lieter_

@lieter@mastodon.lieter.nl

pieterlexis

Running containers and OS images with `systemd-nspawn`

Pieter Lexis

November 21, 2019

whoami

Pieter Lexis

- SysAdmin by training, developer by accident
- Senior PowerDNS Engineer at PowerDNS
- Packaging and autotools wizard
- Build and test automation fanatic

POWERDNS

systemd-nspawn

what is systemd-nspawn?

> **DESCRIPTION**
>
> systemd-nspawn may be used to run a command or OS in a light-weight namespace container. In many ways it is similar to chroot(1), but more powerful since it fully virtualizes the file system hierarchy, as well as the process tree, the various IPC subsystems and the host and domain name.[1]

- “chroot on steroids”
- Included in and integrated with systemd
- Everything is documented with manpages
- High degree of container configuration

[1] systemd-nspawn(8)

vs Docker/k8s

With `systemd-nspawn`...

- Image creation etc. considered out of scope
- One image is one container
- Containers are considered persistent
- No multi-node orchestration exists
- Processes are directly managed by systemd
- No separate service (like `dockerd`) required

vs `chroot`

`systemd-nspawn` does ...

- Fully virtualize the
    - filesystem hierarchy (using mount namespaces)
    - process tree (first process = PID 1)
    - network isolation
- Resource limiting with cgroups
- Sending the journal to the host

vs LXC

Compared to `systemd-nspawn`…

- LXC is more low-level
- Not integrated into the service manager

vs libvirt/KVM and friends

- No kernel is booted in `systemd-nspawn`
- VM introspection not possible from host using KVM
- `systemd-nspawn` containers must be relatively modern

Using `systemd-nspawn`

systemd-nspawn commandline usage

NAME

systemd-nspawn - Spawn a command or OS in a light-weight container

SYNOPSIS[2]

```
systemd-nspawn [OPTIONS...] [COMMAND [ARGS...]]
systemd-nspawn --boot [OPTIONS...] [ARGS...]
```

- Works with images (`-i`), directories (`-D`) or machines (`-M`)
- First invocation is "like chroot"
- Second used in `systemd-nspawn@.service`

[2] systemd-nspawn(8)

systemd-nspawn@.service

- Service to control containers from /var/lib/machines
- Uses --boot
- Private networking by default
- User namespaces by default
- Can be enabled/started like any other service

Configuring containers

“drop-ins” used for configuration overrides[3].

- `/run/systemd/nspawn/$machine.nspawn`
- `/etc/systemd/nspawn/$machine.nspawn`
- `/var/lib/machines/$machine.nspawn` (Unable to elevate privileges)

[3] systemd.nspawn(5)

systemd-nspawn execution options ([Exec])

- Boot/--boot – Find and start init in the container
- Ephemeral/--ephemeral – Run with a snapshot of the filesystem, removed after termination
- ProcessTwo/--as-pid2 – Only with Boot=no, run program as PID 2, inserting a stub init
- Parameters
    - With Boot=no: Run program with parameters as main program
    - With Boot=yes: Pass these parameters to init
- User/--user – Run main program as specified user
- Environment/--setenv – Set environment variables in the container

Filesystem options (`[Files]`)

- `Volatile`/`--volatile` – Mount root as tmpfs, `/usr` as read-only, all state is lost on restart
- `ReadOnly`/`--readonly` – Run with the root read-only
- `Bind`,`BindReadOnly`/`--bind`,`--bind-ro` – Add bind mounts from host into container
- `Overlay`,`OverlayReadOnly`/`--overlay=`,`--overlay-ro` – Overlay of multiple directories from the host to a directory in the container

Networking options (`[Network]`)

- `Private`/`--private-network` – Disconnect from host network namespace
- `Port`/`-p, --port` – Map a port from the host to the container
- `VirtualEthernet`/`--network-veth` – Create a veth interface*
- `Zone`/`--network-zone` – As bridge, but creates bridge with DHCP and NAT*
- `Interface`/`--network-interface` – Assign interface from host to container*
- Other options like `Bridge`, `MACVLAN`, and `IPVLAN`

* = Implies `Private`

Resource and Security options (`[Exec]`)

Too many to discuss!

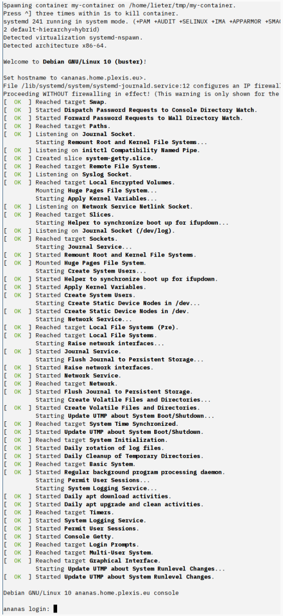
- `NoNewPrivileges`
- `PrivateUsers`, `PrivateUsersChown`
- `SystemCallFilter`
- `LimitCPU`, `LimitNPROC`, `LimitRSS`, `LimitNOFILE`
- `CPUAffinity`
- `Capability`, `DropCapability`

.nspawn files

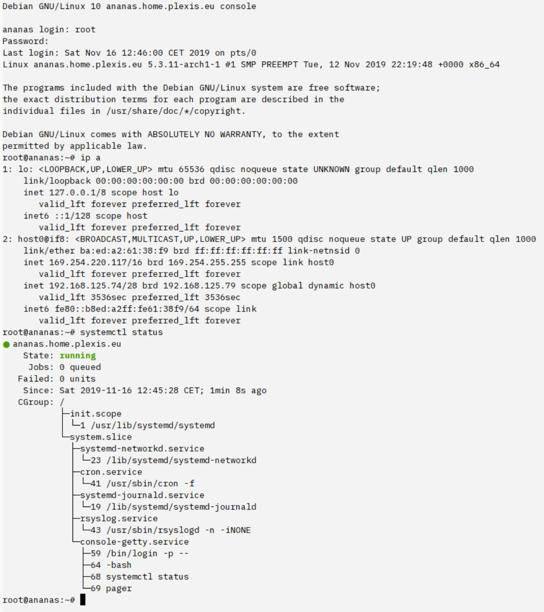
```
[Exec]
Parameters=/usr/bin/foo --foreground
User=foo
ProcessTwo=true
Ephemeral=true

[Files]
Overlay=/srv/dir1:/srv/dir2:/srv/foo

[Network]
Interface=enp0s31f6
```

Creating and booting a single container

```
debootstrap buster /var/lib/machines/my-container

systemd-nspawn -M my-container --as-pid2 \
    systemctl enable systemd-networkd

systemd-nspawn -M my-container --as-pid2 \
    passwd root

systemd-nspawn -M my-container --as-pid2 \
    rm /etc/securetty

systemd-nspawn -M my-container -b -n
```



```
Spawning container my-container on /home/lieter/tmp/my-container.
Press ^] three times within 1s to kill container.
systemd 241 running in system mode. (+PAM +AUDIT +SELINUX +IMA +APPARMOR +SMAC
2 default-hierarchy=hybrid)
Detected virtualization systemd-nspawn.
Detected architecture x86-64.

Welcome to Debian GNU/Linux 10 (buster)!

Set hostname to <ananas.home.plexis.eu>.
File /lib/systemd/system/systemd-journald.service:12 configures an IP firewal
Proceeding WITHOUT firewalling in effect! (This warning is only shown for the
[  OK  ] Reached target Swap.
[  OK  ] Started Dispatch Password Requests to Console Directory Watch.
[  OK  ] Started Forward Password Requests to Wall Directory Watch.
[  OK  ] Reached target Paths.
[  OK  ] Listening on Journal Socket.
         Starting Remount Root and Kernel File Systems...
[  OK  ] Listening on initctl Compatibility Named Pipe.
[  OK  ] Created slice system-getty.slice.
[  OK  ] Reached target Remote File Systems.
[  OK  ] Listening on Syslog Socket.
[  OK  ] Reached target Local Encrypted Volumes.
         Mounting Huge Pages File System...
         Starting Apply Kernel Variables...
[  OK  ] Listening on Network Service Netlink Socket.
[  OK  ] Reached target Slices.
         Starting Helper to synchronize boot up for ifupdown...
[  OK  ] Listening on Journal Socket (/dev/log).
[  OK  ] Reached target Sockets.
         Starting Journal Service...
[  OK  ] Started Remount Root and Kernel File Systems.
[  OK  ] Mounted Huge Pages File System.
         Starting Create System Users...
[  OK  ] Started Helper to synchronize boot up for ifupdown.
[  OK  ] Started Apply Kernel Variables.
[  OK  ] Started Create System Users.
         Starting Create Static Device Nodes in /dev...
[  OK  ] Started Create Static Device Nodes in /dev.
         Starting Network Service...
[  OK  ] Reached target Local File Systems (Pre).
[  OK  ] Reached target Local File Systems.
         Starting Raise network interfaces...
[  OK  ] Started Journal Service.
         Starting Flush Journal to Persistent Storage...
[  OK  ] Started Raise network interfaces.
[  OK  ] Started Network Service.
[  OK  ] Reached target Network.
[  OK  ] Started Flush Journal to Persistent Storage.
         Starting Create Volatile Files and Directories...
[  OK  ] Started Create Volatile Files and Directories.
         Starting Update UTMP about System Boot/Shutdown...
[  OK  ] Reached target System Time Synchronized.
[  OK  ] Started Update UTMP about System Boot/Shutdown.
[  OK  ] Reached target System Initialization.
[  OK  ] Started Daily rotation of log files.
[  OK  ] Started Daily Cleanup of Temporary Directories.
[  OK  ] Reached target Basic System.
[  OK  ] Started Regular background program processing daemon.
         Starting Permit User Sessions...
         Starting System Logging Service...
[  OK  ] Started Daily apt download activities.
[  OK  ] Started Daily apt upgrade and clean activities.
[  OK  ] Reached target Timers.
[  OK  ] Started System Logging Service.
[  OK  ] Started Permit User Sessions.
[  OK  ] Started Console Getty.
[  OK  ] Reached target Login Prompts.
[  OK  ] Reached target Multi-User System.
[  OK  ] Reached target Graphical Interface.
         Starting Update UTMP about System Runlevel Changes...
[  OK  ] Started Update UTMP about System Runlevel Changes.

Debian GNU/Linux 10 ananas.home.plexis.eu console

ananas login: 
```



```
Debian GNU/Linux 10 ananas.home.plexis.eu console

ananas login: root
Password:
Last login: Sat Nov 16 12:46:00 CET 2019 on pts/0
Linux ananas.home.plexis.eu 5.3.11-arch1-1 #1 SMP PREEMPT Tue, 12 Nov 2019 22:19:48 +0000 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@ananas:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: host0@if8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether ba:ed:a2:61:38:f9 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 169.254.220.117/16 brd 169.254.255.255 scope link host0
       valid_lft forever preferred_lft forever
    inet 192.168.125.74/28 brd 192.168.125.79 scope global dynamic host0
       valid_lft 3536sec preferred_lft 3536sec
    inet6 fe80::b8ed:a2ff:fe61:38f9/64 scope link
       valid_lft forever preferred_lft forever
root@ananas:~# systemctl status
● ananas.home.plexis.eu
    State: running
     Jobs: 0 queued
   Failed: 0 units
    Since: Sat 2019-11-16 12:45:28 CET; 1min 8s ago
   CGroup: /
           ├─init.scope
           │ └─1 /usr/lib/systemd/systemd
           └─system.slice
             ├─systemd-networkd.service
             │ └─23 /lib/systemd/systemd-networkd
             ├─cron.service
             │ └─41 /usr/sbin/cron -f
             ├─systemd-journald.service
             │ └─19 /lib/systemd/systemd-journald
             ├─rsyslog.service
             │ └─43 /usr/sbin/rsyslogd -n -iNONE
             └─console-getty.service
               ├─59 /bin/login -p --
               ├─64 -bash
               ├─68 systemctl status
               └─69 pager
root@ananas:~# █
```

machinectl

What is machinectl?

DESCRIPTION

machinectl may be used to introspect and control the state of the systemd(1) virtual machine and container registration manager systemd-machined.service(8). machinectl may be used to execute operations on machines and images.[4]

[4] machinectl(1)

Machine registration service – `systemd-machined`[6]

- Automatically started when `machine.slice` is started
- Keeps track of running containers and their processes
- Optionally allows local resolving of machine-names[5]
- Machines can be controlled with `machinectl`

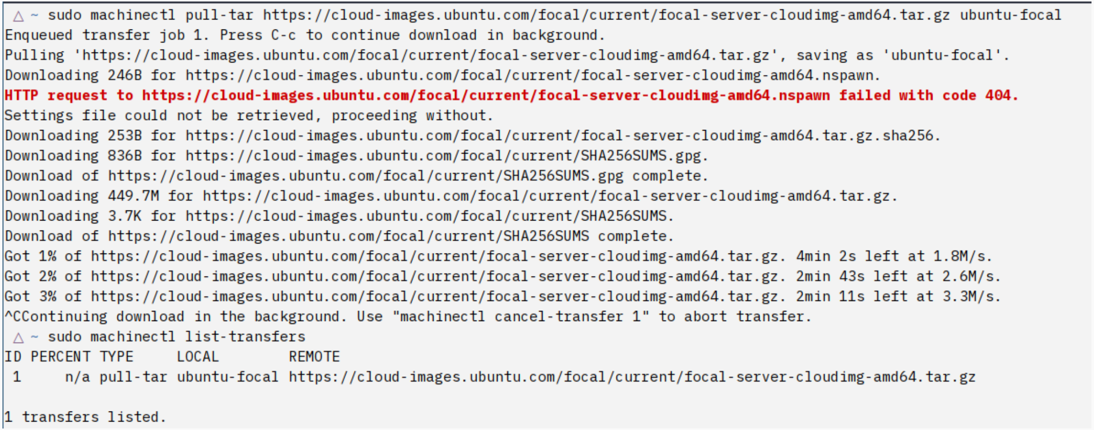
[5] nss-mymachines(8)

[6] systemd-machined(8)

`machinectl` – Image management i

- Download images/directories
- Clone
- Rename
- Import/Export images/directories

machinectl – Image management ii

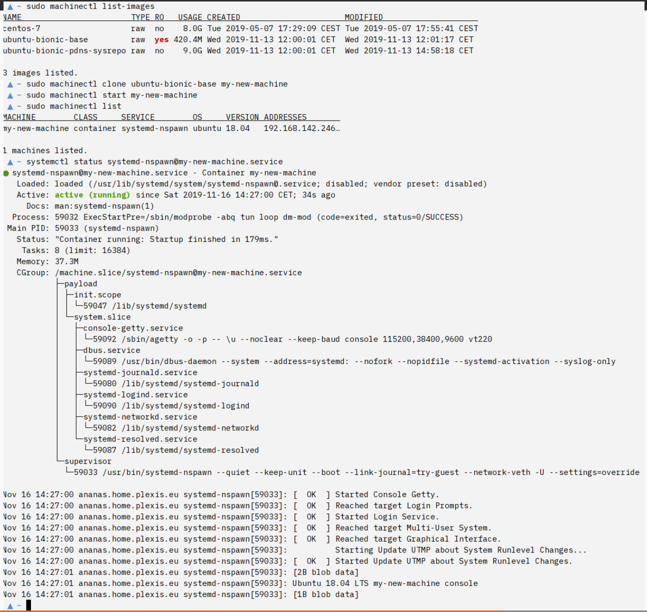


```
 △ ~ sudo machinectl pull-tar https://cloud-images.ubuntu.com/focal/current/focal-server-cloudimg-amd64.tar.gz ubuntu-focal
Enqueued transfer job 1. Press C-c to continue download in background.
Pulling 'https://cloud-images.ubuntu.com/focal/current/focal-server-cloudimg-amd64.tar.gz', saving as 'ubuntu-focal'.
Downloading 246B for https://cloud-images.ubuntu.com/focal/current/focal-server-cloudimg-amd64.nspawn.
HTTP request to https://cloud-images.ubuntu.com/focal/current/focal-server-cloudimg-amd64.nspawn failed with code 404.
Settings file could not be retrieved, proceeding without.
Downloading 253B for https://cloud-images.ubuntu.com/focal/current/focal-server-cloudimg-amd64.tar.gz.sha256.
Downloading 836B for https://cloud-images.ubuntu.com/focal/current/SHA256SUMS.gpg.
Download of https://cloud-images.ubuntu.com/focal/current/SHA256SUMS.gpg complete.
Downloading 449.7M for https://cloud-images.ubuntu.com/focal/current/focal-server-cloudimg-amd64.tar.gz.
Downloading 3.7K for https://cloud-images.ubuntu.com/focal/current/SHA256SUMS.
Download of https://cloud-images.ubuntu.com/focal/current/SHA256SUMS complete.
Got 1% of https://cloud-images.ubuntu.com/focal/current/focal-server-cloudimg-amd64.tar.gz. 4min 2s left at 1.8M/s.
Got 2% of https://cloud-images.ubuntu.com/focal/current/focal-server-cloudimg-amd64.tar.gz. 2min 43s left at 2.6M/s.
Got 3% of https://cloud-images.ubuntu.com/focal/current/focal-server-cloudimg-amd64.tar.gz. 2min 11s left at 3.3M/s.
^CContinuing download in the background. Use "machinectl cancel-transfer 1" to abort transfer.
 △ ~ sudo machinectl list-transfers
ID PERCENT TYPE     LOCAL        REMOTE
 1     n/a pull-tar ubuntu-focal https://cloud-images.ubuntu.com/focal/current/focal-server-cloudimg-amd64.tar.gz

1 transfers listed.
```

`machinectl` – Container management

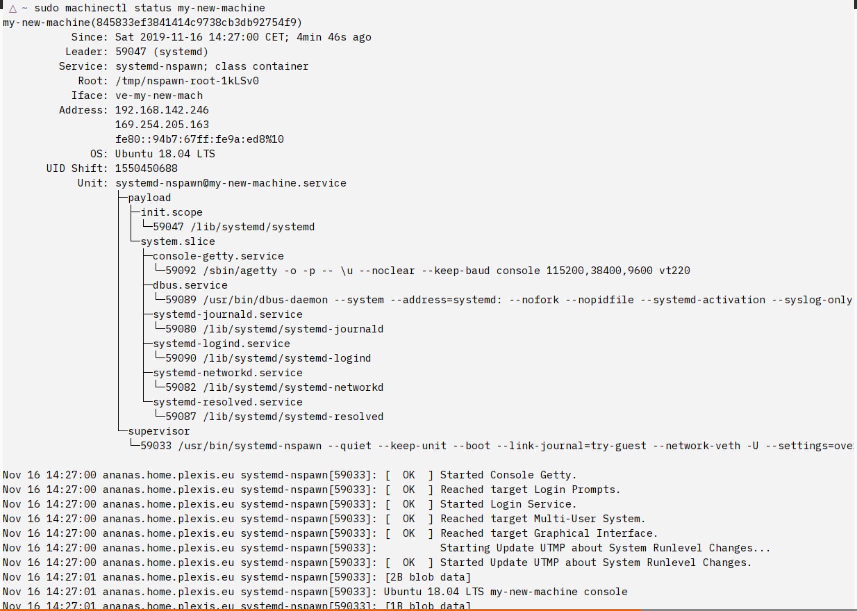
- Start/Stop
- Enable/Disable
- Login/Shell
- Reboot/Poweroff/Terminate
- Copy files from/to the container
- Bind mount a dir in the container

machinectl – Cloning and starting



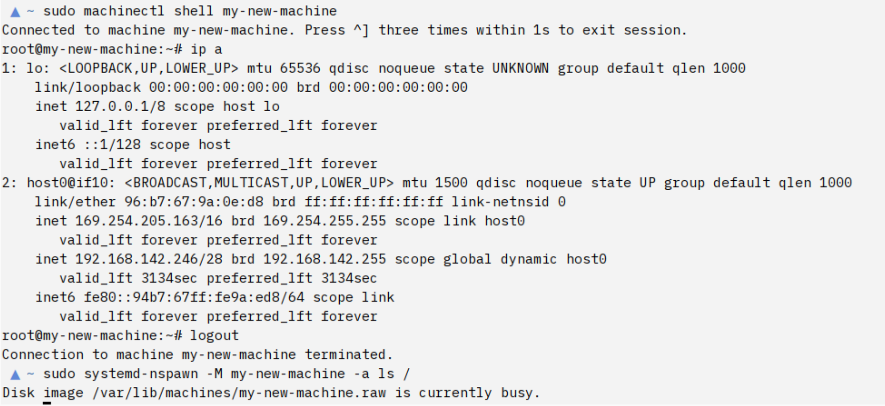
```
▲ ~ sudo machinectl list-images
NAME                      TYPE RO   USAGE CREATED                      MODIFIED
centos-7                  raw  no    8.0G Tue 2019-05-07 17:29:09 CEST Tue 2019-05-07 17:55:41 CEST
ubuntu-bionic-base        raw  yes 420.4M Wed 2019-11-13 12:00:01 CET  Wed 2019-11-13 12:01:17 CET
ubuntu-bionic-pdns-sysrepo raw no    9.0G Wed 2019-11-13 12:00:01 CET  Wed 2019-11-13 14:58:18 CET

3 images listed.
▲ ~ sudo machinectl clone ubuntu-bionic-base my-new-machine
▲ ~ sudo machinectl start my-new-machine
▲ ~ sudo machinectl list
MACHINE        CLASS     SERVICE        OS     VERSION ADDRESSES
my-new-machine container systemd-nspawn ubuntu 18.04   192.168.142.246…

1 machines listed.
▲ ~ systemctl status systemd-nspawn@my-new-machine.service
● systemd-nspawn@my-new-machine.service - Container my-new-machine
   Loaded: loaded (/usr/lib/systemd/system/systemd-nspawn@.service; disabled; vendor preset: disabled)
   Active: active (running) since Sat 2019-11-16 14:27:00 CET; 34s ago
     Docs: man:systemd-nspawn(1)
  Process: 59032 ExecStartPre=/sbin/modprobe -abq tun loop dm-mod (code=exited, status=0/SUCCESS)
 Main PID: 59033 (systemd-nspawn)
   Status: "Container running: Startup finished in 179ms."
    Tasks: 8 (limit: 16384)
   Memory: 37.3M
   CGroup: /machine.slice/systemd-nspawn@my-new-machine.service
           ├─payload
           │ ├─init.scope
           │ │ └─59047 /lib/systemd/systemd
           │ └─system.slice
           │   ├─console-getty.service
           │   │ └─59092 /sbin/agetty -o -p -- \u --noclear --keep-baud console 115200,38400,9600 vt220
           │   ├─dbus.service
           │   │ └─59089 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
           │   ├─systemd-journald.service
           │   │ └─59080 /lib/systemd/systemd-journald
           │   ├─systemd-logind.service
           │   │ └─59090 /lib/systemd/systemd-logind
           │   ├─systemd-networkd.service
           │   │ └─59082 /lib/systemd/systemd-networkd
           │   └─systemd-resolved.service
           │     └─59087 /lib/systemd/systemd-resolved
           └─supervisor
             └─59033 /usr/bin/systemd-nspawn --quiet --keep-unit --boot --link-journal=try-guest --network-veth -U --settings=override

Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Started Console Getty.
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Reached target Login Prompts.
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Started Login Service.
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Reached target Multi-User System.
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Reached target Graphical Interface.
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]:          Starting Update UTMP about System Runlevel Changes...
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Started Update UTMP about System Runlevel Changes.
Nov 16 14:27:01 ananas.home.plexis.eu systemd-nspawn[59033]: [2B blob data]
Nov 16 14:27:01 ananas.home.plexis.eu systemd-nspawn[59033]: Ubuntu 18.04 LTS my-new-machine console
Nov 16 14:27:01 ananas.home.plexis.eu systemd-nspawn[59033]: [1B blob data]
▲ ~ █
```

machinectl – Status



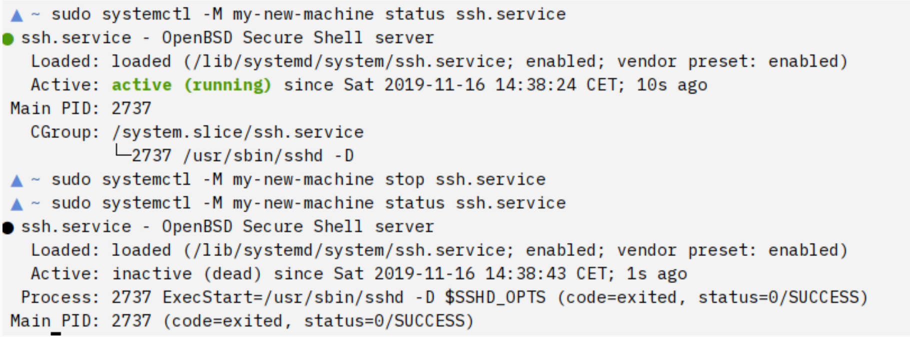
```
△ ~ sudo machinectl status my-new-machine
my-new-machine(845833ef3841414c9738cb3db92754f9)
           Since: Sat 2019-11-16 14:27:00 CET; 4min 46s ago
          Leader: 59047 (systemd)
         Service: systemd-nspawn; class container
            Root: /tmp/nspawn-root-1kLSv0
           Iface: ve-my-new-mach
         Address: 192.168.142.246
                  169.254.205.163
                  fe80::94b7:67ff:fe9a:ed8%10
              OS: Ubuntu 18.04 LTS
       UID Shift: 1550450688
            Unit: systemd-nspawn@my-new-machine.service
                  ├─payload
                  │ ├─init.scope
                  │ │ └─59047 /lib/systemd/systemd
                  │ └─system.slice
                  │   ├─console-getty.service
                  │   │ └─59092 /sbin/agetty -o -p -- \u --noclear --keep-baud console 115200,38400,9600 vt220
                  │   ├─dbus.service
                  │   │ └─59089 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
                  │   ├─systemd-journald.service
                  │   │ └─59080 /lib/systemd/systemd-journald
                  │   ├─systemd-logind.service
                  │   │ └─59090 /lib/systemd/systemd-logind
                  │   ├─systemd-networkd.service
                  │   │ └─59082 /lib/systemd/systemd-networkd
                  │   └─systemd-resolved.service
                  │     └─59087 /lib/systemd/systemd-resolved
                  └─supervisor
                    └─59033 /usr/bin/systemd-nspawn --quiet --keep-unit --boot --link-journal=try-guest --network-veth -U --settings=ove

Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Started Console Getty.
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Reached target Login Prompts.
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Started Login Service.
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Reached target Multi-User System.
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Reached target Graphical Interface.
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]:          Starting Update UTMP about System Runlevel Changes...
Nov 16 14:27:00 ananas.home.plexis.eu systemd-nspawn[59033]: [  OK  ] Started Update UTMP about System Runlevel Changes.
Nov 16 14:27:01 ananas.home.plexis.eu systemd-nspawn[59033]: [2B blob data]
Nov 16 14:27:01 ananas.home.plexis.eu systemd-nspawn[59033]: Ubuntu 18.04 LTS my-new-machine console
Nov 16 14:27:01 ananas.home.plexis.eu systemd-nspawn[59033]: [1B blob data]
```

machinectl – Networking and shell



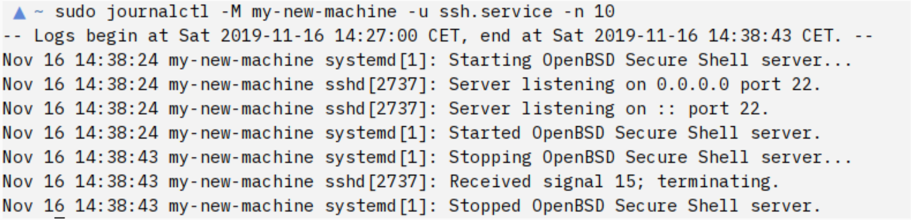
```
 ▲ ~ sudo machinectl shell my-new-machine
Connected to machine my-new-machine. Press ^] three times within 1s to exit session.
root@my-new-machine:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: host0@if10: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 96:b7:67:9a:0e:d8 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 169.254.205.163/16 brd 169.254.255.255 scope link host0
       valid_lft forever preferred_lft forever
    inet 192.168.142.246/28 brd 192.168.142.255 scope global dynamic host0
       valid_lft 3134sec preferred_lft 3134sec
    inet6 fe80::94b7:67ff:fe9a:ed8/64 scope link
       valid_lft forever preferred_lft forever
root@my-new-machine:~# logout
Connection to machine my-new-machine terminated.
 ▲ ~ sudo systemd-nspawn -M my-new-machine -a ls /
Disk image /var/lib/machines/my-new-machine.raw is currently busy.
```

systemctl and machined integration



```
▲ ~ sudo systemctl -M my-new-machine status ssh.service
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2019-11-16 14:38:24 CET; 10s ago
 Main PID: 2737
   CGroup: /system.slice/ssh.service
           └─2737 /usr/sbin/sshd -D
▲ ~ sudo systemctl -M my-new-machine stop ssh.service
▲ ~ sudo systemctl -M my-new-machine status ssh.service
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: inactive (dead) since Sat 2019-11-16 14:38:43 CET; 1s ago
  Process: 2737 ExecStart=/usr/sbin/sshd -D $SSHD_OPTS (code=exited, status=0/SUCCESS)
 Main PID: 2737 (code=exited, status=0/SUCCESS)
```

journald integration



```
▲ ~ sudo journalctl -M my-new-machine -u ssh.service -n 10
-- Logs begin at Sat 2019-11-16 14:27:00 CET, end at Sat 2019-11-16 14:38:43 CET. --
Nov 16 14:38:24 my-new-machine systemd[1]: Starting OpenBSD Secure Shell server...
Nov 16 14:38:24 my-new-machine sshd[2737]: Server listening on 0.0.0.0 port 22.
Nov 16 14:38:24 my-new-machine sshd[2737]: Server listening on :: port 22.
Nov 16 14:38:24 my-new-machine systemd[1]: Started OpenBSD Secure Shell server.
Nov 16 14:38:43 my-new-machine systemd[1]: Stopping OpenBSD Secure Shell server...
Nov 16 14:38:43 my-new-machine sshd[2737]: Received signal 15; terminating.
Nov 16 14:38:43 my-new-machine systemd[1]: Stopped OpenBSD Secure Shell server.
```

Not using sudo

- machined is polkit enabled
- Permissions in org.freedesktop.machine1.*

/etc/polkit-1/rules.d/machined.rules

```
polkit.addRule(function(action, subject) {
    if (action.id == "org.freedesktop.machine1.manage-images" &&
        subject.isInGroup("users")) {
      return polkit.Result.YES;
    }
});
```

Creating container images

Traditional bootstrap tools

- `debootstrap`, `dnf`, `pacstrap`, `zypper`
- Use `systemd-nspawn -a […] passwd root` to set a root password
- Do ensure that `systemd-container` is installed
- Using `systemd-networkd` is highly recommended

docker build

```
docker build -t nspawn-test:latest
cd /var/lib/machines
mkdir nspawn-from-docker
docker export \
  $(docker create nspawn-test:latest true) | \
    tar -x -C nspawn-from-docker
```

mkosi: Make Operating System Image

"Build Legacy-Free OS Images"[7]

- From the systemd team[8]
- Many output formats:
  - Raw GPT disks (ext4, xfs or btrfs), optionally bootable, optionally LUKS'ed
  - QCOW2 image of the above
  - squashfs image
  - Directory, tarball
  - btrfs subvolume
- Checksumming and signing images

[7] systemd/mkosi

[8] 0pointer.net blog

mkosi: Files and directories

- mkosi.default – Configuration file: distro, output etc.
- mkosi.postinst – Script run after install of packages
- mkosi.nspawn – systemd-nspawn configuration for the resulting image
- mkosi.skeleton/ – Directory tree copied before OS install
- mkosi.extra/ – Directory tree copied after OS install
- mkosi.cache/ – Packages are stores here for faster rebuilds
- mkosi.rootpw – Root password (may be hashed)
- mkosi.passphrase – LUKS passphrase file

mkosi files: mkosi.default

```
[Distribution]
Distribution=debian
Release=buster

[Output]
Format=tar

[Packages]
Packages=systemd-container
```

mkosi build

```
$ sudo mkosi build
[..]

$ ls -lsh
total 57M
 57M -rw-r--r-- 1 root   root   57M May  3 12:13 image.tar.xz
4.0K drwxr-xr-x 3 lieter users 4.0K May  3 12:10 mkosi.cache
4.0K -rw-r--r-- 1 lieter users  110 May  3 12:10 mkosi.default

$ sudo machinectl import-tar image.tar.xz debian-buster-base
```

Conclusion

Conclusion

`systemd-nspawn` offers

- long-lived containers
- extremely customizable runtime configuration
- full integration in the systemd “ecosystem”
- a solution for those that don’t need
    - Image layering
    - Orchestration frameworks
- building blocks for advanced infrastructures

`mkosi` offers

- lightweight, customizable image building

Thanks

Questions?