Adventures in High(ish) Availability
Peter Chubb | Principal Research Engineer
January 21, 2019
services

- DNS, DHCP/BOOTP, LDAP, NFS, TFTP, Postgres, kitty, web services, CI services (bamboo\textsuperscript{TM}), login, hg, git, machine-queue, bitbucket\textsuperscript{TM} \ldots
services

- DNS, DHCP/BOOTP, LDAP, NFS, TFTP, Postgres, kitty, web services, CI services (bamboo\textsuperscript{TM}), login, hg, git, machine-queue, bitbucket\textsuperscript{TM} \ldots
- around 40 desktops using DHCP, NFS and LDAP
services

- DNS, DHCP/BOOTP, LDAP, NFS, TFTP, Postgres, kitty, web services, CI services (bamboo™), login, hg, git, machine-queue, bitbucket™ …
- around 40 desktops using DHCP, NFS and LDAP
- around 30 dev boards and test machines using BOOTP, TFTP, and NFS
  - rebooting every few minutes; different mac address every reboot
The Situation

• Ancient server hardware (donated to us in 2000 or thereabouts)
• Only some services replicated (DNS, LDAP both master/slave)
• Growing group — downtime costs more
• Desire for planned downtime (kernel upgrades, hardware changes etc)
The Situation

- Ancient server hardware (donated to us in 2000 or thereabouts)
- Only some services replicated (DNS, LDAP both master/slave)
- Growing group — downtime costs more
- Desire for planned downtime (kernel upgrades, hardware changes etc)
- applied for Capex funding for new server
  - Huge corporate discount
The Situation

- Ancient server hardware (donated to us in 2000 or thereabouts)
- Only some services replicated (DNS, LDAP both master/slave)
- Growing group — downtime costs more
- Desire for planned downtime (kernel upgrades, hardware changes etc)
- applied for Capex funding for new server
  - Huge corporate discount
    → Buy Two!
High(ish) availability

99.99999999999999999999999999999999999999999999999999999999%
High(ish) availability

99.99999999999999999999999999999999999999999999999999999999%

99.99999999999999999999999999999999999999999999999999999999%
High(ish) availability

99.99999999999999999999999999999999999999999999999999999999%

A few minutes here and there don’t matter
High(ish) availability

99.99999999999999999999999999999999999999999999999999999999%

A few minutes here and there don’t matter
Manual failover for new kernel, replace network card etc. OK
Two Servers!

- 24 core
- 300G Ram
- 16Tb spinning Disk with 1.2Tb RAID-1 nVME cache
- 2x10Gb/s fibre, 8x1Gb/s copper

Replication and/or failover *possible*. 
Two Servers!

Hosts

Cellar Brewer

Containers

Stopped

Running

Cellar

DNS

NFS

tftp

ldap

web

login

Brewer

Containers

Stopped

Running

DNS

NFS

tftp

ldap

web

login

CSIRO Data61 Copyright © 2019 CC BY-SA Adventures in High(ish) Availability
Two Servers!

Cellar Brewer

DNS
NFS
tftp
ldap
web
login

Brewer

DNS
NFS
tftp
ldap
web
login

lsyncd
Two Servers!

Cellar Brewer

DNS
NFS
tftptftp
ldap
web
login

Brewer

DNS
NFS
tftptftp
ldap
web
login
Two Servers!
7.00am  Came into work; Turned coffee machine on; checked logwatch
7.00am  Came into work; Turned coffee machine on; checked logwatch
7:15am  Attempted failover: shutdown one host
Testing

7.00am  Came into work; Turned coffee machine on; checked logwatch
7:15am  Attempted failover: shutdown one host
7:40am  Looking good: services all transferred and running
Testing

7.00am  Came into work; Turned coffee machine on; checked logwatch
7:15am  Attempted failover: shutdown one host
7:40am  Looking good: services all transferred and running
7:45am  get coffee
7:50am Notice login xterms have frozen: can’t log back in. Attempt to get into host’s consoles — can’t do it as me; manage to remember root password. Very slow response.
Testing

7:50am Notice login xterms have frozen: can’t log back in. Attempt to get into host’s consoles — can’t do it as me; manage to remember root password. Very slow response.

8:00am get warning (to phone) that webservers are down
7:50am Notice login xterms have frozen: can’t log back in. Attempt to get into host’s consoles — can’t do it as me; manage to remember root password. Very slow response.

8:00am get warning (to phone) that webservers are down

8:10am On console, NFS server not responding; can’t connect to nfshomes: no DNS entry.
7:50am Notice login xterms have frozen: can’t log back in. Attempt to get into host’s consoles — can’t do it as me; manage to remember root password. Very slow response.

8:00am get warning (to phone) that webservers are down

8:10am On console, NFS server not responding; can’t connect to nfshomes: no DNS entry.

8:15am (people start arriving at work; can’t work: no local DNS)
7:50am Notice login xterms have frozen: can’t log back in. Attempt to get into host’s consoles — can’t do it as me; manage to remember root password. Very slow response.

8:00am get warning (to phone) that webservers are down

8:10am On console, NFS server not responding; can’t connect to nfshomes: no DNS entry.

8:15am (people start arriving at work; can’t work: no local DNS)

8:20am reboot original server; restart original services one at a time; fail back
7:50am Notice login xterms have frozen: can’t log back in. Attempt to get into host’s consoles — can’t do it as me; manage to remember root password. Very slow response.

8:00am get warning (to phone) that webservers are down

8:10am On console, NFS server not responding; can’t connect to nfshomes: no DNS entry.

8:15am (people start arriving at work; can’t work: no local DNS)

8:20am reboot original server; restart original services one at a time; fail back

11am Everything seems normal again; get another coffee
PROBLEMS

- DHCP can’t update names on slave server
PROBLEMS

- DHCP can’t update names on slave server
- DNS entries time out if master is down.
  - Timeouts are short to cope with devboard short lease lifetimes
  - Everything stops if DNS stops
PROBLEMS

- DHCP can’t update names on slave server
- DNS entries time out if master is down.
- NFS after failover fails
  - Handle based on inode number and File-System ID — inode numbers different
  - NFSv4 is stateful
PROBLEMS

- DHCP can’t update names on slave server
- DNS entries time out if master is down.
- NFS after failover fails
- Run out of watch slots for lsyncd
PROBLEMS

- DHCP can’t update names on slave server
- DNS entries time out if master is down.
- NFS after failover fails
- Run out of watch slots for lsyncd
- Postgres failover (sort-of) OK; fail-back difficult
Second attempt

- Stateless services as before
- Per-service solutions for the rest
LDAP

- Not hard to make openldap replicate master-master.
- Round-robin DNS allows load sharing
- SSSD on clients mean short outages don’t matter (much).
LDAP

- Not hard to make openldap replicate master-master.
- Round-robin DNS allows load sharing
- SSSD on clients mean short outages don’t matter (much).

Works!
DNS

- LDAP replication working ...
  - So use LDAP as backend.
    * `bind9-dyndb-ldap` already packaged for Debian
  - Works well with BIND 9.11
  - Multi-master DNS ‘tricky’, but seems to work.
  - Running in containers on both hosts as masters; watchdog ensures containers are running
DNS

- LDAP replication working ...
  - So use LDAP as backend.
    * bind9-dyndb-ldap already packaged for Debian
  - Works well with BIND 9.11
  - Multi-master DNS ‘tricky’, but seems to work.
  - Running in containers on both hosts as masters; watchdog ensures containers are running

Works!
DHCP

- Still have `bootp` clients — can’t use native replication
- Server runs in same container as one of the DNS servers, to allow name update
- watchdog in each DNS container starts DHCPD if it is not running on the DNS replica
- `/etc/dhcpd.conf` held in GIT, `git pull` on start.
DHCP

- Still have `bootp` clients — can’t use native replication
- Server runs in same container as one of the DNS servers, to allow name update
- watchdog in each DNS container starts DHCPD if it is not running on the DNS replica
- `/etc/dhcpd.conf` held in GIT, `git pull` on start.

Works
NFS

- DRBD for underlying FS
- NFSv4 state on one of the replicated volumes
NFS

1. Check switches are up. Abort if not
NFS

1. Check switches are up. Abort if not
2. Check if DRBD is up-to-date. Abort if not.
NFS

1. Check switches are up. Abort if not

2. Check if DRBD is up-to-date. Abort if not.

3. If remote is up, shut it down:
   - stop `nfs-kernel-server` and `rpcbind`
   - unmount exported volumes
   - delete the HA address
   - Check to see that the HA address is gone; if not, destroy the container.
1. Check switches are up. Abort if not
2. Check if DRBD is up-to-date. Abort if not.
3. If remote is up, shut it down:
4. switch the local DRBD to primary
NFS

1. Check switches are up. Abort if not.
2. Check if DRBD is up-to-date. Abort if not.
3. If remote is up, shut it down:
4. Switch the local DRBD to primary.
5. Start the local container if nec.
NFS

1. Check switches are up. Abort if not
2. Check if DRBD is up-to-date. Abort if not.
3. If remote is up, shut it down:
4. switch the local DRBD to primary
5. Start the local container if nec.
6. (in container) mount the filesystems, add the HA address, start `nfs-kernel-server`
NFS

Sort-of works.
NFS

Sort-of works.

Planned failovers work
NFS

Sort-of works.

Planned failovers work

Often see partial failover (DRBD switches rôles for some discs)
NFS

Sort-of works.

Planned failovers work

Often see partial failover (DRBD switches rôles for *some* discs)

Still investigating — packet loss?
NFS

Sort-of works.

Planned failovers work

Often see partial failover (DRBD switches rôles for *some* discs)

Still investigating — packet loss?

Also DAD races for IPv6.
Postgres

- Write-Ahead Log shipping for replication supported
Postgres

- Write-Ahead Log shipping for replication supported
  - With ‘just a bit’ of configuration
Postgres

- Write-Ahead Log shipping for replication supported
  - With ‘just a bit’ of configuration

- Easy to trigger failover
Postgres

- Write-Ahead Log shipping for replication supported
  - With ‘just a bit’ of configuration
- Easy to trigger failover

BUT
Postgres

- Write-Ahead Log shipping for replication supported
  - With ‘just a bit’ of configuration
- Easy to trigger failover

BUT

- Clients don’t know of failover
- No load balancing between active instances
- Fail-back is hard
Postgres

Investigating Patroni as a solution.
is_up() {
    ping -c 1 "$1" > /dev/null 2>&1
}

Remaining Issues
packet loss or congestion causes false down indications.

```bash
is_up() {
    for t in 5 10 30
do
        ping -c 1 "$1" > /dev/null 2>&1 && return 0
        sleep $t
    done
    ping -c1 "$1" > /dev/null 2>&1
}
```
Remaining Issues

Where possible check service not container:

```bash
is_up() {
  pg_isready "$1" > /dev/null 2>&1
}
```
$ ps axf

...  
26313  ?  Sl  0:00  /usr/lib/libvirt/libvirt_lxc --name nfshomes ...
26355  ?  Ss  0:19  \_ /sbin/init
26455  ?  Ss  1:49  \_ /lib/systemd/systemd-journald
26468  ?  Ss  0:00  \_ /usr/sbin/blkmapd

...
Orphan Zombies

```
$ ps axf
...
25234 ?   Ss 2:16 [init]
...
```

- Orphan Zombies
Orphan Zombies

$ ps axf
...
25234 ? Ss 2:16 [init]
...

- Orphan Zombies
  - Kill them all!
    * every 30 min
      /usr/local/bin/kill-orphans
But why not use …

- corosync and pacemaker
- piranha
- Etc
Available at: https://bitbucket.csiro.au/projects/TRUSTWORTHYSYSTEMS/repos/hiavail/browse