Software-RAID mit Linux

Sven Velt
wampire@lusc.de

Übersicht

- Einführung in RAID
  - Was ist RAID
  - Technische Implementierung
- Soft-RAID unter Linux
  - Unterstützung
  - Aufsetzen eines RAIDs
  - Fehler beseitigen
- Linux-Installation auf RAID
- Einschub: LVM
- Recovery von Multiple-Disk-Failure

Einführung in RAID

- Was ist RAID
  - Wozu RAID
  - RAID-Levels
- Technische Implementierung
  - Hardware-RAID
  - Software-RAID
  - "Host-RAID"

Was ist RAID, Wozu RAID

Abkürzung "RAID":

- Reduntant Array of Inexpensive Disks
- also ein redundanter Stapel von günstigen Platten

Die Idee dahinter:

- Ausgangspunkt: mehrere, meist mit geringen Speicherplatz, günstige Festplatten
- Endpunkt: eine virtuelle Festplatte mit viel Speicherplatz und/oder Redundanz

Was ist RAID, Wozu RAID

Vorteile:

- 4x 100GB günstiger als 1x 400GB
- 800GB nicht erhältlich (8x 100GB)

- Redundanz: u.U. kann eine HD "einfach" kaputt gehen
- Geschwindigkeit: Verteilung der Zugriffe über die Platten
- Größe: >1TB-HDs gibt es (noch) nicht

RAID-Levels

Es gibt im wesentlichen 7 RAID-Levels (0 bis 6)

- Linux unterstützt die wichtigsten, manche auf mehreren Wegen
- Weitere RAID-Levels aus diesen aufgebaut bzw. kombiniert

Es wird nur auf die von Linux unterstützten eingegangen, jeweils mit Beispiel dazu

- verfügbare Hardware: 2, 4, 8 HDs mit jeweils 100GB

RAID-0

RAID-0 aka "Stripping"

- Gesamtgröße: 2 x 100GB, 400GB, 800GB
- Paralleles Lesen/Schreiben auf/von HDs, hohe Geschwindigkeit
- Bestes Verhältnis Brutto-/Netto-Speicherplatz (100%)
- ABER: eine defekte HD macht ganzes Array kaputt

RAID-1

RAID-1 aka "Mirroring"

- Gesamtgröße: immer 100GB
- Schreiben immer auf alle HD, langsam
- Lesen von allen HD parallel, schnell
- Hohe Redundanz, 3 defekte HDs werden verkraftet
- ABER: Ungünstiges Verhältnis Brutto-/Netto-Speicherplatz (50%, 25%, 12,5%)

RAID-4/5 Gemeinsamkeiten

RAID-4, RAID-5

- Gesamtgröße: n/a, 300GB, 700GB
- Paralleles Lesen/Schreiben auf/von HDs, rel. hohe Geschwindigkeit
- "letzte" HD wird mit Checksumme aus den anderen HDs beschrieben
- "leichte" Redundanz: Eine HD kann kaputt gehen
- Gutes Verhältnis Brutto-/Netto-Speicherplatz (n/a, 75%, 87,5%)
- ABER: CPU-Last für Berechnung der Checksummen

RAID-4/5 Unterschiede

RAID-4

- Eine HD für alle Checksummen
- Array leicht erweiterbar
- leicht ungleichmäßige Auslastung

RAID-5

- Checksummen rotierend auf HDs verteilt
- gleichmäßige Auslastung
- ABER: Array nicht erweiterbar

RAID-6

RAID-6 aka "Double Parity"

- Gesamtgröße: n/a, 200GB, 600GB
- ähnlich RAID-5, aber 2. HD für Checksummen (diagonal)
- gute, bezahlbare Redundanz: 2 HDs können kaputt gehen
- Bei vielen HD gutes Verhältnis Brutto-/Netto-Speicherplatz (n/a, 50%, 75%)

RAID-Levels

- Beispiel-Kombination "RAID-5+1"
  - 2 RAID-5-Arrays werden mit RAID-1 gespiegelt
  - Bei 2x4 HDs a 100GB ergibt das Netto 300GB (37,5%)
  - hohe Redundanz: mindestens 3 HDs können ausfallen, maximal 5
- Sinn und Unsinn solcher Lösungen lassen sich lange diskutieren

Technische Implementierungen

- Hardware-RAID
- Software-RAID
- "Host-RAID"

Hardware-RAID

- Spezielle Controller erforderlich
- besondere Treiber notwendig
- teuer, da eigener Prozessor
- kaum CPU-Last, da eigener Prozessor

- häufig in Verbindung mit SCSI-HotSwap-HDs in Servern zu finden
- SATA-Versionen immer häufiger zu finden

Software-RAID

- Standard-HDs an Standard-Controller
- einfache Linux-Unterstützung reicht
- gerine Extrakosten für weitere HD-Controller
- höhere CPU-Last
- nicht immer so "stabil" wie Hardware-RAID

- günstige Möglichkeit für Home- bzw. kleine Internet-Server
- ATA aufgrund diversers Beschränkungen nicht so gut geeignet
- sehr "hübsch" mit SATA-HDs

"Host-RAID"

- Besonderer Controller, angeblich mit RAID-Funktionalität, günstig
  - meist nur RAID-0 und RAID-1
- unter Windows: "einwandfrei"
- unter Linux: einzelne HDs, keine Sicht von RAID

- RAID-Funktionalität steckt im Treiber!
  - Keine CPU-Entlastung
  - teilweise eigenes Format, Probleme bei Crash

- Linux-Treiber nur sehr vereinzelt verfügbar

Soft-RAID unter Linux

- Unterstützung
- Aufsetzen eines RAIDs
  - im bestehenden System
  - für ein neues System

Unterstützung

Alle vorgestellten RAID-Levels werden von Linux unterstützt
- RAID-0 und RAID-1 über "Multiple Devices" und "LVM"
- RAID-4, -5, -6 nur über "Multiple Devices"

- Entsprechende Unterstützung sollte im Kernel aktiviert sein

Weitere Software
- (raidtools)
- mdadm

Aufsetzen eines RAIDs

Erster Schritt: Daten-Partition:

- Daten-Partition auf RAID sehr einfach zu realisieren
- System ist bereits gebootet, Module und Tools können nachgeladen werden

Vorgaben:

- 2 Partitionen (hde1 und hdg1)
- Partitions-Type mit fdisk oder cfdisk auf "FD" gesetzt (Linux RAID autodetect)
- RAID-1 gewünscht

RAID-1 Daten-Partition

- `dd if=/dev/zero of=/dev/hde1 bs=1024 count=1`
- `dd if=/dev/zero of=/dev/hdg1 bs=1024 count=1`
- `mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/hde1 /dev/hdg1`
- `cat /proc/mdstat`
- `mkfs -t ext3 /dev/md0`

- Ab sofort kann auf das RAID-Laufwerk zugegriffen werden

RAID zum Testen

Keine Partition frei zum Testen? Kein Problem:

- `dd if=/dev/zero of=file1 bs=1M count=50`
- `dd if=/dev/zero of=file2 bs=1M count=50`
- `losetup /dev/loop0 file1`
- `losetup /dev/loop1 file2`
- `mdadm -C /dev/md0 -l1 -n2 /dev/loop0 /dev/loop1`
- `cat /proc/mdstat`
- `mkfs -t ext3 /dev/md0`

Stoppen der RAID-Arrays

- Stoppen des Arrays
  - `mdadm -S /dev/md0`
  - `losetup -d /dev/file1`
  - `losetup -d /dev/file2`

Anlegen eines "großen" RAID-5

/dev/loop0-2 sind konfiguriert

- Es wird zusätzlich der RAID-5-Parity-Algorithmus angegeben (-p ls) und die Chunk-Size (-c 128k)

- `mdadm -C /dev/md0 -l5 -n3 -p ls -c 128k /dev/loop[012]`

- Man kann sich auch genauere Informationen anzeigen lassen

- `mdadm --detail /dev/md0`

Simulierter Fehler im RAID-5

Voraussetzungen:

- /dev/md0 ist ein RAID mit Redundanz (hier RAID-5)
- /proc/mdstat zeigt an, dass keine Fehler existieren
  - `md0 : active raid5 [dev 07:02][2] [dev 07:01][1] [dev 07:00][0]`
  - `      102272 blocks level 5, 64k chunk, algorithm 2 [3/3] [UUU]`

Simulierter Fehler im RAID-5

- Soft-Failure einer HD
  - `mdadm --manage --set-faulty /dev/md0 /dev/loop0`
  - `cat /proc/mdstat`
    - `md9 : active raid5 [dev 07:02][2] [dev 07:01][1] [dev 07:00][0](F)`
    - `      102272 blocks level 5, 64k chunk, algorithm 2 [3/2] [_UU]`
  - `mdadm /dev/md0 --remove /dev/loop0`

Fehler beseitigen

Voraussetzung:

- "Neue" HD ist eingebaut (dd hilft ;-)

- `mdadm /dev/md0 --add /dev/loop0`
- `cat /proc/mdstat`
  - `md9 : active raid5 [dev 07:00][3] [dev 07:02][2] [dev 07:01][1]`
  - `      102272 blocks level 5, 64k chunk, algorithm 2 [3/2] [_UU]`
  - `      [>....................]  recovery =  0.0% (868/51136) finish=1.8min speed=434K/sec`

Linux-Installation auf RAID

- Je nach Distribution unterschiedlich
- Teilweise in die Installation integriert
- Nicht immer einfach

- "Von Hand" hat Vorteile
  - man weiß wie es aufgesetzt wurde
  - kann im Fehlerfall leichter recovern
  - besser mit den Tools vertraut

Debian schematisch, mit Knoppix

- Knoppix booten
- Automounter abschalten! (/etc/init.d/autofs stop)
- Festplatten partitionieren
- RAID-Laufwerke anlegen, anschließend formatieren
- Verzeichnis anlegen, späteres / dort hinmounten
- andere Mount-Points anlegen
- debootstrap woody MOUNTPOINT
- chroot MOUNTPOINT
- Kernel compilieren (mit RAID!)
- LILO einrichten

LILO - für RAID

Die wichtigsten ˜nderungen an der lilo.conf

- boot=/dev/md0
- root=/dev/md1
- raid-extra-boot=/dev/hda,/dev/hdi

Debian-Umzug schematisch

- Installation auf eine HD
  - Extra-HD
  - eine aus dem späteren RAID
- Anlegen des RAID, evtl. mit Failed HD
- Kopieren des Systems auf das RAID (Harddisk-Upgrade-Howto)
  - Knoppix booten
  - Im System selbst (init 1)
- chroot auf das RAID
- LILO-Konfiguration ändern

Ausblick: LVM

- Wie partitioniert man mehrere hundert GB?
  - Lösung: LVM

- LVM - Logical Volume Manager
  - Partitionen im Betrieb vergrößer- und verkleinerbar
  - RAID-0 und RAID-1 "integriert"
  - Am Anfang nicht ganz einfach

Recovery

Simulation

- RAID-5 mit 4 Files einrichten und starten, befüllen
- RAID beenden, ein File kopieren, RAID starten, weiter befüllen
- RAID beenden

- RAID starten mit 2 aktuellen und dem alten File
  - siehe Software-RAID-HowTo, Kapitel 8.1

Software-RAID mit Linux

Sven Velt
wampire@lusc.de