SOLARIS @STEP

DOAG 2013 Nürnberg
ZFS
News in Oracle Solaris 11

Martin Muschkiet, as-systeme

Themen

- Filesystemlayout und Bootenvironments in Solaris 11 dank ZFS schnell, flexibel und effizient
- Zones on Shared Storage (ZOSS) - portable Zonen in jeweils eigenem zpool
- ZFS Verschlüsselung

Dateisystem-Bootenvironment-Paketverwaltung

- Bei Solaris11 ein Team
  - :: Nur noch ZFS als Root-FS; gilt auch für non-globale Zonen
- FS-Layout
  - :: mit separaten */var*, */export*, */export/home* Dateisystemen
  - :: konfigurierbar bei der Installation (AI) per Manifest
  - :: Layout identisch für non-globale Zonen

NAME	USED	AVAIL	REFER	MOUNTPOINT
rpool	16,4G	3,21G	4,59M	/rpool
rpool/ROOT	3,86G	3,21G	31K	legacy
rpool/ROOT/sol11-u1	3,86G	3,21G	1,91G	/
rpool/ROOT/sol11-u1/var	1009M	3,21G	899M	/var
rpool/VARSHARE	81K	3,21G	81K	/var/share
rpool/export	10,3G	3,21G	326M	/export
rpool/export/home	119K	3,21G	32K	/export/home

Bootenvironment (BE)

- Ein BE ist eine bootfähige Instanz eines Solaris 11 Images
- Basiert auf ZFS Snapshots und Clones
- Mehrere BEs pro System möglich
- In BEs können unterschiedliche Softwareversionen installiert sein
- geringes Risiko bei OS-Upgrade Produktiv-BE wird nicht verändert

- Administration
  - CLI *beadm*
  - GUI *packagemanager*
- Auswahl beim Boot
  - X86: GRUB-menu
  - Sparc: im OBP *ok boot -L*

Unterschiede zu Solaris 10 BEs

- Solaris 11 synchronisiert Daten zwischen BEs nicht:
  z.B. Logdateien, Mail, User-Accounts..
  - Solaris 10 LiveUpgrade verwendet dafür */etc/lu/synclist*
- Ab 11.1 neues Dateisystem *rpool/VARSHARE*, Mountpunkt */var/share*
  - in allen BEs gemountet
  - erlaubt den Zugriff auf
    Mail-
    Audit-
    Coredateien
    Crashdumps
    von allen BEs

Bootenvironments (BEs)

- Der Administrator kann Softwarepakete in
  - das aktive BE
  - das aktive BE, nachdem ein Clone des aktuellen BE erstellt wurde (Bsp. unten)
  - ein inaktives, gemountetes BE
  - in ein neues BE installieren
  - Paketaktionen können "von sich aus" neue BEs verlangen

```
# pkg install --require-backup-be --backup-be-name  vorher  apptrace
        Packages to install:                1
        Create boot environment:            No
        Create backup boot environment:     Yes
...
```

BEs und non-globale Zonen (NGZ)

- für jedes BE das in der globalen Zone neu erstellt wird, wird für jede NGZ ebenfalls ein neues BE erstellt
- Der Administrator der NGZ kann weitere, eigene BEs erstellen
- Das BE der NGZ muss zum BE der globalen Zone passen
  - sonst wird ist das BE der NGZ als nicht Bootfähig angezeigt "!R"
  - Zugehörigkeit wird über eine ZFS User-property festgestellt
    Root-FS NGZ: *org.opensolaris.libbe:parentbe*

```
# zlogin zone1 beadm list
BE         Active  Mountpoint  Space    Policy  Created
--         ------  ----------  -----    ------  -------
solaris    NR      /           442.15M  static  2013-03-25  04:28
solaris-1  !R      -           3.0K     static  2013-05-27  13:43
```

Warum Zonen migrieren?

- um den virtuellen Server auf eine andere physische Maschine zu bewegen, um das Ausgangssystem herunter zu fahren oder zu warten
- die Zone ist dem Ausgangssystem "über den Kopf gewachsen"
- das Ausgangssystem ist ausgefallen
- Lifecycle Management:
  - auf Laptop entwickeln
  - auf Testsystem migrieren
  - auf das Produktivsystem übertragen

Zonen auf Shared Storage (ZoSS)

- Solaris 11.0
  - eigenes ZFS-Dateisystem als *zoneroot* für jede NGZ
  - Zpool besteht aus lokalen oder SAN Platten
- Solaris 11.1 zusätzlich ZoSS
  - einfache und sichere Migration von NGZ, die auf iSCSI oder FC installiert wurden
  - ein eigener Zpool für jede NGZ
  - Beim Abhängen der NGZ *zoneadm -z <zone> detach* wird der Zpool exportiert

Konfiguration

- Neue zonecfg Resource *rootzpool*
  - :: Legt einen dedizierten zpool bei der Installation der NGZ an
  - :: Führt zu Ex/Im-port des Pools bei zone *detach/attach*
  - :: *storage* Property gibt das Backend an
    - *dev* lokale Pfad URI
    - *lu* Fibre Channel und SAS
    - *iscsi* iSCSI
  - :: Das Hinzufügen von zwei *storage* Resourcen führt zu einem gespiegelten Pool

```
root@Bnode:~# zonecfg -z zoss
zonecfg:zoss> add rootzpool
zonecfg:zoss:rootzpool> add storage iscsi://suri1
zonecfg:zoss:rootzpool> add storage iscsi://suri2
zonecfg:zoss:rootzpool> end
```

Konfiguration

- Das *suriadm* Kommando kann helfen, den oft recht komplexen Pfad zu konstruieren.
  Bsp iSCSI-Volume vom Target s11-serv1 bereits vom OS erkannt

```
root@Bnode:~# suriadm lookup-uri -t iscsi \
/dev/dsk/c0t600144F09F3D4000000051A3C32D0001d0s0
iscsi://s11-serv1.mydomain.com/luname.naa.600144f09f3d4000000051a3c32d0001
```

Beispiel:

- Storage Server s11-serv1, gibt ein zvol per iSCSI frei
- Erstes System Anode
- Zweites System Bnode
- Zone zoss
- Ablauf:
  - iSCSI einrichten
  - Konfiguration der Zonen auf beiden Systemen
  - Installation auf einem der Systeme
  - Detach am Ausgangssystem
  - Attach am anderen System

- Konfiguration der NGZ auf Bnode, kopieren der Konfiguration

```
root@Bnode:~# zonecfg -z zoss
Use 'create' to begin configuring a new zone.
zonecfg:zoss> create
create: Using system default template 'SYSdefault'
zonecfg:zoss> set zonepath=/zones/zoss
zonecfg:zoss> add rootzpool
zonecfg:zoss:rootzpool> add storage iscsi://s11-serv1.mydomain.com/luname.naa
.600144f09f3d4000000051a3c32d0001
zonecfg:zoss:rootzpool> end
zonecfg:zoss> exit
root@Bnode:~# zonecfg -z zoss export  -f zoss.cfg
root@Bnode:~# scp  zoss.cfg  Anode:
root@Bnode:~# ssh root@Anode
Password:
root@Anode:~# zonecfg -z zoss -f zoss.cfg
```

- Installation auf einem node, Boot, Test
  - :: iSCSI "Platte" wurde vorab nicht partitioniert, kein Pool darauf erstellt

```
root@Bnode:~# zoneadm -z zoss install -c /root/sc-zoss.xml
zoneadm: zone 'zoss': zone zpool 'zoss_rpool' has been created, but failed to mount
at zonepath '/zones/zoss'
Configured zone storage resource(s) from:
    iscsi://s11-serv1.mydomain.com/luname.naa.600144f09f3d4000000051a3c32d0
001
Created zone zpool: zoss_rpool
Installation: Succeeded
<Ausgabe verkürzt>
        Done: Installation completed in 86,290 seconds.
root@Bnode:~#   zoneadm -z zoss boot
root@Bnode:~#   zlogin zoss
[Connected to zone 'zoss' pts/3]
Oracle Corporation     SunOS 5.11    11.1  September 2012
root@zoss:~#
```

:: Detach und Attach

```
root@Bnode:~# zoneadm -z zoss halt
root@Bnode:~# zoneadm -z zoss detach
zoneadm: zone 'zoss': warning(s) occured during processing URI: 'iscsi://s11-serv1.
mydomain.com/luname.naa.600144f09f3d4000000051a3c32d0001'
Could not remove one or more iSCSI discovery addresses because logical unit is in use
Exported zone zpool: zoss_rpool
Unconfigured zone storage resource(s) from:
    iscsi://s11-serv1.mydomain.com/luname.naa.600144f09f3d4000000051a3c32d0001
root@Bnode:~# zpool list
NAME  SIZE     ALLOC  FREE    CAP  DEDUP HEALTH       ALTROOT
rpool    11,9G    10,0G    1,94G    83%  1.00x    ONLINE    -
root@Bnode:~# ssh root@Anode
Password:
Last login: Tue May 28 01:52:44 2013 from bnode.mydomain.
Oracle Corporation      SunOS 5.11      11.1    September 2012
Anode@~#
```

ZFS Verschlüsselung

- Kann Daten vor unerlaubtem Zugriff sichern
- Ermöglicht sicheres Löschen durch "Vergessen" des Schlüssels
- Kann nur beim Erstellen des Filesystems aktiviert werden
- 2 Schlüssel Modell
  - :: innerer Schlüssel Std: aes-128-ccm, konfigurierbar
  - :: äußerer Schlüssel, kann gewechselt werden

    keysource=raw | hex | passphrase,prompt |file://|pkcs11:|https:/

    Der Schlüssel muss beim Import des Pools (also auch bei jedem Boot) verfügbar sein
- man zfs_encypt

ZFS Verschlüsselung, basic

```
# zfs create -o encryption=on muppets/piggy
Enter passphrase for 'muppets/piggy': xxxxxxxx
Enter again: xxxxxxxx
# zfs get keystatus,keysource muppets/piggy
NAME              PROPERTY    VALUE                SOURCE
muppets/piggy     keystatus   available            -
muppets/piggy     keysource   passphrase,prompt    local
# zfs unmount muppets/piggy
# zfs get keystatus muppets/piggy#unmount enfernt den Schlüssel NICHT
NAME              PROPERTY    VALUE                SOURCE
muppets/piggy     keystatus   available            -
# zfs mount muppets/piggy          #mount verlangt den Schlüssel nicht
# zfs key -u muppets/piggy         #entfernt den Schlüssel
# zfs mount muppets/piggy
Enter passphrase for 'muppets/piggy': xxxxxxxx
```

Zoss und Verschlüsselung

- Die Kombination beider Features ermöglicht
  - Zugangsbeschränkung: Nur Nodes, die den Schlüssel kennen, können die Zone anhängen
  - Sicherheit: Die verschlüsselten Datenblöcke werden im Netzwerk übertragen
  - Letzte Dinge: Das sichere Löschen der Zonen *zvols*
  - Die NGZ selbst und der Storage-Server brauchen den Schlüssel nicht

Zoss und Verschlüsselung: Konfiguration

- Die Verschlüsselung kann nur beim Anlegen von Dateisystemen aktiviert werden, also muss vor der Installation der Zone der verschlüsselte Pool erstellt werden.
  - :: Bsp wie zuvor. (zone ist *configured*, zpool *exportiert* oder *destroyed*)

```
bnode@~# pktool genkey keystore=file keytype=aes keylen=128 \
         outkey=/zones/zkeystore/zosskey
bnode@~# zpool create -fO encryption=on \
         -O keysource=passphrase,file:///zones/zkeystore/zosskey   zoss_rpool \
  /dev/dsk/c0t600144F09F3D4000000052866662000 1d0
bnode@~# zpool export zoss_rpool
bnode@~# zoneadm -z zoss install
Configured zone storage resource(s) from:
      iscsi://s11-serv1.mydomain.com/luname.naa.600144f09f
3d40000000528666620001
Imported zone zpool: zoss_rpool
Progress being logged to /var/log/zones/zoneadm.20131119T154736Z.zoss.install
```

Zoss und Verschlüsselung:

```
bnode@~# zoneadm list -cv
  ID   NAME      STATUS    PATH              BRAND    IP
   0   global    running   /                 solaris  shared
   1   zoss      running   /zones/zoss       solaris  excl
bnode@~# zlogin zoss
[Connected to zone 'zoss' pts/2]
Oracle Corporation    SunOS 5.11    11.1  September 2012
zoss@~# zfs get encryption,keysource,keystatus rpool
NAME  PROPERTY    VALUE                              SOURCE
rpool    encryption    on                             inherited from $globalzone
rpool    keysource     passphrase,file:///<keyfile>   inherited from $globalzone
rpool  keystatus       available                      -
zoss@~# zfs key -c rpool
cannot change wrapping key for 'rpool': keysource property not local, change key on
'$globalzone'.
```

Links, Quellen

- Solaris 11 Cheatsheet *Joerg Moellenkamp*
  http://www.oracle.com/technetwork/server-storage/solaris11/documentation/solaris-11-cheat-sheet-1556378.pdf
- To Dedupe or not to Dedupe...Constantin Gonzalez
  http://constantin.glez.de/blog/2011/07/zfs-dedupe-or-not-dedupe
- Dedupe – be careful!
  http://www.zfsbuild.com/2011/11/18/dedupe-be-careful
- Oracle Solaris: Zones on Shared Storage
  http://www.oracle.com/technetwork/articles/servers-storage-admin/zones-on-shared-storage-1896088.html
- Immutable Zones on Encrypted ZFS
  http://blogs.oracle.com/darren/entry/immutable_zones_on_encrypted_zfs
- User home directory encryption with ZFS
  https://blogs.oracle.com/darren/entry/user_user_home_directory_encryption

SOLARIS @STEP

Danke

Martin Muschkiet, as-systeme