23 on /tmp/f23.
Press ^] three times within 1s to kill container.
systemd 222 running in system mode.
Detected virtualization systemd-nspawn.

Welcome to Fedora 23 (Twenty Three)!

[  OK  ] Reached target Swap.
[  OK  ] Created slice Root Slice.
[  OK  ] Created slice User and Session Slice.
[  OK  ] Listening on Journal Socket (/dev/log).
[  OK  ] Listening on /dev/initctl Compatibility Named Pipe.
...

Fedora release 23 (Twenty Three)
Kernel 4.2.0-0.rc5.git0.2.fc23.x86_64 on an x86_64 (console)

f23 login:
```

Create an Image

- systemd uses a highly proprietary container image format
- tarballs

Create an Image

```
$ cd /tmp/f23
$ tar cfa ~/f32.tar.xz *
```

Signing and Integrity

- systemd uses a highly proprietary signing and integrity check mechanism for container images
- sha256sum and gpg2

Signing and Integrity

```
$ sha256sum -b f23.tar.xz > SHA256SUMS
$ gpg2 -sb -o SHA256SUMS.gpg SHA256SUMS
```

Host an Image

- systemd pulls images from a highly proprietary image registry
- Any HTTP/FTP server

Host an Image

```
$ python -m SimpleHTTPServer 8080
```

Download/Import an Image

```
$ machinectl import-tar f23.tar.xz
```

or

```
$ machinectl pull-tar \
http://localhost:8080/f23.tar.xz
```

```
$ machinectl list-images
NAME TYPE      RO  USAGE  CREATED                          MODIFIED
f23  subvolume no  494.7M Mon 2015-08-10 14:26:45 CDT n/a

1 images listed.
```

```
$ machinectl read-only f23 true
```

Image Storage

- systemd uses a highly proprietary copy-on-write (COW) mechanism to avoid on-disk duplication among containers
- BTRFS

Image Storage

```
$ machinectl clone f23 test
$ cd /var/lib/machines
$ btrfs subvolume show f23
/var/lib/machines/f23
Name:       f23
...
Flags:        readonly
Snapshot(s):
test
```

Start a Container

```
$ machinectl start test
$ machinectl
MACHINE CLASS     SERVICE
test    container nspawn

1 machines listed.
```

Log Into a Container

```
$ machinectl login test

Connected to machine test. Press ^] three times within 1s
to exit session.


Fedora release 23 (Twenty Three)

Kernel 4.2.0-0.rc5.git0.2.fc23.x86_64 on an x86_64 (pts/0)


test login: root

Password:

Last login: Tue Aug 11 10:14:56 on pts/0

[root@test ~]$
```

Start Container on Boot

```
$ systemctl enable machines.target
$ machinectl enable test
```

Container Status from Host

```
$ machinectl status test
test
           Since: Tue 2015-08-11 15:41:41 CDT; 4s ago
          Leader: 1380 (systemd)
         Service: nspawn; class container
            Root: /var/lib/machines/test
           Iface: ve-test
         Address: 10.0.0.2
                  fe80::90f6:3fff:fee0:a7c1%4
              OS: Fedora 23 (Cloud Edition)
            Unit: systemd-nspawn@test.service
                  ├─1377 /usr/bin/systemd-nspawn --quiet --keep-unit --boot
--link-journal=try-guest --network-veth --machine=test
                  ├─1380 /usr/lib/systemd/systemd
                  └─system.slice
                    ├─dbus.service
                    │ └─1454 /usr/bin/dbus-daemon --system --address=systemd:
--nofork --nopidfile --systemd-activation
```

Container Service Status from Host

```
$ systemctl -M test status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service;
enabled; vendor preset: enabled)
   Active: active (running) since Tue 2015-08-11 13:57:28
CDT; 1h 23min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
 Main PID: 66
   CGroup: /machine.slice/systemd-
nspawn@test.service/system.slice/sshd.service
           └─2991 /usr/sbin/sshd -D
```

Container Journal from Host

```
$ journalctl -M test -b -o cat -n 5

test systemd[1]: Started dnf makecache.

test systemd[1]: Startup finished in 293ms.

test systemd-networkd[31]: host0: Configured

test systemd[1]: Starting Cleanup of Temporary
Directories...

systemd[1]: Started Cleanup of Temporary Directories.
```

(timestamps removed for formatting)