FREEBSD BASED HIGH DENSITY FILERS

Baptiste Daroussin
<bapt@gandi.net>
<bapt@FreeBSD.org>

BSDCan 2016

GANDI.NET

DOMAIN NAMES

SSL

SIMPLE HOSTING

SERVER

REFRESHING THE FILERS

- Nexenta based since 2007
- Difficulty to provide non attended setup
- Kernel patches for multipath on new disks
- Python stuck to an old (and buggy) version
- Very long boot time
  - OS
  - zpool import
  - iSCSI export

STUDY

STUDY

REQUIREMENTS

- ZFS
- Ability to server 1000 NFS and 900 iSCSI
- Possible extension to 1500 NFS and 1000 iSCSI
- Support NFSv4 with delegation
- Powerfull debugging tools (in particular dtrace)
- Support accessing JBOD with multipath
- OpenSource with an active community
- Ability to easily upstream patches
- Ability to run containers

STUDY

CANDIDATES

- Illumos family:
  - OpenIndiana
  - OmniOS
  - SmartOS
  - Newer Nexenta
- FreeBSD
- Linux (with ZoL)

LINUX

Rejected:

- ZoL cannot be upstreamed due to license incompatiblities
- Lots of regressions due to not being part of the upstream kernel

ILLUMOS

ILLUMOS

NEXENTA

Rejected:

- Community version limited to 18TB
- Upstreaming not easy

ILLUMOS

OPENINDIANA

Rejected:

- Small community
- Fragile build system
- Old python (2.6)

ILLUMOS

SMARTOS

Rejected:

- Global zone hard to customize
- no iSCSI/NFS management delegation
- Not design to make filers

ILLUMOS

- Exporting lots of iSCSI targets still long: more than 5 minutes
- Kernel still has to be patched for new disk manufacturers

FREEBSD

FREEBSD

THE GOOD

- Strong reputation on storage area
- Support modern ZFS and dtrace
- ctld(4)
- Very fast iSCSI export: few seconds
- good NFSv4 support
- mdb -> sysctl/kgdb
- Fast zpool import (tips: disable trim support)

FREEBSD

THE BAD

- Bad support for diskless netbooting
- Slow to boot on large MFSROOT
- No multiboot support == no proper iPXE support

DESIGN: DISKLESS

- Unattended setup via puppet
- Upgradability: just reboot
- Easy backtracking: just reboot
- Free from admin heroes
- Easy migration from Nexenta
- Safe migration from Nexenta

DESIGN: BOOTING SEQUENCE

EARLY BOOT

1. DHCP request
2. tftp get pxeboot
3. tftp get /boot/ configs
4. tftp get kernel, modules, miniroot

DESIGN: BOOTING SEQUENCE

BOOT MINIROOT

1. run custom rc
2. create a ramdisk
3. http get filer.txz config.txz puppet-<hostname>.txz
4. extract into ramdisk
5. reroot on ramdisk

DESIGN: BOOTING SEQUENCE

FINAL BOOT

1. zpool import
2. puppet run
3. starts Gandi's middleware
4. ready to serve

CONTRIBUTIONS

CONTRIBUTIONS

PY-LIBZFS (FREENAS)

- Implement zfs clone support
- Implement zfs promote support
- Implement support for properties (including custom)
- Implement volume support
- Bug fixing

CONTRIBUTIONS

MPSUTIL(8)/MPRUTIL(8) (NETFLIX)

- Finish integration with FreeBSD build system
- implement flashing firmwares/bios

CONTRIBUTIONS

PLAYING THE GUINEA PIG

- reroot (by trasz@)
- smarter mount root wait (by trasz@)

SESUTIL(8)

MANAGING SCSI ENCLOSURE SERVICES

- blink locate led (only disks)
- blink fault led (only disks)
- show the detailed map of an enclosure
- easy to use:

```
$ sesutil locate da3 on
```

```
$ sesutil locale all off
```

SESUTIL(8)

VENDOR TOOLS

- Lots of noise in the logs
- 2 differents tools for SAS2 and SAS3
- Unfriendly UI

SESUTIL(8)

SG_SES (SG3_UTILS)

- Unfriendly UI
- mapping disks complex

REWORK PXEBOOT WITH TFTP SUPPORT

Add support for root-path DHCP option to act like pxeboot with NFS support

RUNNING HEAD

stable most of the time

needed features only available there

easier to upstream patches

find (and fix) as early as possible bugs

Gandi's workload very well identified

TEST LAB

- Drived by Zopkio
- Simulating broken disks using gnop(8)
- Simulating bad network access using ipfw(8) + dummynet(4)
- Simulating crash and reboot under high load from consumers
- Profile based test lab

FUTUR PLANS

FUTUR PLANS

IMPROVE SESUTIL(8)

- libxoify(?)
- Add microcode update support
- Extend locate to support other devices

FUTUR PLANS

IMPROVE ZFS(8)

- Improve zpool import speed
- Tuning tunable like arc_max into safe read/write tunables
- Maybe new features to improve reliability

FUTUR PLANS

IMPROVE FOR IPXE SUPPORT

- Implement a FreeBSD specific loader
  or
- Turn the FreeBSD kernel into multiboot

FUTUR PLANS

IMPROVE CTL(4)

- Convert the number of ports and lun per ports into sysctl
- Turn ctl(4) into using libucl (too late)

FUTUR PLANS

STORAGE RELATED TOOLING

- Implement port some dtrace scripts from Illumos
- Improve geom_multipath algorithm to better match ZFS requirements

THANKS!

Questions?

BSDCan 2016