Doing stupid things with FreeBSD jails

recommend you do not try at home

Dan Langille 2024.09.22 - EuroBSDCon Dublin

First a word about stupid

No, not a self description

- Although I say stupid

What you won’t get

- how-to guides
- detailed instructions
- promises or guarantees
- anything close to useful

What you will get

- some things to avoid
- insights into how my mind wanders
- ideas and concepts for future projects
- prompts to email me later with better ideas

A brief history of Dan’s time with jail

The origin story

- jails arrived with FreeBSD 4.0 (2000)
- I started playing with them since at least FreeBSD 4.1 (also 2000)
- My first documented usage was creating a jail for Open Source Weekend (OSW) - Ottawa, OCLUG in November 2003
- I’ve been using jails on a near-daily basis for many simple solutions
- I recommend you do not try at home without adequate adult supervision
- One thing leads to another…

Abbreviated list of stupid things

part 1 - as listed in talk description - perhaps not in order

- Let's Encrypt via acme.sh, hidden DNS master, public DNS servers, and a public website to distribute new certs via anvil
- FreshPorts uses three jails for ingesting commits and displaying them on the website
- database testing - loads each daily database backup to test it works
- building packages via poudriere in a jail

Abbreviated list of foolish things

part 2

- multiple jails running PostgreSQL
- running a jail within a jail
- modifying pkg-audit to ignore certain jails
- why I had to create a website proxy for all the in-house websites
- having a portal jail on a hosting provider as a front end for jails running in his basement

Abbreviated list of foolish things

part 3

- Time Machines for Apple hardware
- Using SamDrucker to keep track of what packages are installed where
- Applications in their own jails include PostgreSQL, gitea, MSQL, LibreNMS, named, Unifi, Mosquitto
- Why you should use pushover.net

What we are not covering

jail details

- Creation
- Updating (i.e. patching the OS)
- Upgrading (moving to a new release)
- jail managers (or lack thereof)
- general maintenance

Jail management methods

Use what you want, enjoy them

- Started with manual
- Went to ezjail
- Migrated everything to iocage
- Migrated everything to py-iocage
- Migrated everything to jail.conf
- I use mkjaill (full disclosure: I am a contributor / maintainer)

First things first

Usually the website

- Often, the first thing people jail is a webserver
- Let’s get that nasty exploit source isolated; protect the data; protect the host
- Rather straight forward
- often involves redirection of the port to the jail’s IP address
- might not, if you just use the host’s IP address and have nothing else on port 80/443

Next, other web applications

Might as well jail them too

- Into a new jail they go
- sometimes the same jail
- sometimes a new jail
- webs01 - apache based - 11 websites
- webs02 - WordPress blogs and nginx-based stuff - 10 websites

Kicking the baddies

fail2ban

- fail2ban works great on the host
- monitor multiple jails
- ban at the host
- I run this at home and in production

Hey wow, that was easy

Let’s jail the database server

- PostgreSQL - is there any other open source database server?
- Interesting System V share memory issues
- No longer such an issue
- Over time, had multiple PostgreSQL and MySQL instances on the same host: pg9, pg10, pg11, pg12, pg13, pg14, pg15, mysql57, mysql80
- More on that ^ later

What a neat idea!

Let’s move the data into another filesystem

```
[11:43 r730-01 dvl ~] % zfs list | grep pg
data02/jails/dev-pgeu                8.22G   686G  7.11G  /jails/dev-pgeu
data02/jails/pg01                    14.6G   686G  13.3G  /jails/pg01
data02/jails/pg02                    11.1G   686G  9.74G  /jails/pg02
data02/jails/pg03                    14.0G   686G  8.51G  /jails/pg03
data03/pg02                          27.6M  6.08T    88K  none
data03/pg02/postgres                 26.5M  6.08T  18.4M  /jails/pg02/var/db/postgres
data03/pg02/rsyncer                   968K  6.08T   112K  /jails/pg02/usr/home/rsyncer/backups
data03/pg03                           756G  6.08T    88K  none
data03/pg03/postgres                  560G  6.08T   534G  /jails/pg03/var/db/postgres
data03/pg03/rsyncer                   196G  6.08T  32.8G  /jails/pg03/usr/home/rsyncer/backups
data03/poudriere/ports/pgeu           971M  6.08T   971M  /usr/local/poudriere/ports/pgeu
```

Mounting file systems within jails

Are they mounted or not mounted?

- Stop the jail
- manipulate the filesystem
- can’t, because something is mounted in there

```
exec.created+="zfs set jailed=on data02/…/cache/categories”;
exec.poststop+="zfs set jailed=off data02/…/cache/categories”;
exec.poststop+="zfs umount data02/…/cache/categories”;
```

Doing good snapshot backups

via Bacula - is there any other open source backup solution?

- snapshot the filesystems you want to backup
- backup the snapshots you just created
- destroy the snapshots

- see `https://git.langille.org/dvl/sundry-scripts`

Specific package needs

I have so much running, I need monitoring, special options, etc

- The FreeBSD package repos are great
- However, if you want:
  - non-default package configuration
  - package vulns patched overnight
  - different version of Python on different hosts
  - THEN YOU might want to build your own packages

poudriere

builder of packages

- great tool for building packages
- fantastic tool for port maintainers - testport is a gift
- Started using it in 2014 (FreeBSD 9.2)
- By 2019, I wanted it in a jail - because that’s the thing to do
- Took a while; success in Oct 2019
- see `https://dan.langille.org/2024/01/19/configuration-for-running-poudriere-in-a-jail-on-freebsd-14/`

See how quickly this escalates?

jails beget jails

Monitoring, all the things

If you’re not monitoring, is it really running?

- Been using Nagios since 2010
- It’s in a jail, but a webserver jail, not a nagios-only jail
- custom build options for plugins (hence poudriere):

```
net-mgmt_nagios-plugins_SET+=PGSQL
net-mgmt_nagios-plugins_UNSET+=DNS_BASE
net-mgmt_nagios-plugins_SET+=MYSQL
net-mgmt_nagios-plugins_SET+=DNS_BINDTOOLS
```

- So old, uses `/usr/websites`, not `/usr/local/www`

Monitor vulns - OS and apps

```
% cd /usr/local/etc/periodic/security/
% ls *audit*
405.pkg-base-audit
410.pkg-audit
```

- Use them
- You’ll find vulns you never knew you had
- see `https://git.langille.org/dvl/nagios`

	Status	Last Check	Duration	Attempt	Status Information
fix_queue	WARNING	05-28-2023 10:40:38	1d 10h 0m 54s	4/4	7 mail(s) in queue
	CRITICAL	05-28-2023 09:36:38	0d 1h 8m 46s	4/4	Checking for security vulnerabilities in base (userland & kernel): Host system: vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: keycloak vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: dns1 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: cliff2 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: mysql01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pg01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pg02 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pg03 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: dev-ingress01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: dev-ingress01.freshports vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: dev-nginx01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: test-ingress01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: test-ingress01.freshports vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: stage-ingress01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: stage-ingress01.freshports vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: bacula vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: certs-rsync vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: certs vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: git vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: svn vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: webserver vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: zm vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: dev-pgeu vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: bacula-sd-02 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: jail-testing vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: talos vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: samdrucker vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: nsnotify vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: bacula-sd-03 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: fileserver vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: mydev vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: jail_within_jail vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: serpico vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: dns-hidden-master vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: unifi01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: besser vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: mqtt01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: test-nginx01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: stage-nginx01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-01 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-01-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-03 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-03-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-07 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-24 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-07-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-24-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-06 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-06-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-05 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-25 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-05-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-25-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-17 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-17-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-09 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-22 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-09-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-22-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-13 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-12 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-16 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-21 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-12-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-13-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-16-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-21-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-08 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-08-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-10 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-10-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-20 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-20-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-15 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-15-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-18 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-18-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-02 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-02-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-14 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-14-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-23 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-23-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-19 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-19-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-04 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-04-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-27 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-27-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-11 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-11-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-26 vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found. jail: pkg01.131amd64-default-pg15-job-26-n vulnxml file up-to-date 0 problem(s) in 0 installed package(s) found.

rvices

Oh, wait, but not *those* jails

Don’t monitor jails started by poudriere

- transient, no network, no
- Added:

```
security_status_baseaudit_jails_ignore_wild="pkg01.14"
security_status_pkgaudit_jails_ignore_wild="pkg01.14"
```

- see `https://github.com/freebsd/pkg/commit/4c72d06e3559b62a0694c6eccc1edf27fa724b17`

To test that, I needed a jail within a jail

Inception

- Could have used bhyve … but a jail was easier
- or … was it?

- see `https://git.langille.org/dvl/sundry-scripts`

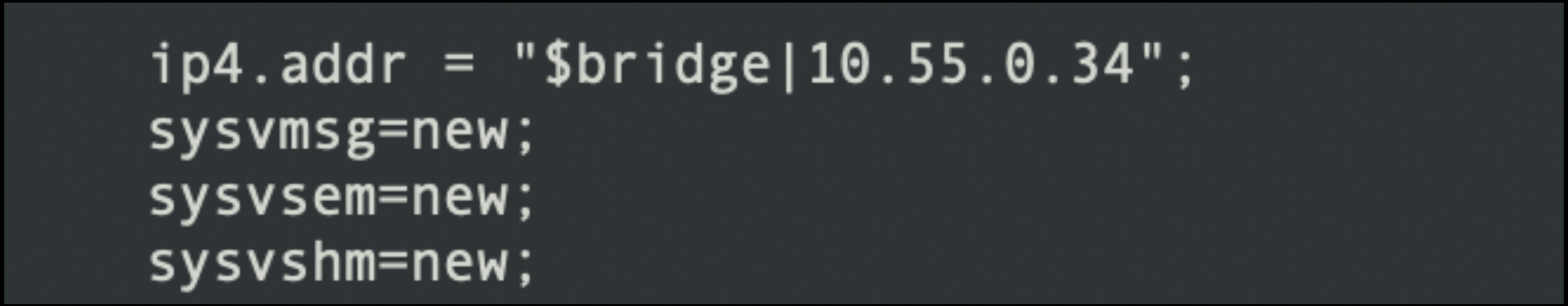
FreshPorts

The Place For Ports

- parses git commit logs
- **ingress** jail (pulls in commits)
- **website** jail (pulls data from database and displays it to you)
- **database** jail (holds the commit information for use by above two jails)
- Nice separation
- take down website, without affecting commit processing
- Interrupt commit processing without affecting website

The PostgreSQL jail - today

So very simple



```
ip4.addr = "$bridge|10.55.0.34";
sysvmsg=new;
sysvsem=new;
sysvshm=new;
```

Multiple PostgreSQL jails - same host

Shared memory required different user ids - back in 2007

Shared memory and UID

PostgreSQL makes use of shared memory. When running multiple instances of PostgreSQL the shared memory for one instance can be stomped on by another instance. That's not nice. The key to avoiding this is using a different UID for each instance. You can see that here:

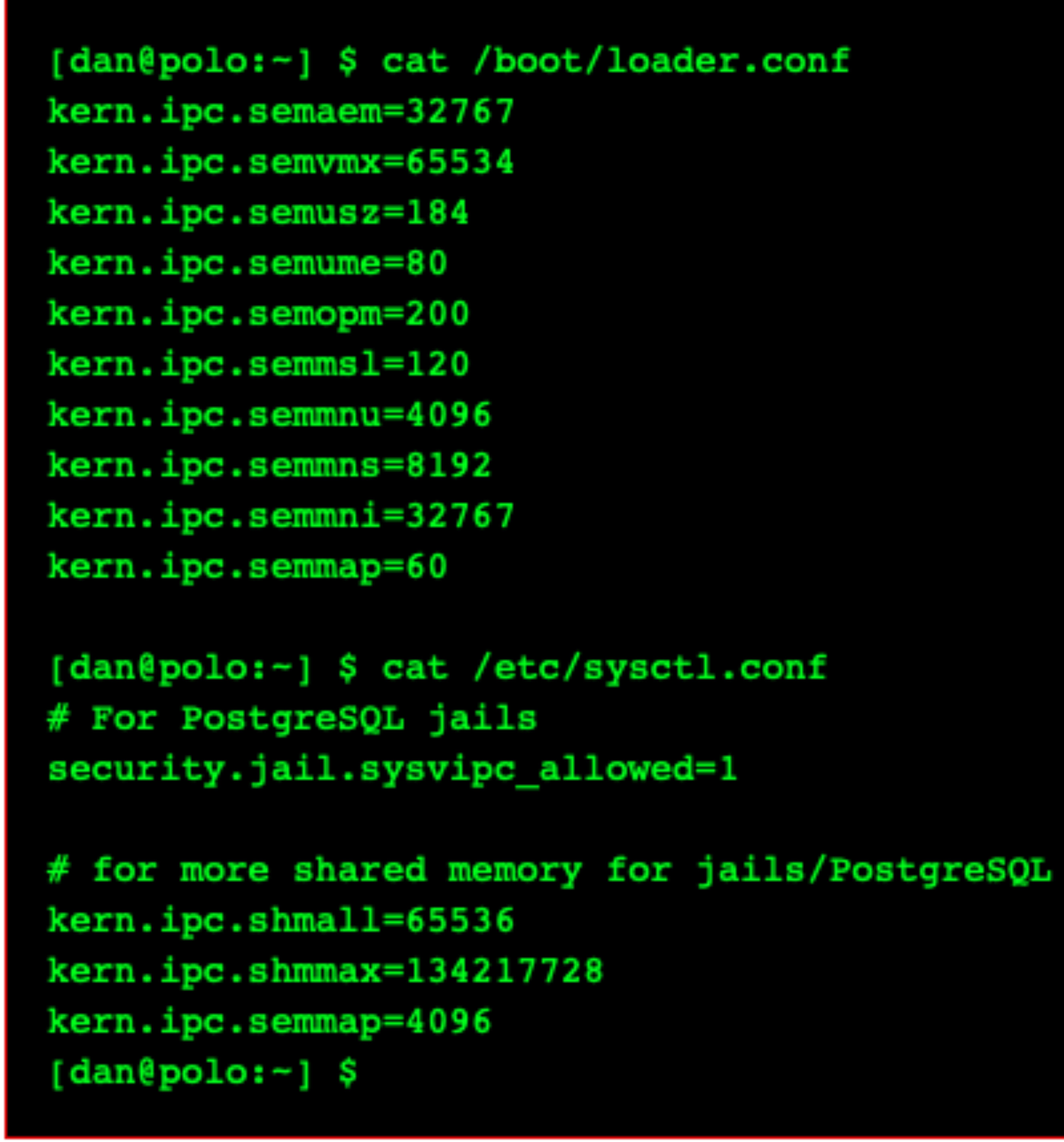
```
$ grep -h pgsql /usr/jail/*.unixathome.org/etc/passwd
pgsql:*:1073:70:PostgreSQL Daemon:/usr/local/pgsql:/bin/sh
pgsql:*:1074:70:PostgreSQL Daemon:/usr/local/pgsql:/bin/sh
pgsql:*:1080:70:PostgreSQL Daemon:/usr/local/pgsql:/bin/sh
pgsql:*:1081:70:PostgreSQL Daemon:/usr/local/pgsql:/bin/sh
pgsql:*:1082:70:PostgreSQL Daemon:/usr/local/pgsql:/bin/sh
```

I used a UID that would relate to the version of PostgreSQL that was running. For example, UID=1073 is PostgreSQL version 7.3. There is no need to follow this convention. I did it merely because I could.

The PostgreSQL jail

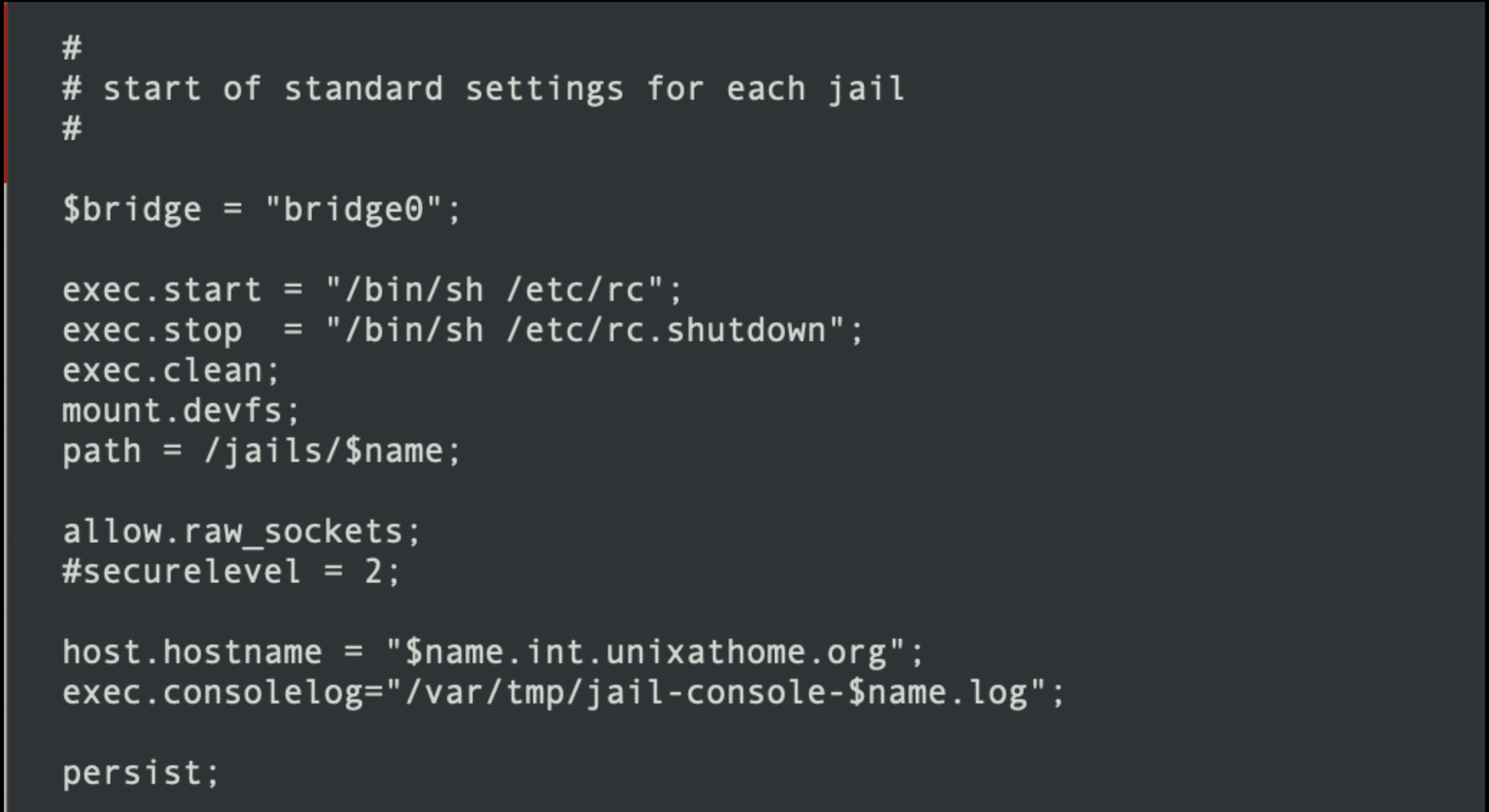
Wasn’t always so easy

```
# /etc/rc.conf
jail_sysvipc_allow="YES"
```



```
[dan@polo:~] $ cat /boot/loader.conf
kern.ipc.semaem=32767
kern.ipc.semvmx=65534
kern.ipc.semusz=184
kern.ipc.semume=80
kern.ipc.semopm=200
kern.ipc.semmsl=120
kern.ipc.semmnu=4096
kern.ipc.semmns=8192
kern.ipc.semmni=32767
kern.ipc.semmap=60

[dan@polo:~] $ cat /etc/sysctl.conf
# For PostgreSQL jails
security.jail.sysvipc_allowed=1

# for more shared memory for jails/PostgreSQL
kern.ipc.shmall=65536
kern.ipc.shmmax=134217728
kern.ipc.semmap=4096
[dan@polo:~] $
```

My standard jail settings (/etc/jail.conf.d/)



```
#
# start of standard settings for each jail
#

$bridge = "bridge0";

exec.start = "/bin/sh /etc/rc";
exec.stop  = "/bin/sh /etc/rc.shutdown";
exec.clean;
mount.devfs;
path = /jails/$name;

allow.raw_sockets;
#securelevel = 2;

host.hostname = "$name.int.unixathome.org";
exec.consolelog="/var/tmp/jail-console-$name.log";

persist;
```

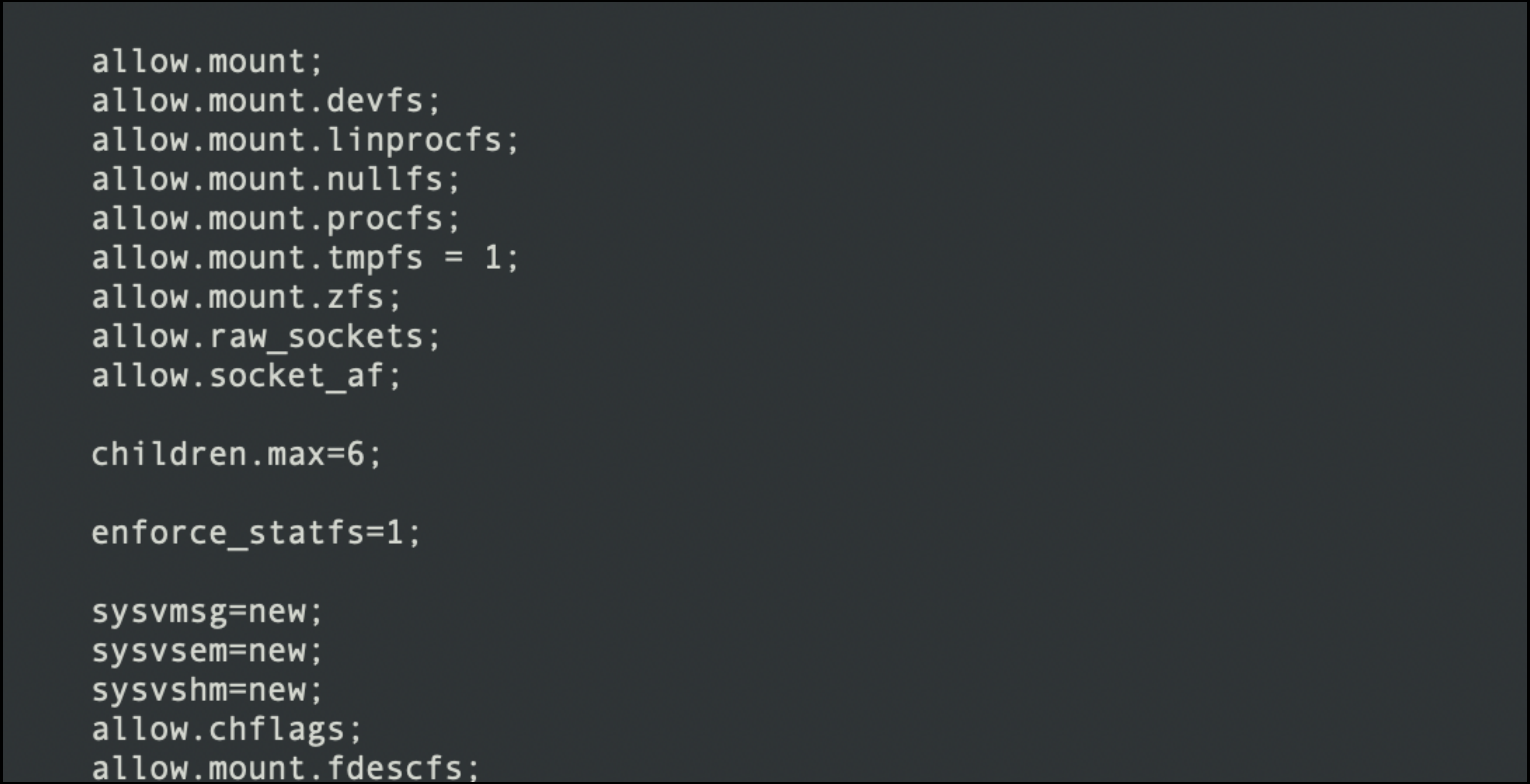
The ingress jail

part 1

```
ip4.addr = "$bridge|10.55.0.37";

allow.mount=true;
allow.mount.zfs=true;
enforce_statfs=1;
devfs_ruleset=5;

# because want to mount zfs, we do that before the rc start up
# not sure if we MUST do it in that order.
exec.start="zfs mount -a";
exec.start+="/bin/sh /etc/rc";

exec.created+="zfs set jailed=on data02/freshports/jailed/dev-ingress01";
exec.created+="zfs jail $name    data02/freshports/jailed/dev-ingress01";
```

The ingress jail

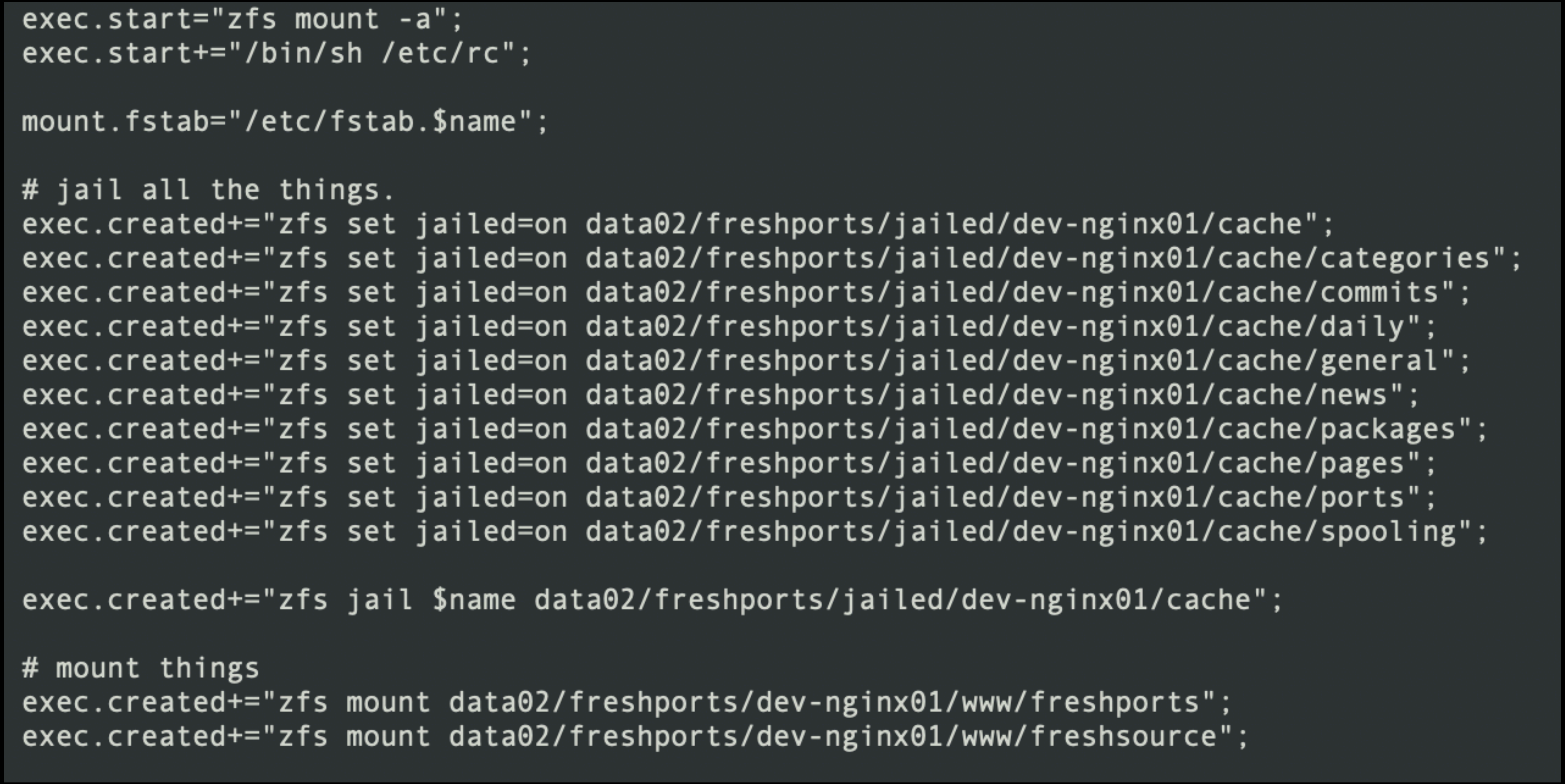
part 2



```
allow.mount;
allow.mount.devfs;
allow.mount.linprocfs;
allow.mount.nullfs;
allow.mount.procfs;
allow.mount.tmpfs = 1;
allow.mount.zfs;
allow.raw_sockets;
allow.socket_af;

children.max=6;

enforce_statfs=1;

sysvmsg=new;
sysvsem=new;
sysvshm=new;
allow.chflags;
allow.mount.fdescfs;
```

The webserver jail

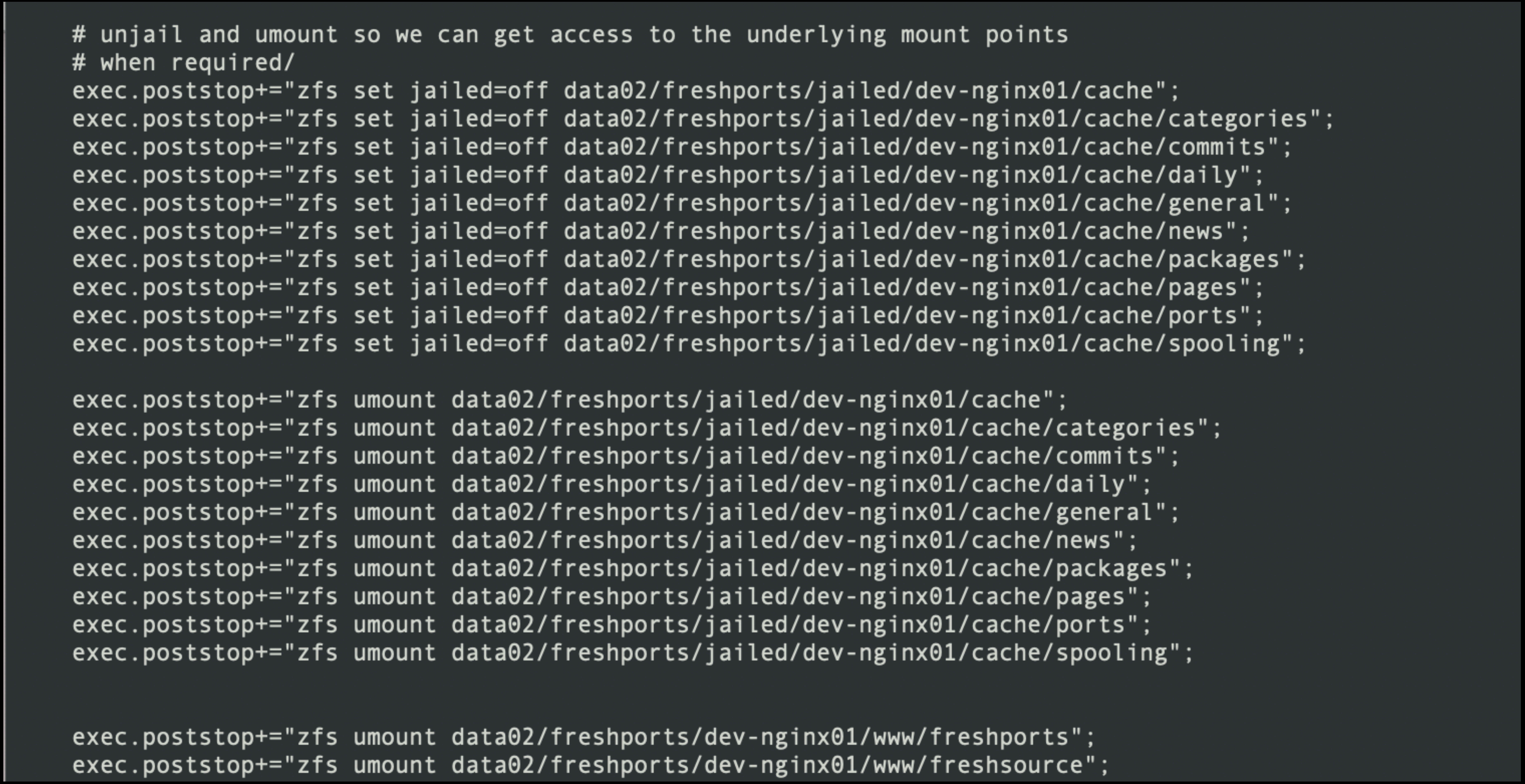
Part 1



```
exec.start="zfs mount -a";
exec.start+="/bin/sh /etc/rc";

mount.fstab="/etc/fstab.$name";

# jail all the things.
exec.created+="zfs set jailed=on data02/freshports/jailed/dev-nginx01/cache";
exec.created+="zfs set jailed=on data02/freshports/jailed/dev-nginx01/cache/categories";
exec.created+="zfs set jailed=on data02/freshports/jailed/dev-nginx01/cache/commits";
exec.created+="zfs set jailed=on data02/freshports/jailed/dev-nginx01/cache/daily";
exec.created+="zfs set jailed=on data02/freshports/jailed/dev-nginx01/cache/general";
exec.created+="zfs set jailed=on data02/freshports/jailed/dev-nginx01/cache/news";
exec.created+="zfs set jailed=on data02/freshports/jailed/dev-nginx01/cache/packages";
exec.created+="zfs set jailed=on data02/freshports/jailed/dev-nginx01/cache/pages";
exec.created+="zfs set jailed=on data02/freshports/jailed/dev-nginx01/cache/ports";
exec.created+="zfs set jailed=on data02/freshports/jailed/dev-nginx01/cache/spooling";

exec.created+="zfs jail $name data02/freshports/jailed/dev-nginx01/cache";

# mount things
exec.created+="zfs mount data02/freshports/dev-nginx01/www/freshports";
exec.created+="zfs mount data02/freshports/dev-nginx01/www/freshsource";
```

The webserver jail

Part 2



```
# unjail and umount so we can get access to the underlying mount points
# when required/
exec.poststop+="zfs set jailed=off data02/freshports/jailed/dev-nginx01/cache";
exec.poststop+="zfs set jailed=off data02/freshports/jailed/dev-nginx01/cache/categories";
exec.poststop+="zfs set jailed=off data02/freshports/jailed/dev-nginx01/cache/commits";
exec.poststop+="zfs set jailed=off data02/freshports/jailed/dev-nginx01/cache/daily";
exec.poststop+="zfs set jailed=off data02/freshports/jailed/dev-nginx01/cache/general";
exec.poststop+="zfs set jailed=off data02/freshports/jailed/dev-nginx01/cache/news";
exec.poststop+="zfs set jailed=off data02/freshports/jailed/dev-nginx01/cache/packages";
exec.poststop+="zfs set jailed=off data02/freshports/jailed/dev-nginx01/cache/pages";
exec.poststop+="zfs set jailed=off data02/freshports/jailed/dev-nginx01/cache/ports";
exec.poststop+="zfs set jailed=off data02/freshports/jailed/dev-nginx01/cache/spooling";

exec.poststop+="zfs umount data02/freshports/jailed/dev-nginx01/cache";
exec.poststop+="zfs umount data02/freshports/jailed/dev-nginx01/cache/categories";
exec.poststop+="zfs umount data02/freshports/jailed/dev-nginx01/cache/commits";
exec.poststop+="zfs umount data02/freshports/jailed/dev-nginx01/cache/daily";
exec.poststop+="zfs umount data02/freshports/jailed/dev-nginx01/cache/general";
exec.poststop+="zfs umount data02/freshports/jailed/dev-nginx01/cache/news";
exec.poststop+="zfs umount data02/freshports/jailed/dev-nginx01/cache/packages";
exec.poststop+="zfs umount data02/freshports/jailed/dev-nginx01/cache/pages";
exec.poststop+="zfs umount data02/freshports/jailed/dev-nginx01/cache/ports";
exec.poststop+="zfs umount data02/freshports/jailed/dev-nginx01/cache/spooling";


exec.poststop+="zfs umount data02/freshports/dev-nginx01/www/freshports";
exec.poststop+="zfs umount data02/freshports/dev-nginx01/www/freshsource";
```

The great jail migration

thin to thick to vanilla

thin to thick

migration from ezjail to iocage

- I tired of having to update all the jails at once
- It broke too many things - you also have to upgrade ALL YOUR JAILS RIGHT NOW (consider binary changes in major releases) - apps need updating
- Migration via a script helped (see URL)
- See `https://dan.langille.org/2019/04/08/converting-thin-jails-to-thick-jails/`

Thick to vanilla

migration from iocage to plain vanilla jails

- Vanilla is straight forward
- Configuration file is plain text
- what you see is what you specified

- See `https://dan.langille.org/2021/02/28/converting-an-iocage-jail-to-a-vanilla-jail/`

mkjail

not for configuration

- create (filesystems)
- update (patch the os)
- upgrade (upgrade the OS to a new release)
- All configuration is in the jail configuration files

- see `https://github.com/mkjail/mkjail`

So many websites

only one rdr

- Many websites, on different hosts
- Can’t just redirect port 80/443
- I need a web proxy
- I know, I’ll create a jail
- That’s how serpico was born.

serpico

Now that I have so many websites in the basement, let’s hide it

dan — ssh serpico — 110×12

```
[22:50 serpico dvl /usr/local/etc/nginx/includes] % ls
_default.conf                  fedex.unixathome.org.conf              proxy_set_header.inc
beta.bsdcan.org.conf           fretbsd.unixathome.org.conf            serpico.unixathome.org.conf
beta.pgcon.org.conf            fretbsd.unixathome.org.confpass        services.unixathome.org.conf
dansdirtyclothes.net.conf      git.langille.org.conf                  ssl-common.inc
dev-pgeu.pgcon.org.conf        ha.unixathome.org.conf.disabled        stage.freshports.org.conf
dev.freshports.org.conf        kvm01-proxy.int.unixathome.org.conf    stage.freshsource.org.conf
dev.freshsource.org.conf       laundry.unixathome.org.conf            test.freshports.org.conf
dvl.freshports.org.conf        m.freshports.org.conf.inactive         test.freshsource.org.conf
dvl.freshsource.org.conf       nginx-status.conf                      ups02-proxy.int.unixathome.org.conf
fedex.int.unixathome.org.conf  proxy_set_header-no-host.inc
[22:53 serpico dvl /usr/local/etc/nginx/includes] %
```

```
proxy_pass https://dev-freshports.int.unixathome.org/;
```

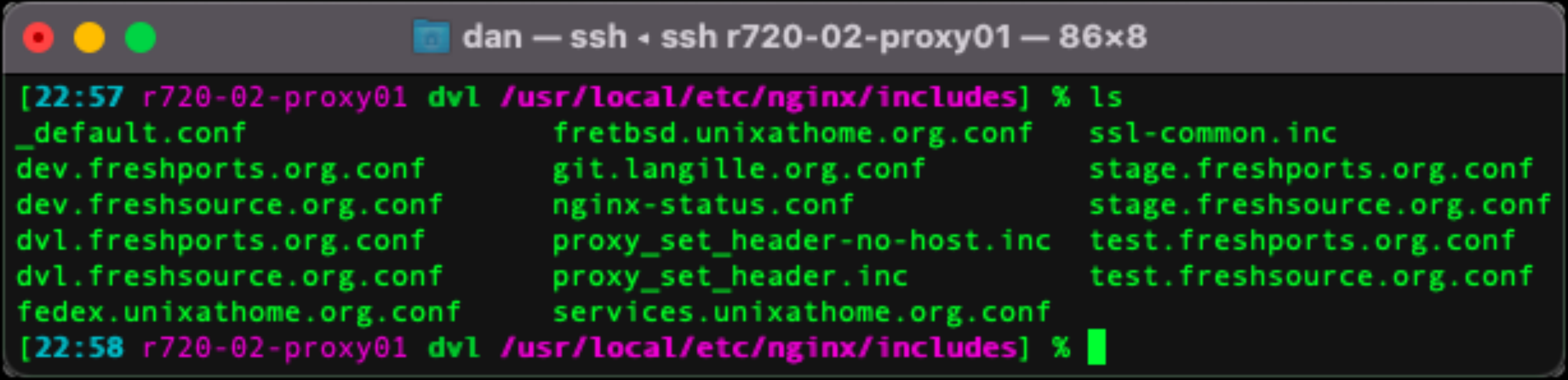
Web proxy - on the web

Now that I have so many websites in the basement, let’s hide it

- Instead of keeping a CNAME active and everyone knowing the IP address at home..
- Replicate serpico to a server in a data center…
- Now the IP addresses point to the box in the data center
- Of course, it’s running in a jail

Web proxy - on the web

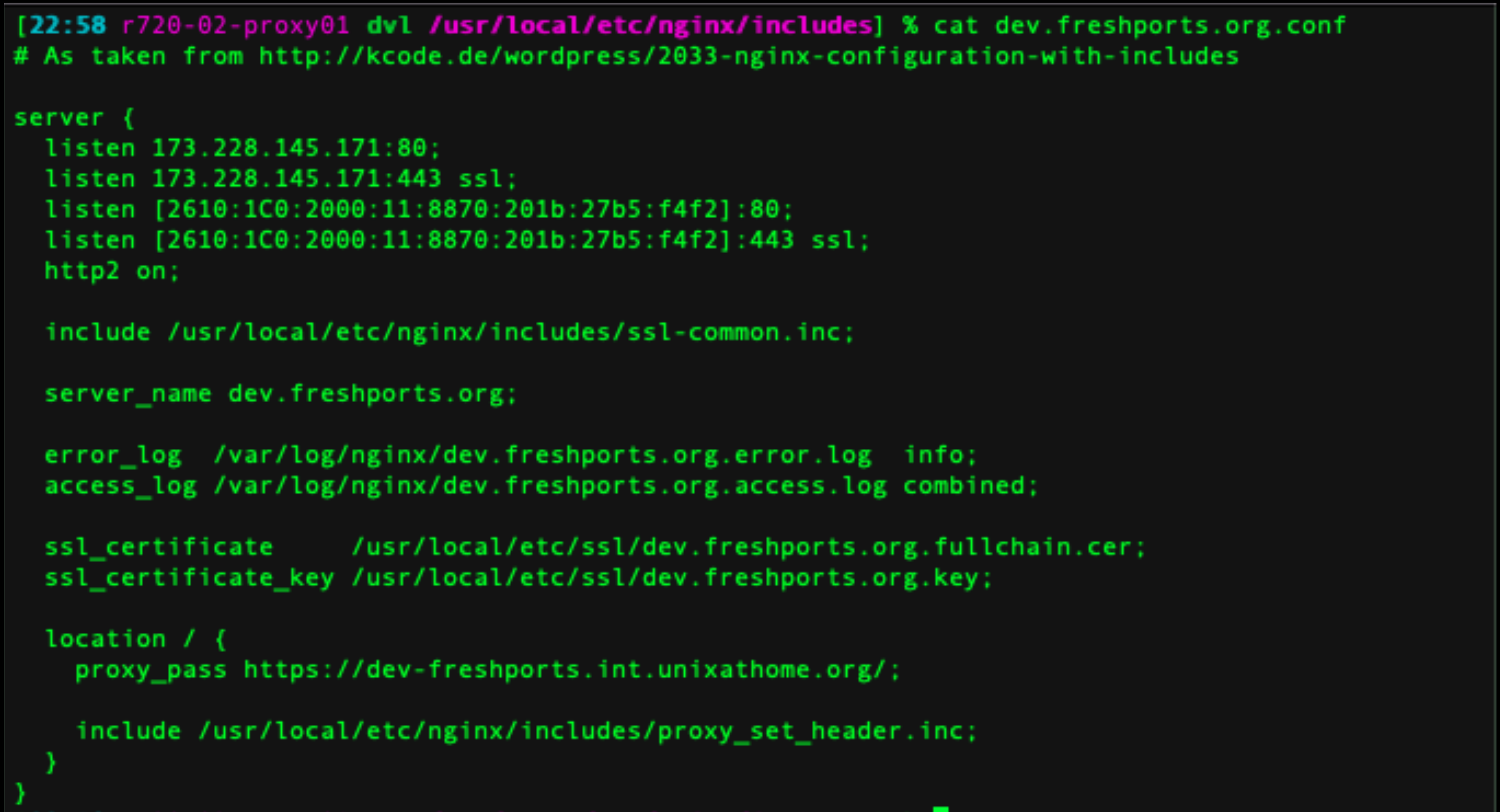
Much smaller than serpico



```
[22:57 r720-02-proxy01 dvl /usr/local/etc/nginx/includes] % ls
_default.conf                  fretbsd.unixathome.org.conf    ssl-common.inc
dev.freshports.org.conf        git.langille.org.conf          stage.freshports.org.conf
dev.freshsource.org.conf       nginx-status.conf              stage.freshsource.org.conf
dvl.freshports.org.conf        proxy_set_header-no-host.inc   test.freshports.org.conf
dvl.freshsource.org.conf       proxy_set_header.inc           test.freshsource.org.conf
fedex.unixathome.org.conf      services.unixathome.org.conf
[22:58 r720-02-proxy01 dvl /usr/local/etc/nginx/includes] %
```

Web proxy - on the web

Example configuration



```
[22:58 r720-02-proxy01 dvl /usr/local/etc/nginx/includes] % cat dev.freshports.org.conf
# As taken from http://kcode.de/wordpress/2033-nginx-configuration-with-includes

server {
  listen 173.228.145.171:80;
  listen 173.228.145.171:443 ssl;
  listen [2610:1C0:2000:11:8870:201b:27b5:f4f2]:80;
  listen [2610:1C0:2000:11:8870:201b:27b5:f4f2]:443 ssl;
  http2 on;

  include /usr/local/etc/nginx/includes/ssl-common.inc;

  server_name dev.freshports.org;

  error_log  /var/log/nginx/dev.freshports.org.error.log  info;
  access_log /var/log/nginx/dev.freshports.org.access.log combined;

  ssl_certificate     /usr/local/etc/ssl/dev.freshports.org.fullchain.cer;
  ssl_certificate_key /usr/local/etc/ssl/dev.freshports.org.key;

  location / {
    proxy_pass https://dev-freshports.int.unixathome.org/;

    include /usr/local/etc/nginx/includes/proxy_set_header.inc;
  }
}
```

Web proxy - on the web

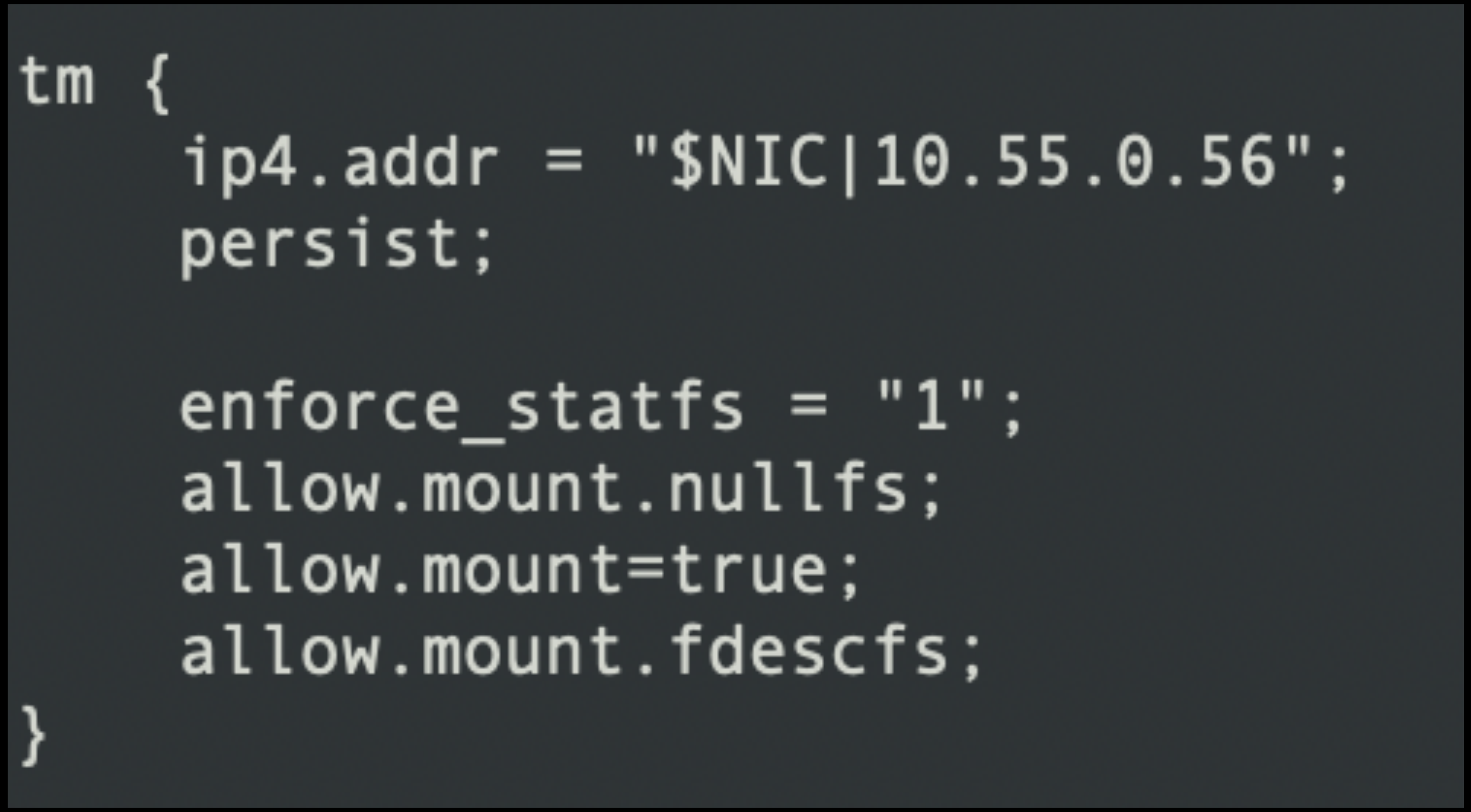
Common proxy stuff

dan — ssh ◂ ssh r720-02-proxy01 — 86×14

```
[[23:00 r720-02-proxy01 dvl /usr/local/etc/nginx/includes] % cat /usr/local/etc/nginx/i]
ncludes/proxy_set_header.inc
# this is meant to be included in every host which is proxied.

    proxy_http_version 1.1;

    proxy_set_header Host              $http_host;

    proxy_set_header X-Real-IP         $remote_addr;
    proxy_set_header X-Forwarded-For   $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Host  $host;
    proxy_set_header X-Forwarded-Port  $server_port;
    proxy_set_header X-Forwarded-Proto $scheme;
[23:00 r720-02-proxy01 dvl /usr/local/etc/nginx/includes] %
```

I got ZFS, why not backup my Mac?

Time Machines for Apple Hardware



```
tm {
    ip4.addr = "$NIC|10.55.0.56";
    persist;

    enforce_statfs = "1";
    allow.mount.nullfs;
    allow.mount=true;
    allow.mount.fdescfs;
}
```

- see `https://dan.langille.org/2024/01/06/creating-a-time-capsule-instance-using-samba-freebsd-and-zfs-2/`

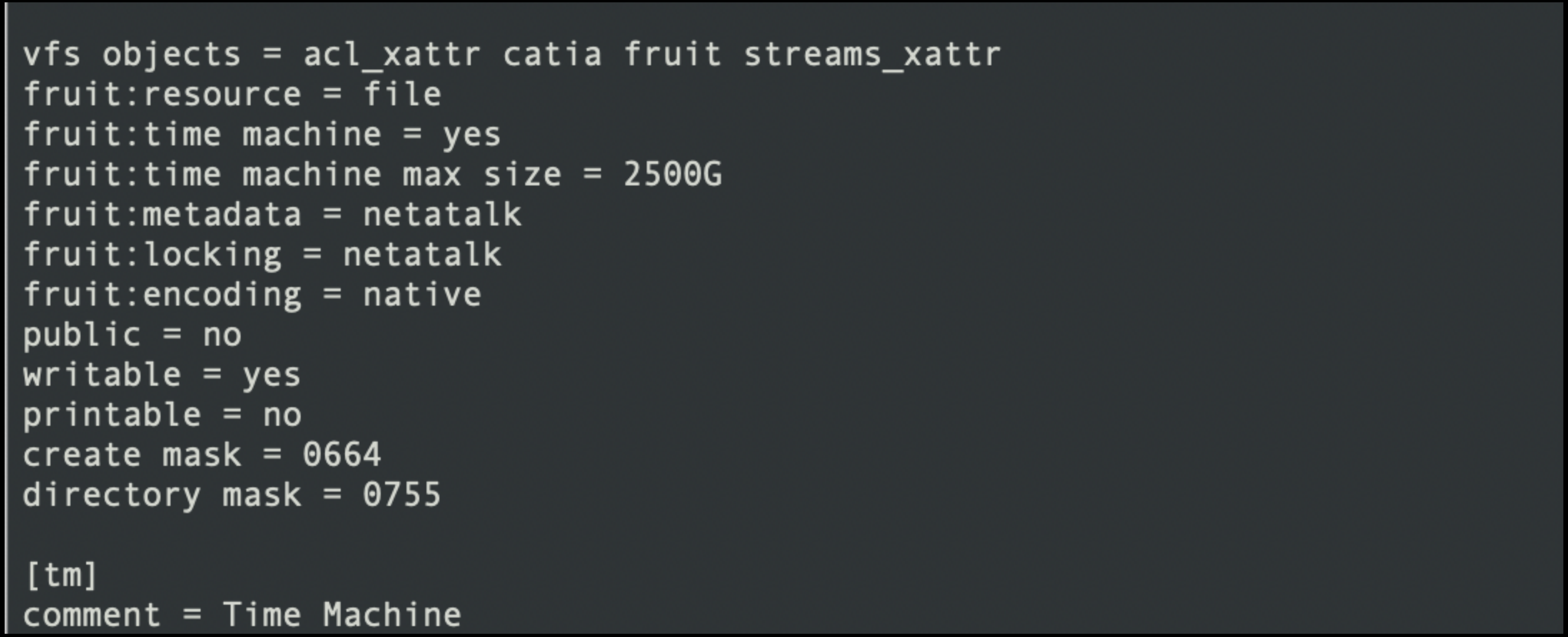
Time Machines for Apple Hardware

The samba configuration 1/2

```
[[18:15 tm dvl ~] % cat /usr/local/etc/smb4.conf
# This instance is used only for Time Machines. Nothing else.
# from https://bsky.app/profile/sweordbora.hausen.com/post/3kafje4ovq52z

[global]

workgroup = TimeMachine

# add these two lines to avoid smbd_open_one_socket: open_socket_in failed: Protocol not supported
bind interfaces only = yes
interfaces = bridge0

remote announce = 10.55.0.255
security = user
encrypt passwords = yes

# re: https://github.com/mbentley/docker-timemachine/issues/105#issuecomment-1130483951
#server min protocol = SMB2

path = /usr/local/timemachine/%U


# Taken from https://forums.freebsd.org/threads/samba-functions-but-unable-to-use-it-as-a-macos-time-machine-destination.79896/


fruit:aapl = yes
fruit:nfs_aces = yes
fruit:copyfile = no
fruit:model = MacSamba
```

Time Machines for Apple Hardware

The samba configuration 2/2



```
vfs objects = acl_xattr catia fruit streams_xattr
fruit:resource = file
fruit:time machine = yes
fruit:time machine max size = 2500G
fruit:metadata = netatalk
fruit:locking = netatalk
fruit:encoding = native
public = no
writable = yes
printable = no
create mask = 0664
directory mask = 0755

[tm]
comment = Time Machine
```

Time Machines for Apple Hardware

zfs list

```
[[18:10 r730-03 dvl ~] % zfs list | grep tm
 data01/jails/tm                                    4.29G  8.23T  2.17G  /jails/tm
 data01/timemachine                                 3.13T  8.23T   112K  /jails/tm/usr/local/timemachine
 data01/timemachine/dvl-air01                        273G  8.23T   241G  /jails/tm/usr/local/timemachine/dvl-air01
 data01/timemachine/dvl-pro02                       1.01T   467G   316G  /jails/tm/usr/local/timemachine/dvl-pro02
 data01/timemachine/dvl-pro03                        116G   908G   116G  /jails/tm/usr/local/timemachine/dvl-pro03
 data01/timemachine/dvl-pro04                        906G  8.23T   804G  /jails/tm/usr/local/timemachine/dvl-pro04
```

snapshots provided and maintained by sanoid

SamDrucker

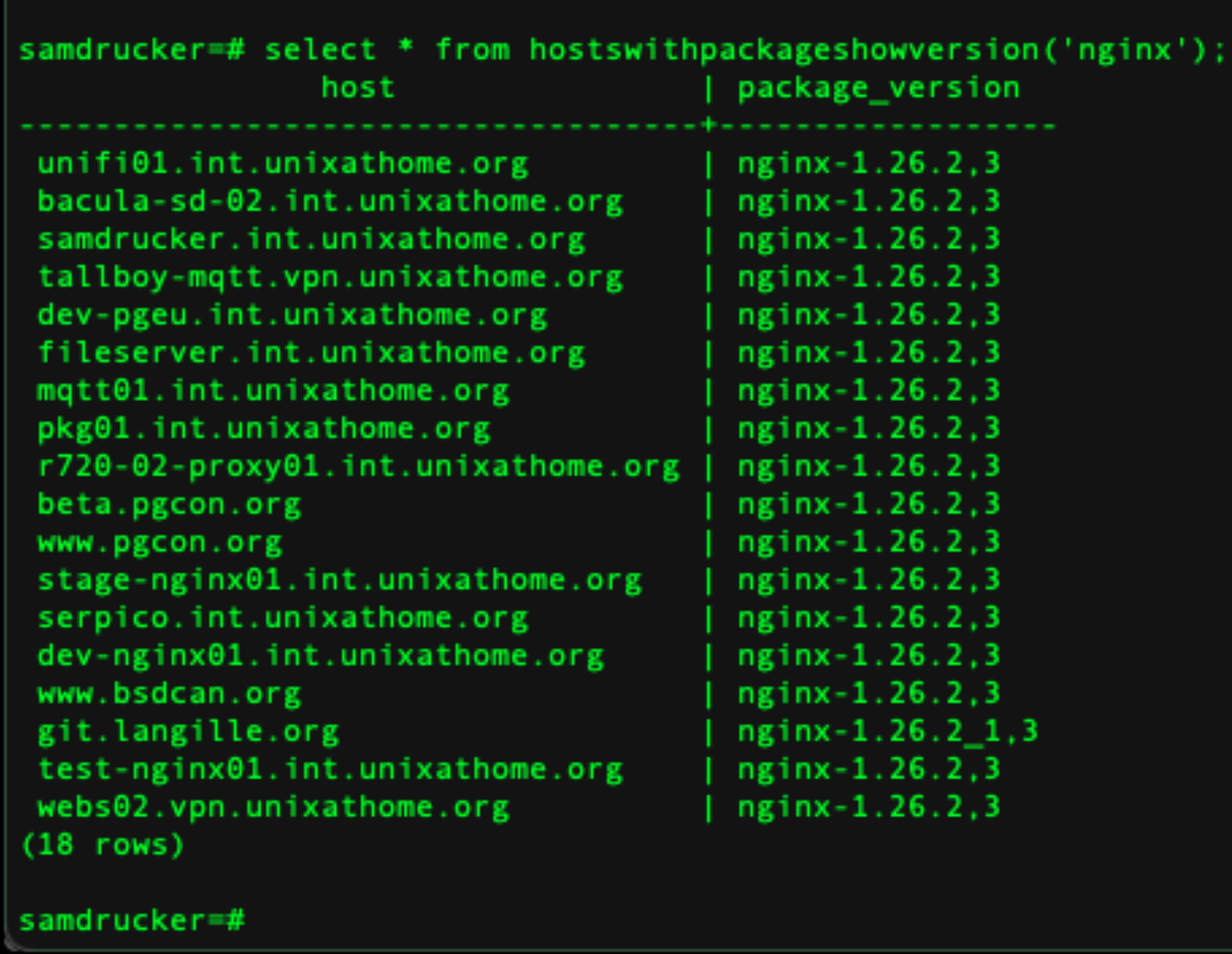
Postmaster for Petticoat Junction (1960’s American TV show)

- So many jails
- new vuln come in
- poudriere builds it
- Now, what needs to be upgraded?
- Sam Drucker knows!

- See `https://github.com/dlangille/SamDrucker`

SamDrucker

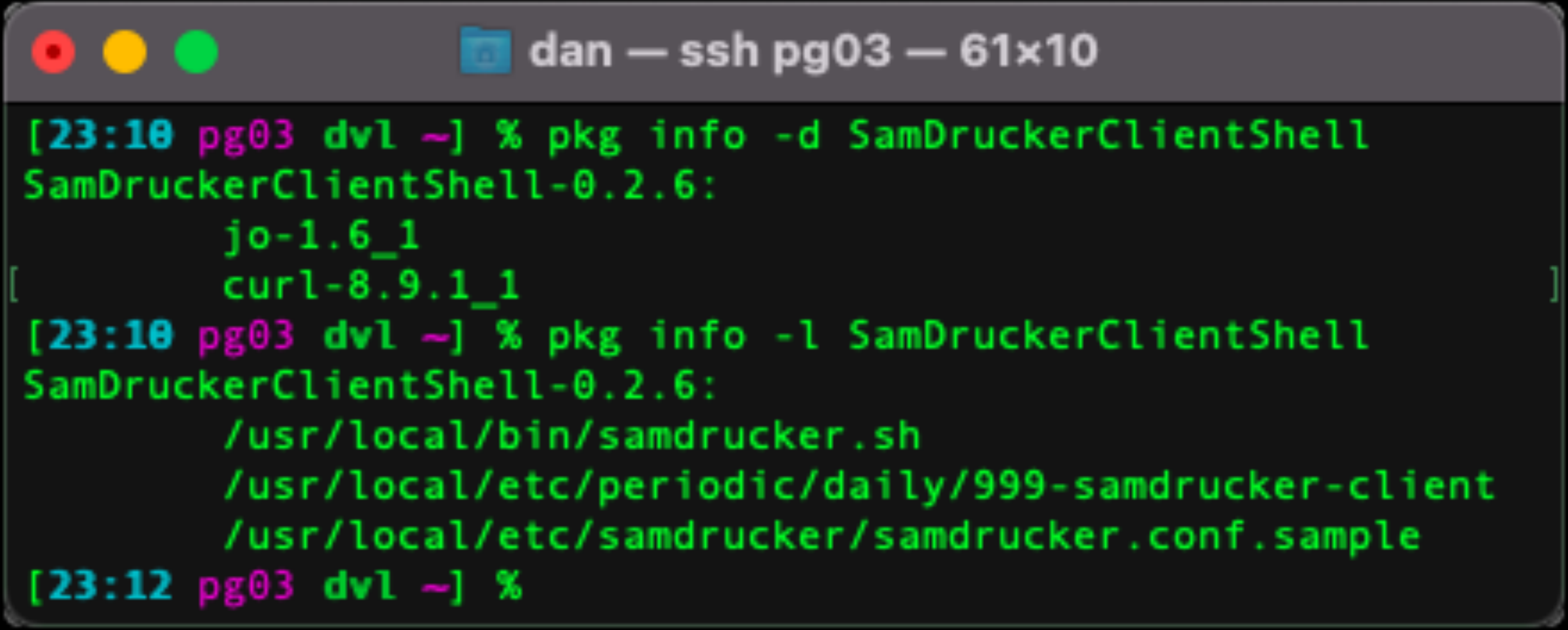
Who’s got nginx?



```
samdrucker=# select * from hostswithpackageshowversion('nginx');
                host                 | package_version
-------------------------------------+-------------------
 unifi01.int.unixathome.org          | nginx-1.26.2,3
 bacula-sd-02.int.unixathome.org     | nginx-1.26.2,3
 samdrucker.int.unixathome.org       | nginx-1.26.2,3
 tallboy-mqtt.vpn.unixathome.org     | nginx-1.26.2,3
 dev-pgeu.int.unixathome.org         | nginx-1.26.2,3
 fileserver.int.unixathome.org       | nginx-1.26.2,3
 mqtt01.int.unixathome.org           | nginx-1.26.2,3
 pkg01.int.unixathome.org            | nginx-1.26.2,3
 r720-02-proxy01.int.unixathome.org | nginx-1.26.2,3
 beta.pgcon.org                      | nginx-1.26.2,3
 www.pgcon.org                       | nginx-1.26.2,3
 stage-nginx01.int.unixathome.org    | nginx-1.26.2,3
 serpico.int.unixathome.org          | nginx-1.26.2,3
 dev-nginx01.int.unixathome.org      | nginx-1.26.2,3
 www.bsdcan.org                      | nginx-1.26.2,3
 git.langille.org                    | nginx-1.26.2_1,3
 test-nginx01.int.unixathome.org     | nginx-1.26.2,3
 webs02.vpn.unixathome.org           | nginx-1.26.2,3
(18 rows)

samdrucker=#
```

SamDrucker

Clients are lightweight



```
[23:10 pg03 dvl ~] % pkg info -d SamDruckerClientShell
SamDruckerClientShell-0.2.6:
        jo-1.6_1
        curl-8.9.1_1
[23:10 pg03 dvl ~] % pkg info -l SamDruckerClientShell
SamDruckerClientShell-0.2.6:
        /usr/local/bin/samdrucker.sh
        /usr/local/etc/periodic/daily/999-samdrucker-client
        /usr/local/etc/samdrucker/samdrucker.conf.sample
[23:12 pg03 dvl ~] %
```

gitea

“Private, Fast, Reliable”

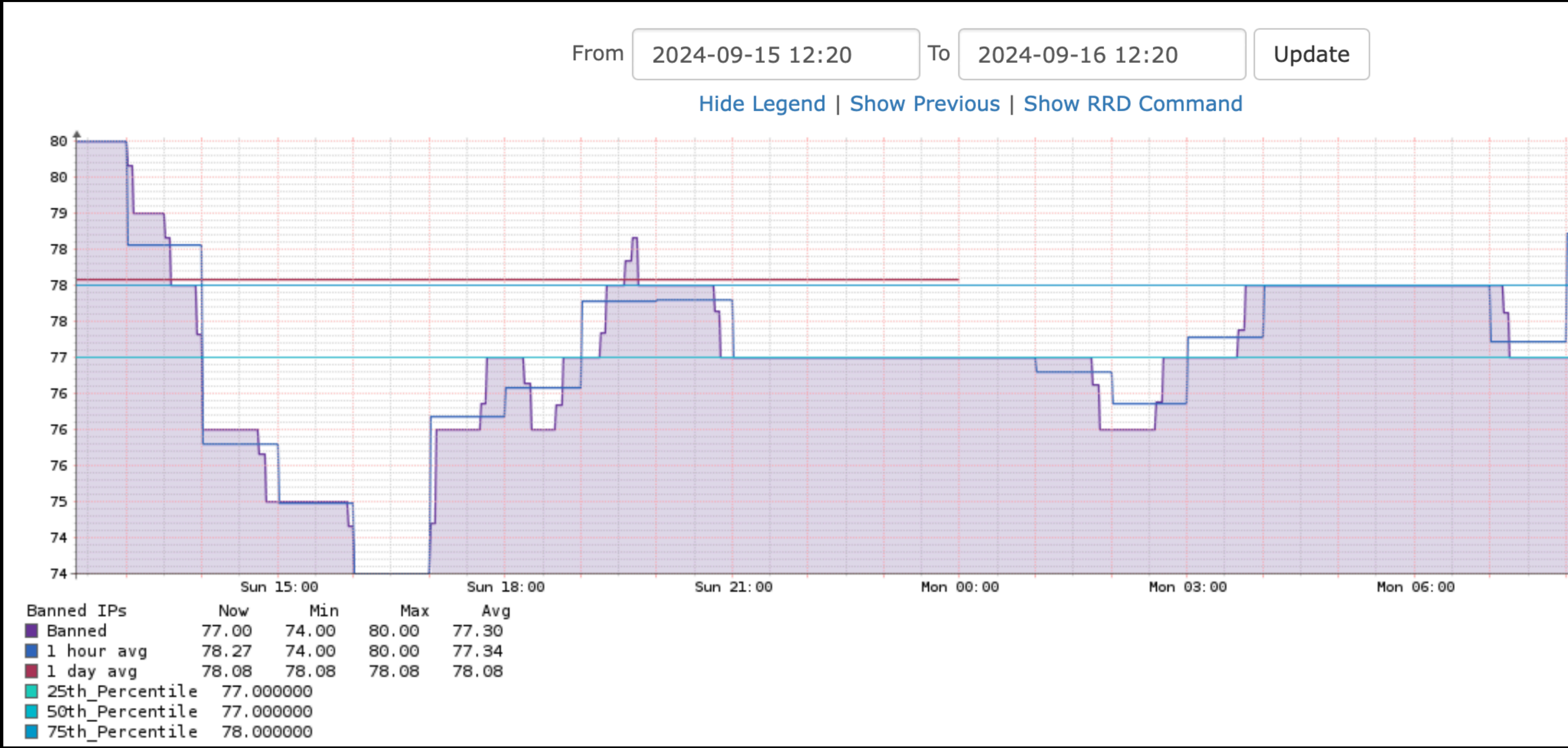
- I have all this code, where do I put it?
- `git.langille.org`
- runs `nginx`
- `/usr/local/sbin/gitea web`
- so very reliable for me
- gitlab seemed too complex, but had nice features
- Cannot recall other considerations

LibreNMS

I use it for metrics, others also use it for monitoring

- I’ve got all these jails, I really should monitor / measure them….
- Step one, create a jail
- Works well in a jail - nothing special required
- More graphs than you’ll ever need
- Monitoring for poudriere, zfs, disk io, CPU usage
- OSLV monitoring - resources on a jail by jail basis

LibreNMS



From 2024-09-15 12:20 To 2024-09-16 12:20 Update

Hide Legend | Show Previous | Show RRD Command

```
Banned IPs           Now      Min      Max      Avg
🟪 Banned            77.00    74.00    80.00    77.30
🟦 1 hour avg        78.27    74.00    80.00    77.34
🟥 1 day avg         78.08    78.08    78.08    78.08
🟩 25th_Percentile   77.000000
🟦 50th_Percentile   77.000000
🟦 75th_Percentile   78.000000
```

All these websites!

How do I get new certs for each one?

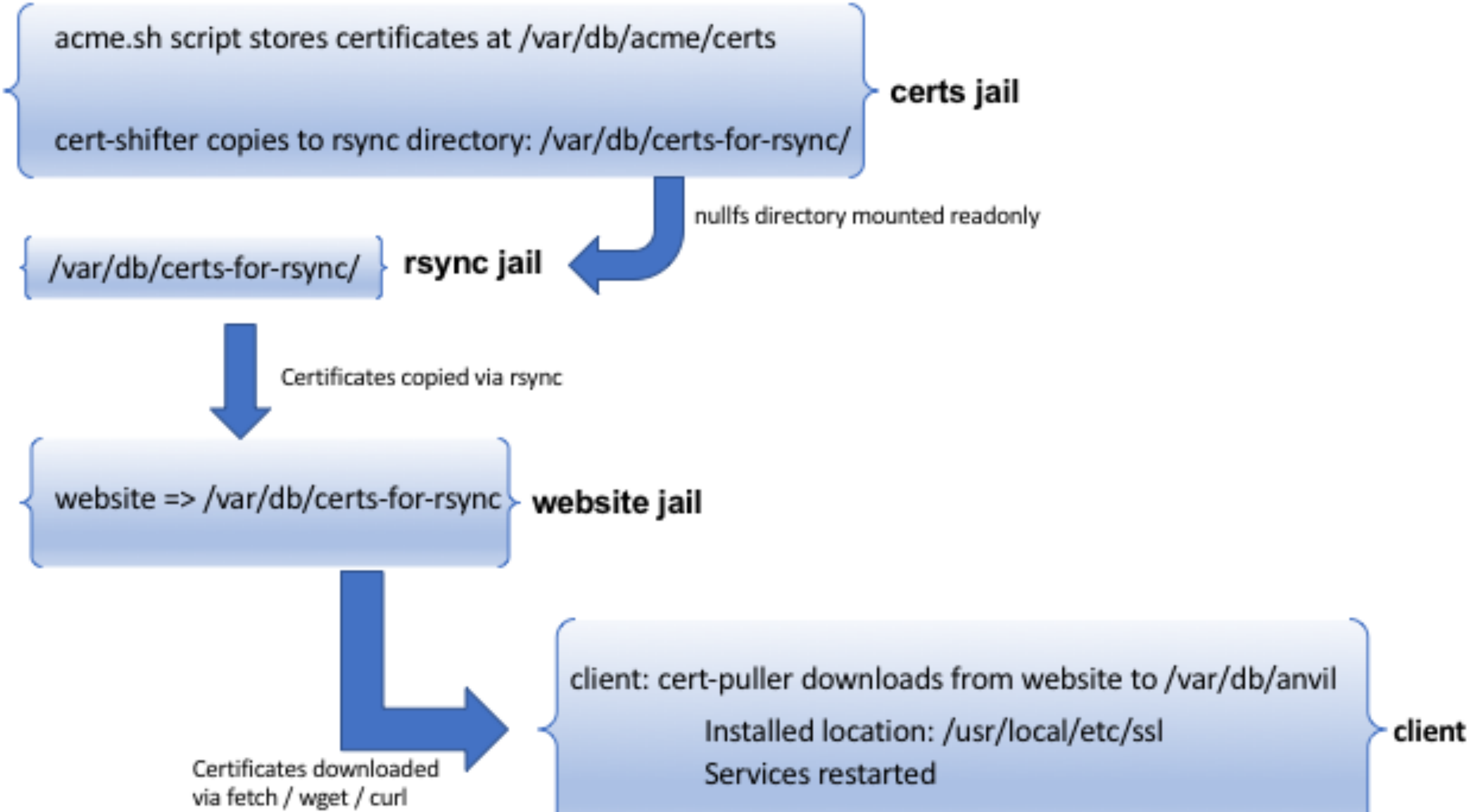
- Run a Let’s Encrypt Client on each website - no
- Run Let’s Encrypt on one central host - how do you distribute the new certs?
- selected acme.sh because Peter Wemm did that for FreeBSD.org
- certs created in central jail - no public access
- talks to local hidden dns master - no public access

- How to distribute those certs: `https://github.com/dlangille/anvil/`

Anvil

Named for Road Runner cartoons - anvils from Acme

- Must distribute certificate key manually
- Certs are public by nature - no worries uploading them to a website
- can be used by services (e.g. smtp) which do not have port 80/443 available
- Been using it since 2017 without regrets



acme.sh script stores certificates at /var/db/acme/certs

cert-shifter copies to rsync directory: /var/db/certs-for-rsync/

certs jail

nullfs directory mounted readonly

/var/db/certs-for-rsync/

rsync jail

Certificates copied via rsync

website => /var/db/certs-for-rsync

website jail

Certificates downloaded
via fetch / wget / curl

client: cert-puller downloads from website to /var/db/anvil

Installed location: /usr/local/etc/ssl

Services restarted

client

cert-puller

the required sudo permissions

```
[[17:46 git dvl ~] % cert-puller -s
 anvil   ALL=(ALL) NOPASSWD:/bin/cp -a /var/db/anvil/ca.cer /usr/local/etc/ssl/ca.cer.tmp
 anvil   ALL=(ALL) NOPASSWD:/bin/mv /usr/local/etc/ssl/ca.cer.tmp /usr/local/etc/ssl/ca.cer
 anvil   ALL=(ALL) NOPASSWD:/bin/cp -a /var/db/anvil/git.langille.org.cer /usr/local/etc/ssl/git.langille.org.cer.tmp
 anvil   ALL=(ALL) NOPASSWD:/bin/mv /usr/local/etc/ssl/git.langille.org.cer.tmp /usr/local/etc/ssl/git.langille.org.cer
 anvil   ALL=(ALL) NOPASSWD:/bin/cp -a /var/db/anvil/git.langille.org.fullchain.cer /usr/local/etc/ssl/git.langille.org.fullchain.cer.tmp
 anvil   ALL=(ALL) NOPASSWD:/bin/mv /usr/local/etc/ssl/git.langille.org.fullchain.cer.tmp /usr/local/etc/ssl/git.langille.org.fullchain.cer
 anvil   ALL=(ALL) NOPASSWD:/usr/sbin/service nginx restart
 [17:46 git dvl ~] %
```

```
% cert-puller -s
```

Let’s Encrypt

related blog posts

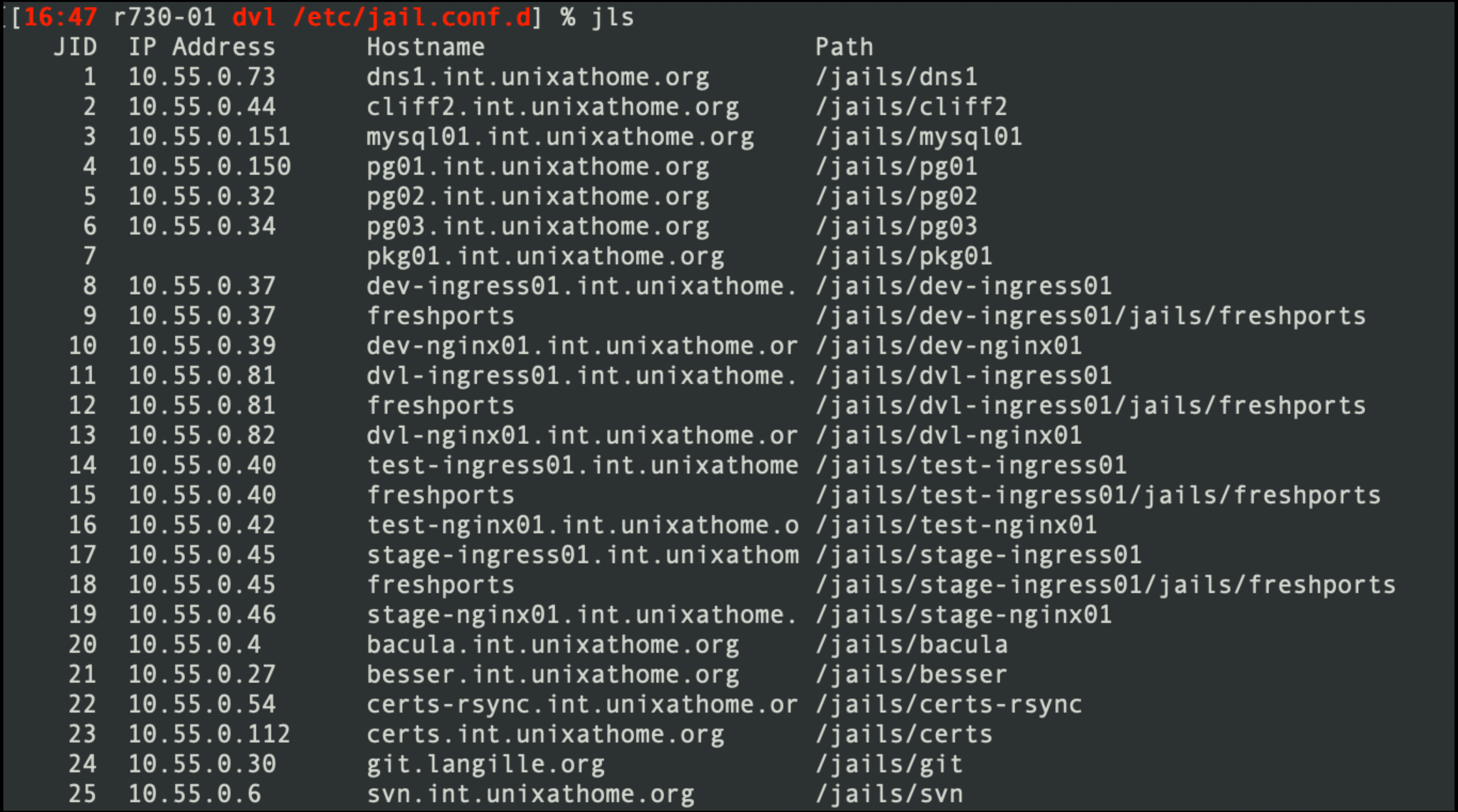
- https://dan.langille.org/2017/07/04/acme-sh-getting-free-ssl-certificates-installation-configuration-on-freebsd/
- https://dan.langille.org/2017/07/15/introducing-anvil-tools-for-distributing-ssl-certificates/
- https://dan.langille.org/2017/05/31/creating-a-txt-only-nsupdate-connection-for-lets-encrypt/

Copying an existing jail to try bind918

Try before you buy!

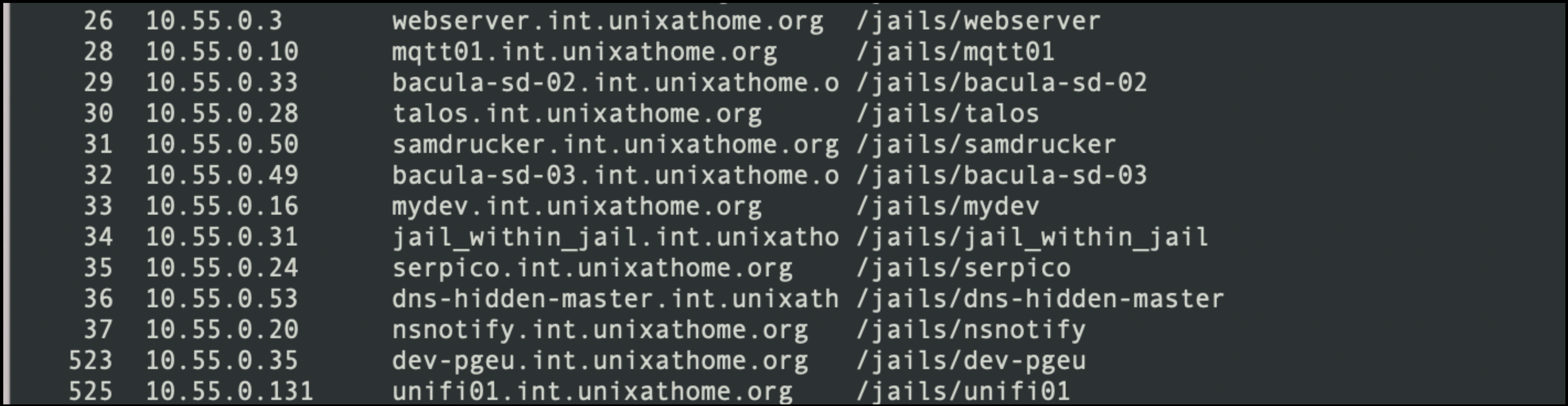
- syncoid for the copy
- New jail.conf entry - rename some stuff
- Very easy way to test something like that
- see `https://dan.langille.org/2024/02/29/copying-an-existing-jail-to-try-bind918/`

jails at home - 1/3 (host 1)



```
[16:47 r730-01 dvl /etc/jail.conf.d] % jls
   JID  IP Address      Hostname                      Path
     1  10.55.0.73      dns1.int.unixathome.org       /jails/dns1
     2  10.55.0.44      cliff2.int.unixathome.org     /jails/cliff2
     3  10.55.0.151     mysql01.int.unixathome.org    /jails/mysql01
     4  10.55.0.150     pg01.int.unixathome.org       /jails/pg01
     5  10.55.0.32      pg02.int.unixathome.org       /jails/pg02
     6  10.55.0.34      pg03.int.unixathome.org       /jails/pg03
     7                  pkg01.int.unixathome.org      /jails/pkg01
     8  10.55.0.37      dev-ingress01.int.unixathome. /jails/dev-ingress01
     9  10.55.0.37      freshports                    /jails/dev-ingress01/jails/freshports
    10  10.55.0.39      dev-nginx01.int.unixathome.or /jails/dev-nginx01
    11  10.55.0.81      dvl-ingress01.int.unixathome. /jails/dvl-ingress01
    12  10.55.0.81      freshports                    /jails/dvl-ingress01/jails/freshports
    13  10.55.0.82      dvl-nginx01.int.unixathome.or /jails/dvl-nginx01
    14  10.55.0.40      test-ingress01.int.unixathome /jails/test-ingress01
    15  10.55.0.40      freshports                    /jails/test-ingress01/jails/freshports
    16  10.55.0.42      test-nginx01.int.unixathome.o /jails/test-nginx01
    17  10.55.0.45      stage-ingress01.int.unixathom /jails/stage-ingress01
    18  10.55.0.45      freshports                    /jails/stage-ingress01/jails/freshports
    19  10.55.0.46      stage-nginx01.int.unixathome. /jails/stage-nginx01
    20  10.55.0.4       bacula.int.unixathome.org     /jails/bacula
    21  10.55.0.27      besser.int.unixathome.org     /jails/besser
    22  10.55.0.54      certs-rsync.int.unixathome.or /jails/certs-rsync
    23  10.55.0.112     certs.int.unixathome.org      /jails/certs
    24  10.55.0.30      git.langille.org              /jails/git
    25  10.55.0.6       svn.int.unixathome.org        /jails/svn
```

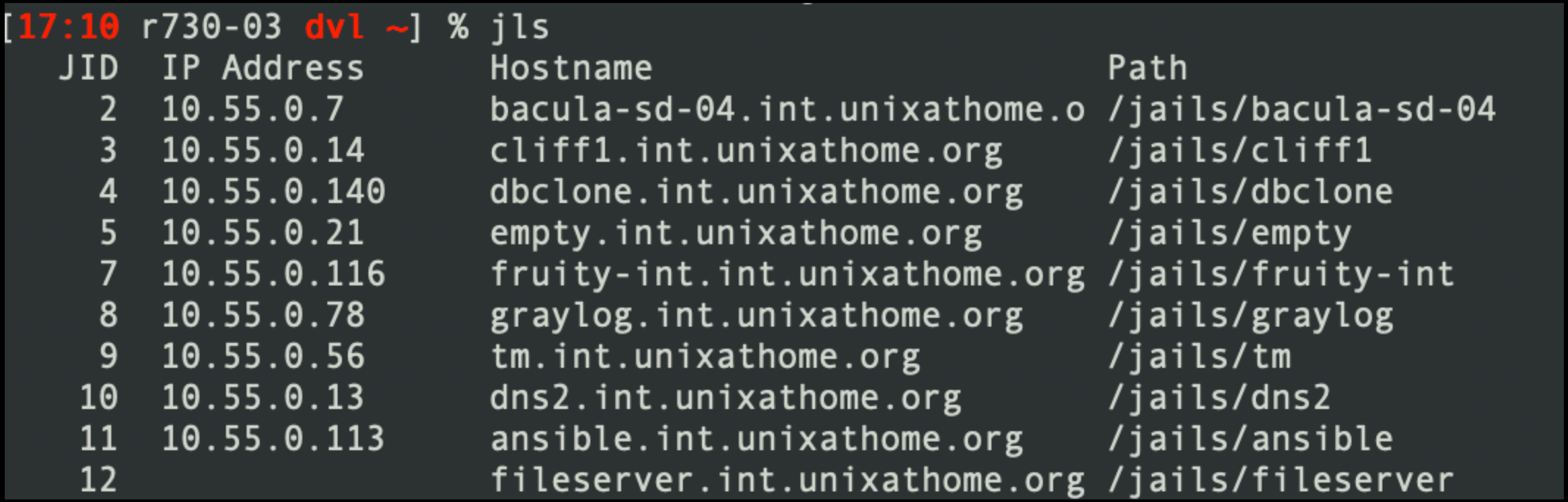
jails at home - 2/3 (still host 1)



```
  26  10.55.0.3       webserver.int.unixathome.org  /jails/webserver
  28  10.55.0.10      mqtt01.int.unixathome.org     /jails/mqtt01
  29  10.55.0.33      bacula-sd-02.int.unixathome.o /jails/bacula-sd-02
  30  10.55.0.28      talos.int.unixathome.org      /jails/talos
  31  10.55.0.50      samdrucker.int.unixathome.org /jails/samdrucker
  32  10.55.0.49      bacula-sd-03.int.unixathome.o /jails/bacula-sd-03
  33  10.55.0.16      mydev.int.unixathome.org      /jails/mydev
  34  10.55.0.31      jail_within_jail.int.unixatho /jails/jail_within_jail
  35  10.55.0.24      serpico.int.unixathome.org    /jails/serpico
  36  10.55.0.53      dns-hidden-master.int.unixath /jails/dns-hidden-master
  37  10.55.0.20      nsnotify.int.unixathome.org   /jails/nsnotify
 523  10.55.0.35      dev-pgeu.int.unixathome.org   /jails/dev-pgeu
 525  10.55.0.131     unifi01.int.unixathome.org    /jails/unifi01
```

see `https://dan.langille.org/2024/02/02/7825/`

jails at home - 3/3 (host 2)



```
[17:10 r730-03 dvl ~] % jls
   JID  IP Address      Hostname                      Path
     2  10.55.0.7       bacula-sd-04.int.unixathome.o /jails/bacula-sd-04
     3  10.55.0.14      cliff1.int.unixathome.org     /jails/cliff1
     4  10.55.0.140     dbclone.int.unixathome.org    /jails/dbclone
     5  10.55.0.21      empty.int.unixathome.org      /jails/empty
     7  10.55.0.116     fruity-int.int.unixathome.org /jails/fruity-int
     8  10.55.0.78      graylog.int.unixathome.org    /jails/graylog
     9  10.55.0.56      tm.int.unixathome.org         /jails/tm
    10  10.55.0.13      dns2.int.unixathome.org       /jails/dns2
    11  10.55.0.113     ansible.int.unixathome.org    /jails/ansible
    12                  fileserver.int.unixathome.org /jails/fileserver
```

see `https://dan.langille.org/2024/02/02/r730-03-4/`

Websites around the world!

one jail to back them up

- I want to backup my databases
- create read-only user: `rsyncer`
- Dumps the databases - in the remote jail
- Calls home to say they are ready
- That call invokes a rsync
- Home, in this case, is dbclone, a jail in my basement

Now that I have these backups…

I need to test them…

- dbclone has all the databases
- daily loads of that data

-
  ```
  for each host_backed_up
    rm -rf ${PGDATADIR}
    service postgresql initdb
    for each database from host
      pg_restore dbname < dbname.dump
    end for
  end for
  ```
- see `https://git.langille.org/dvl/database-backup-testing`

kernel: Limiting closed port RST response from x to y packets/sec

self-inflicted DDoS

- converted a host to use dma, not sendmail
- 7852 sendmail processes running
- 15,000 queued emails
- All jails were trying to contact the host on port 25, like they should
- The phone calls are coming from inside the house
- see `https://dan.langille.org/2024/08/07/kernel-limiting-closed-port-rst-response-from-x-to-y-packets-sec/`

pushover.net

Founded and run by a BSD person

- Poudriere build broke? UPS lost power? Water detected in basement? IP address at home changed?
- Lots of scripts, send a notice
- it notifies you - clients for phone, browser, etc
- cheap & reliable
- easier than sending an email
- Pushover uses notification system on the phone (encrypted)
- Better than SMS (not encrypted & can be intercepted)

Doing stupid things with FreeBSD jails

thank you

Dan Langille 2024.09.22 - EuroBSDCon Dublin