Morgan Stanley

OpenAFS on Solaris 11 x86

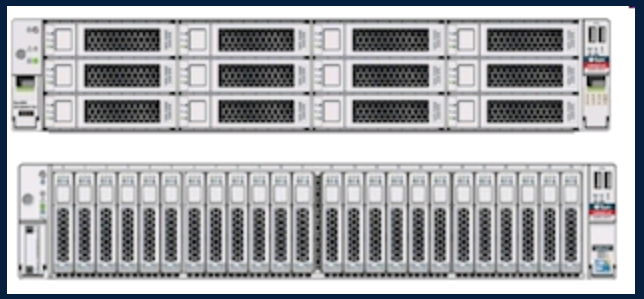
Robert Milkowski, VP
Unix Engineering

AFS on Solaris

- Big $$ savings
  - ZFS compression
  - Local disks as storage instead of external disk arrays
    - Lower TCA and TCO
      - Less power, cooling, rack space, etc.
- Better reliability
  - ZFS checksumming and self-healing
  - FMA + PSH
- Better observability
  - DTrace

AFS RO

- Oracle/Sun X4-2L
- 2U
- 2x Intel Xeon E5-2600 v2 Ivy
- Up-to 512GB RAM (16x DIMM)
- 12x 3.5” disks + 2x 2.5” (rear)
- 24x 2.5” disks + 2x 2.5” (rear)
- 4x On-Board 10GbE
- 6x PCIe 3.0
- SAS/SATA JBOD mode



AFS RW

- 2-node VCS cluster stretched across data centres
- ZFS compression enabled (less SAN usage)
- ZFS mirroring across disk arrays in different data centres
  - Over 3x less data to transfer over FC links
  - Fewer clusters
  - No fsck
  - Checksumming and self healing
- Backups using ZFS snapshots and ZFS replication

AFS RW Backups with ZFS

- Create a consistent snapshot of an AFS partition
  - vos freeze –server localhost –part vicepa –timeout 60
  - zfs snapshot pool/vicepa@2014-01-15-10:58
  - vos unfreeze –server localhost –part vicepa
- Send the snapshot to a remote host
  - ZFS send pool/vicepa@2014-01-15-10:58 | … | zfs receive …
  - Incremental zfs send –i snap1 snap2 | … | zfs receive …
- voldump(8) to restore a volume from snapshot or remote server
- Ideally Fileserver should be able to serve volumes present in snapshots

Fault Management Architecture (FMA)

- Automated diagnosis of faulty software and hardware
  - Isolate HW problems
  - Restart affected software
  - Centralized log repository for all reported faults
    - Identifies physical components (Topology Framework)
    - Keeps very detailed information for events
  - Alerting
- Predictive Self Healing
  - Proactively black list a memory page or a memory DIMM
  - Proactively attach hot-spare if a disk is generating too many errors

Server-Monitor

- Daemon to monitor basic OS/HW health
- Sends email and/or Netcool alerts
- Runs under SMF as svc:/ms/monitoring/server-monitor
- Consumes FMA alerts
  - Along with the topology framework can identify physical components, their part numbers and serial numbers
  - In case of local disk drives it can provide physical location
  - All information required to replace FRU is included in the alert
- Runs additional checks on networking, ZFS, etc.

ZFS SCRUB

- ZFS POOL SCRUB
  - Scan all data and meta-data and validate checksums
  - Selfheal corrupted blocks
  - Generate FMA alert
  - Generate alert to Ops
- AFS servers scrub all ZFS pools on weekly basis
  - This also stress-tests disks
    - Already detected some failing and bad behaving disks

Data Corruptions

- So far experienced four cases of a disk returning bad data in local disk set-up
- In three cases a disk first reported a couple of read errors, and eventually returned bad data, one of them died a moment later
- In the fourth case a disk returned one corrupted block
- In all cases ZFS detected it, obtained the good copy from other mirror, returned good data to applications and fixed corruption
- Disk replacement was fully automatic with no innervation required at OS level

ZFS/FMA – Corruption Handling

- During a weekly zfs pool scrub
  - A disk reported a couple of read errors
  - Multiple checksum errors were detected on the disk as well
  - Affected blocks were automatically corrected by ZFS
  - As number of cksum errors was high FMA activated a hot-spare disk which formed a 3-way mirror
  - We decided to replace the suspicious disk
    - The affected disk was pulled out while AFS was running
    - A replacement was put back in
      - ZFS automatically resilvered the disk
      - The hot-spare was automatically released

Comparing IPS Packages

- AFS binaries delivered as IPS packages
- Bosserver started by SMF
- Compare two IPS manifests from a repository

```
# pkg contents -mr pkg://ms/ms/afs/server-dmz@1.4.11.2,5.11-0:20130613T132308Z >/tmp/m1
# pkg contents -mr pkg://ms/ms/afs/server-dmz@1.4.11.2,5.11-0:20130614T104218Z >/tmp/m2
# pkgdiff /tmp/m1 /tmp/m2
set name=pkg.fmri
  - value=pkg://ms/ms/afs/server-dmz@1.4.11.2,5.11-0:20130613T132308Z
  + value=pkg://ms/ms/afs/server-dmz@1.4.11.2,5.11-0:20130614T104218Z
file path=lib/svc/manifest/ms/ms-afs-server.xml group=sys mode=0444 owner=root
restart_fmri=svc:/system/manifest-import:default
  - ac0987015530cb07219abb73d89e18f3508a2a05
  + db32b7b2f8f7c7d7deb682a53fc380d445656c1b
  - chash=10a250e102125db83bc716be31417189aad2fe30
  + chash=3da0c87684de6e91af4fe19a1c372edf7774ff90
  - pkg.csize=554
  + pkg.csize=573
  - pkg.size=1498
  + pkg.size=1646
```

IPS: pkg verify

- Validate the installation of a package

```
# pkg verify ms/afs/server
PACKAGE                                                                 STATUS
pkg://ms/ms/afs/server                                                   ERROR
        link: usr/afs/wbin/rvosd
                Target: '/ms/dist/afs/PROJ/rvos/2.4/sbin/rvosd' should be
                        '/ms/dist/afs/PROJ/rvos/prod/sbin/rvosd'
```

- Fix the broken package

```
# pkg fix ms/afs/server
Verifying: pkg://ms/ms/afs/server                              ERROR
        link: usr/afs/wbin/rvosd
                Target: '/ms/dist/afs/PROJ/rvos/2.4/sbin/rvosd' should be
                        '/ms/dist/afs/PROJ/rvos/prod/sbin/rvosd'
Created ZFS snapshot: 2013-12-27-11:42:48
Repairing: pkg://ms/ms/afs/server
Creating Plan (Evaluating mediators): -

PHASE                                          ITEMS
Updating modified actions                        1/1
Updating image state                            Done
Creating fast lookup database                   Done
# pkg verify ms/afs/server
#
```

OS Updates

- Automatic and regular OS updates
- Solaris Boot Environments (BE)
  - ZFS clone of root-fs
  - GRUB menu entry added
  - Fast reboot – bypasses BIOS/POST (2-10 minutes quicker reboots)
- We force all package changes to be performed on a new BE
  - If some package installs/updates fails we do not activate the new BE

```
$ beadm list
BE                 Active Mountpoint Space  Policy Created
--                 ------ ---------- -----  ------ -------
after-postinstall  -      -          46.0K  static 2013-12-13 11:40
aquilon-11.1.12.5.0 NR     /          3.73G  static 2013-12-13 11:40
before-postinstall -      -          345.0K static 2013-12-13 11:33
solaris            -      -          4.45M  static 2013-12-13 11:15
```

Fast Reboot

- Reload OS without going thru POST/BIOS
  - Skips BIOS, PCI card firmware initialization, PXE initialization, boot loader, etc.
  - All drivers need to support quiesce() method which must succeed before reboot
    - Tested to work fine on: IBM, HP and Oracle servers
  - Similar to kexec on Linux
  - Saves 2-10+ minutes of reboot time, depending on HW
  - Works across different OS/kernel updates and across BEs
  - Enabled by default

VFS stats

- VFS statistics for AFS client - gerrit 10679
  - fsstat(1M)

```
$ fsstat /ms 1
 new  name   name  attr  attr lookup rddir  read read  write write
 file remov  chng   get   set    ops   ops   ops bytes   ops bytes
    0     9     0  747K     0  10.4M 71.9K 1.87M 7.06G     0     0 /ms
    0     0     0   158     0  2.29K    10   100  264K     0     0 /ms
    0     0     0   157     0  1.92K     2    92  262K     0     0 /ms
    0     0     0    55     0    610     0 2.01K 7.91M     0     0 /ms
    0     0     0   122     0  1.63K     0   659 2.06M     0     0 /ms
```

  - DTrace fsinfo::: provider

```
$ dtrace -q -n fsinfo:::'/args[0]->fi_fs == “afs"/{printf("%Y %s[%d] %s %s\n", walltimestamp, \
                        execname, pid, probename, args[0]->fi_pathname);}'

2014 Jan 16 16:49:07 ifstat[964] open /ms/dist/perl5/PROJ/core/5.8.8-2/.exec/ia32.sunos.5.10/bin/perl
2014 Jan 16 16:49:07 ifstat[964] addmap /ms/dist/perl5/PROJ/core/5.8.8-2/.exec/ia32.sunos.5.10/bin/perl
2014 Jan 16 16:49:07 ifstat[964] addmap /ms/dist/perl5/PROJ/core/5.8.8-2/.exec/ia32.sunos.5.10/lib/perl5/auto/Time/HiRes/HiRes.so
2014 Jan 16 16:49:07 tcpstat[1484] getpage /ms/dist/perl5/PROJ/core/5.8.8-2/.exec/ia32.sunos.5.10/bin/perl
```

  - iostat(1M) in the future?

mkdir() Performance

- During ‘make install’ to AFS some mkdir() taking 3s on Solaris, but not on Linux
- This is due to throttling in AFS file server for too many errors (EEXIST in this case)
- Linux has an optimization on VFS layer, so it won’t call file system specific callback if dnode already exists
- Solaris didn’t have the optimization – fixed in Solaris 11 SRU17 (and Solaris 11.2)
- AFS client could optimize for this and other conditions as well

AFS and Solaris Privileges

- Remove privileges which are not required
  - For example, most AFS daemons (all?) do not require PRIV_PROC_FORK nor PRIV_PROC_EXEC
  - Privilege sets can either be defined outside of AFS (no code changes required) or AFS daemons can be privilege aware
- Extended Policies
  - {file_dac_read}:/usr/afs/etc/*
  - {net_privaddr}:7001/udp
  - …

Solaris Zones

- Multiple AFS cells on the same hardware
  - Useful if a different AFS content needs to be provided to different clients
  - Much smaller overhead compared to full hypervisors
- Rapid AFS cell provisioning for DEV/QA
- Increased security
  - Isolated containers containers
  - Immutable zones

ZFS Tuning for AFS

- atime=off
- recordsize=1MB
- Compression=lzjb or gzip
- zfs:zfs_nocacheflush=1 when using disk arrays with HW RAID
- Increase DNLC size on Solaris
- SSD read cache – might be useful, so far 256GB RAM per server is ok for us
- SSD write cache – not needed on AFS 1.6+ (all writes async)
- Multiple vicep partitions in a ZFS pool (AFS scalability)

Morgan Stanley

Questions?