

ZFS

Maximilian Haupt

me@makz.net

MDLUG e.V.

Magdeburg, 15. April 2008

Ablauf

- Vor ZFS
- ZFS
- Praxis
- Anwendungsbeispiele

vor ZFS 1

- Festplatte = Partition
- 4 primäre Partitionen
- logische Partitionen
- Nachteile:
 - beschränkt auf eine Festplatte
 - keine Datensicherheit
 - geringe Geschwindigkeit

vor ZFS 2

- Hardware RAID
- Nachteil:
 - sehr teuer
 - unflexibel
- Vorteil:
 - eigene CPU

vor ZFS 3

- LVM – Logical Volume Manager
- Vorteil:
 - Abstraktion (PV – VG – LV)
 - Software RAID
- Nachteil:
 - kompliziert
 - „klassische“ Dateisysteme
 - Kompatibilität

vor ZFS 4

- Immer noch zu viele Nachteile!
→ wie verbessern?

ZFS

- einfach!
 - nur zwei Befehle: `zpool` & `zfs`
 - *`zpool create mypool raidz sda sdb sdc sdd`*
 - *`zfs sharenfs mypool`*

ZFS

- simple Abstraktion:
 - Festplatten im Pool
 - Partition = Pool
 - hierarchische „Unterpaktionen“

ZFS

- dynamische Pools
 - zpool [create, add, remove, attach, detach, replace, online, offline]
 - Pool ändert automatisch Dateisystem-Größe

ZFS

- „unendlich“ groß:
 - 128Bit Adressierung
 - 309.485.009.821.345.068.724.781.056 TB

ZFS

- keine Partionen:
 - Baumstruktur
 - Knoten im Baum mit speziellen Eigenschaften
 - Pool = Wurzelknoten
 - *zfs create mypool/homes*

ZFS

- Quotas/Reservation
 - Keine Beschränkung durch Partition
 - Obergrenze/Untergrenze setzen
 - *zfs set quota=5G mypool/homes*
 - *zfs set reservation=2G mypool/homes*

ZFS

- Komprimierung:
 - on the fly
 - on, off, lzjb, gzip, gzip-[1-9]
 - *zfs set compression=gzip-1 mypool/homes*

ZFS

- Backup&Restore:
 - einfacher Snapshot vom Knoten
 - *zfs snapshot mypool/homes@one*
 - *zfs rollback mypool/homes@one*
 - *zfs clone mypool/homes@one mypool/one*

ZFS

- selbstheilend:
 - Prüfsummen auf Festplattenebene
 - bei Fehler: automatische Korrektur
 - *zpool scrub & zpool clear*

ZFS

- logging:
 - alle Befehle werden geloggt
 - *zpool history*

ZFS

- diverse RAID Modi:
 - JBOD, Mirror (RAID 1), RAIDZ, Hot Spare
 - kein Hardware RAID mehr nötig
 - Sun Fire mit 48 Festplatten

ZFS

- Plattformunabhängig 1:
 - Solaris 10 / OpenSolaris
 - Referenzimplementierung von Sun
 - für Zones verwendet
 - Boot – OpenSolaris

ZFS

- Plattformunabhängig 2:
 - FreeBSD:
 - seit April 2007 offiziell im Kernel
 - sehr stabil
 - Boot + iSCSI – work in progress

ZFS

- Plattformunabhängig 3:
 - Mac OS X:
 - sollte offizielles Dateisystem für 10.5 werden
 - in 10.5 nur readonly
 - r/w über Entwicklerpaket

ZFS

- Plattformunabhängig 4:
 - Linux:
 - Lizenzkonflikt -> nicht im Kernel
 - Google Summer of Code -> FUSE
 - Beta Status

Praxis - zpool

- Pool erstellen
- Festplattentausch
- Selbstheilung

Praxis - zfs

- Quots&Reservation
- Snapshots
- Komprimierung

Anwendungsbeispiele

- Desktop PC
 - mypool/home/max & mypool/source
 - komprimiert
 - Snapshots
 - mypool/media
 - unkomprimiert
 - sharenfs

Anwendungsbeispiele

- User Homes

- # zfs list

NAME	USED	AVAIL	REFER	MOUNTPOINT
pool1	183G	487G	28.5K	/pool1
pool1/home	83.7G	487G	28.5K	/export/home
pool1/home/staff	3.38G	487G	30.5K	/export/home/staff
pool1/home/staff/iws	1.92G	487G	37.5K	/export/home/staff/iws
pool1/home/staff/iws/bluemel	15.6M	496M	15.6M	/export/home/staff/iws/bluemel
pool1/home/staff/iws/elkner	453M	4.56G	453M	/export/home/staff/iws/elkner
...				
pool1/home/stud/cse/mhaupt	112M	144M	112M	/export/home/stud/cse/mhaupt
...				
pool1/sfw	91.7G	487G	31.3G	/export/sfw
pool1/sfw/sparc	27.7G	487G	29K	/export/sfw/sparc
pool1/sfw/sparc/apps	7.49G	8.51G	7.49G	/export/sfw/sparc/apps
pool1/sfw/sparc/local	587M	3.43G	587M	/export/sfw/sparc/local

Anwendungsbeispiele

- Solaris Zones
 - Isolierte Instanzen von Solaris
 - jeder Container bekommt ein zfs-FS

Anwendungsbeispiele

- Sun Fire x4500
 - 2 Dual Core Opteron
 - 16GB Ram
 - 48 SATA Festplatten
 - kein Hardware RAID!
 - Solaris 10 + ZFS

Fragen?

- Vielen Dank für die Aufmerksamkeit!

Quellen/Links

- 1) LVM Howto <http://tldp.org/HOWTO/LVM-HOWTO/>
- 2) ZFS Fuse <http://zfs-on-fuse.blogspot.com/>
- 3) ZFS <http://www.lildude.co.uk/zfs-cheatsheet/>