

linuxwochen

PETER-PAUL WITTA



Networked Storage mit NFS und iSCSI auf



Ing. Peter-Paul Witta
paul.witta@CUBiT.at



UnixNetworkStorage: NFS und iSCSI

- iSCSI
 - blockbased
 - Filesystem im Host
 - TCP/IP, bonding
 - multi-gbE oder 10gbE
 - Infiniband
 - (oder FibreChannel?)
 - Performance mit dedizierter Hardware
- NFS
 - filebased
 - Filesystem im Storagesystem
 - TCP/IP
 - multi-gbE oder 10gbE
 - Infiniband
 - Performance mit dedizierter Hardware





NetApp Architektur

- seit > 5 Jahren Entwicklung NFS
- #1 iSCSI Anbieter auf dem Markt
- Marktführer NAS
- dedizierte Hardware (Appliance, nur single-purpose (Storage))
- Administrierbar wie Unix-System
- gebaut für NFS
- logbasiertes hochleistungs-Filesystem
- Snapshots ohne Verarbeitungsaufwand
- Snaprestore, Replikation
- Hardware NVRAM Speicher für NFS Transaktionen
- FibreChannel Backbone
- Cluster



NetApp Architektur

- aktive FibreChannel Controller für Disk-Ansteuerung
- aktive Diskerkennung
- eigene Disk Firmware
- direkte Ansteuerung der Disk-Ports
- hohe Leistung aus echtzeit-Betriebssystem DataONTAP
- redundante Verkabelung
- redundante Architektur
- All Parts hot-swappable
- Skalierbar
- beginnend mit kleinen Systemen mit 1-2 TB
 - bis hin zu 1000 Disk Systemen



NetApp Cluster

- Shared Storage Backend
- Locking und Cache-Mirroring über dedizierten Interconnect
- volles Mirroring der Disks
- 300-500m Distanz
- Cluster-Failover ohne Transaktionsverlust garantiert
- Cluster-Failover für Hostsysteme transparent
- Stretch Metrocluster: <80 km Distanz
- Switches notwendig
- 2 redundante FibreChannel Backend-Fabrics



Unix Filesharing – NFS

- native Unix Filesharing
- Sun NFS ist Grundlage von NFS
- Unix Berechtigungen
- gute Integration in NIS
- Remote-Boot und Remote-Root Filesystem möglich
- Einfaches Prinzip (bis einschl. v3)
- basierend auf Sun RPC
- Schnelles NFS setzt Kernel Daemon voraus im Server
- Userspace NFS langsamer aber sicherer (alt!)



Storage Konsolidierung

- Statt SAN zur Plattenanbindung Ethernet
- warum nicht NFS statt iFCP und iSCSI
- weniger Overhead
- mit ausreichend Tuning entsprechend Schnell
- Standard-Server oft langsam
- Tuning, tuning, tuning und richtige Hardware
 - viel RAM (>5 GB) und viel CPU (dual 2 Ghz)
 - richtiger IO-Bus (Fibre Channel, Escalade Storage Switch)
 - richtige Platten
 - ausreichend Spindeln für rapid random IO
 - ausreichend Bandbreite (mehrere 1000baseTX)
- oder spezielle Maschine (NetApp)



The Next Step: Desktop

- Diskless Linux Desktop
- Remote Boot Solution
- Semi Server-Based: Nutzung lokaler CPU und MEM Ressourcen
- DHCP kann NFS Rootserver mitteilen
- Performance Tuning wichtig
- kein Unterschied zu lokalem Arbeiten
- richtige mount Option, ggf. Failover-Systeme
- FAM für Desktop wichtig



NFS im Detail

- basiert auf Sun RPC
- udp und tcp möglich
- Performance erfordert individuelles Tuning
- Performance wichtig:
 - LAN: 100MBps => 8 MB/sec, Disk: 50 MB/sec
 - auch mit Gigabit LAN langsamer als Disk
 - RawIO theoretisch schneller aber praktisch?
 - mit Tuning kann es verschmerzbar werden
- File (Echtdaten) und Attribut (Metadaten) Zugriff
- Caching von Metadaten zur Entlastung im RAM
- hard/soft Mount Option



Eigenschaften

- Namespace: server:/pfadname/dir/filename.txt
- Z.B. mount cube2:/mnt/shares /mnt/c2/shares
- Mount-Optionen:
 - mount -orsize=8192,hard /mnt/c2/shares
- NFS Filesystemtreiber am Client notwendig
- am Server
- Server auch für kleinere Systeme (PDA)
- Auch im Embedded Bereich möglich
- Auch für Windows-Systeme erhältlich
- Standardprotokoll RFC 1094, RFC 3010 (v4), RFC 1813 (v3)
- Routebar, auch über Internet möglich



NFS im Einsatz

- Mount Options hard/soft
- tcp vs. udp: Wann welche Option?
- Paketgrösse bei UDP und TCP
- Symlink Support (lokale Auflösung!)

sys1:

```
/a-loc 2M
/exp
/sub
/lnk->/a-loc
(2M)
```

sys2:

```
/a-loc 1M
/mnt
/sub
/lnk->/a-loc
(1M)
```

```
mount sys1:/exp /mnt
```

```
sys1;/exp/sub/lnk/a
->sys1:/a-loc
```

```
sys2:/mnt/sub/lnk/a ->
sys2:/a-loc
```



NFS im Einsatz (2)

- Re-Export möglich: Storage-Hubbing
- Manchmal mounten Server Clients :-)
- TCP-IP reicht; Internet-tauglich; mit TCP sogar durch SSH-Tunnel
- zB: Zentral-Server mountet Filialserver via FRAD, erlaubt Zugriff auf alle Filialen und Rollouts per cp in Subdirectories
- volle Symlink, Hardlink, Attribut-Unterstützung
- “true Unix Technology” -> case sensitive, file-locking,...



NFS mit Datenbanken

- Filesystem schnell und komfortabel
- MySQL
- PostgreSQL
- Oracle
- bei Clustern kein DLM notwendig
- r/o Sharing kein Problem
- schneller IO
- Achtung: pro Mountpoint nur eine IO-queue
- Aufspalten über mehrere Mounts für hohe Leistung



Tuning

- TCP vs.UDP
 - TCP bei WAN und unreliable slow Links mit Packetloss
- Paketgrösse bei UDP: Fragmentierung vermeiden
- $MTU = NFS\text{-block-size} + UDP\text{-hdr} + IP\text{-hdr} + Ethernet\text{-hdr}$
- MTU 9000 bei Gigabit Ethernet:
 - richtigen Switch nehmen!!



Timeouts und Tuning im Detail

- acregmin/max: Attribute reguläre Datei
- acdirmin/max: Attribute Verzeichnis
- actimeo= alle 4 :-)
- rsize
- wsize
- timeo=Timeout für Algorithmus in 1/10 sec.
- noac bei gleichzeitigem Zugriff



Linux NFS Details

- Kernel 2.4: nfsv3
- Kernel 2.6: experimentielle Patches für nfsv4
- nfsv3 ist aktuell und stabil nutzbar
- Kernel NFS v3 Server Produktiv
 - neue Features: BS>8k (bis 32K), TCP
- Userspace-NFSd
 - kein Locking Support
- Threadanzahl definierbar
 - Faustregel: $2 * \#cpu$
 - bei langsamen Systemen mehr => weichere Verarbeitung
 - bei schlechten Links mal TCP probieren
 - IO muss mithalten können: sar, vmstat



iSCSI

- RFC3720, 3980 (update), 3783 (command ordering)
- Blockbasiertes Protokoll
- erzeugt Blockdevice (/dev/sda)
 - mit Anbindung via TCP/IP an Speichersystem
 - FibreChannel Semantik
 - SCSI/FC over TCP/IP
 - Anmeldung an Portalserver
 - iqn-ID ersetzt SCSI-ID
 - LUN-ID wird gemappt und angezeigt
 - Nutzung wie herkömmliches Blockdevice
 - Routebar
 - Filesystem läuft im Rechner
 - LVM, DM-Mirror, kombinierbar



iSCSI

- OpeniSCSI und LinuxiSCSI seit April vereint
- Target und Initiatoren für Linux
- einfache Benutzung
- Kernel-Treiber
- Ausfallsicherheit durch Multipathing oder Bonding
- Daemon: iscsid
- Admin: iscsiadm



iSCSIadm

Discover targets at a given IP address:

```
iscsiadm --mode discovery --type sendtargets --portal 192.168.1.10
```

Login, must use a node record id found by the discovery:

```
iscsiadm --mode node --targetname iqn.2001-05.com.doe:test  
--portal 192.168.1.1:3260 --login
```

Logout:

```
iscsiadm --mode node --targetname iqn.2001-05.com.doe:test  
--portal 192.168.1.1:3260 --logout
```

List node records:

```
iscsiadm --mode node
```

Display all data for a given node record:

```
iscsiadm --mode node --targetname iqn.2001-05.com.doe:test  
--portal 192.168.1.1:3260
```



iSCSI Files

FILES

`/etc/iscsi/iscsid.conf`

The configuration file read by `iscsid` and `iscsiadm` on startup.

`/etc/iscsi/initiatorname.iscsi`

The file containing the iSCSI `InitiatorName` and `InitiatorAlias` read by `iscsid` and `iscsiadm` on startup.

`/etc/iscsi/nodes/`

This directory contains the nodes with their targets.

`/etc/iscsi/send_targets`

This directory contains the portals.



iSCSI und NFS

iSCSI

- Block based, kein Application-Level Locking
- kein Filesystemoverhead (kein Filesystem)
- Ablöse von FibreChannel möglich
- iSAN (nicht NAS)
- Windows, UNIX, Linux

NFS

- Filesmantics
- Berechtigungen
- native Unix Filesharing
- Filesystem in Storage Appliance schneller als mit iSCSI + Filesystem
- direktes Caching möglich



Tipps, Tricks, Architektur

- schnelles Netzwerk
- schnelle Disks und IO-Subsystem
- Attribute Caching wo immer möglich
- Server IO-Tunen
- oder gleich NetApp :-)
- NFS ist nicht alt
- NFS v3 und v4 mit NetApp entwickelt
- v4: Byte Range Locking, SMB-ähnliche Features, andere Semantik als v3
- v3: TCP, UDP, Blocksize und Performance
- jedes System kann von NFS profitieren



Dedizierte Hardware

- NetApp Filer mit Cluster Option
- FC/AL Interfaces
- active/active load sharing
- Geheimnis: gute und schnelle Platten
- gute Softwarearchitektur
- viel Cache
- Demo auf den Linuxwochen Wien / CUBiT IT

linuxwochen

PETER-PAUL WITTA



Networked Storage mit NFS und iSCSI

DANKE!

Ing. Peter-Paul Witta
paul.witta@CUBiT.at