

Lokales Storage Teil 1

Linux-Kurs der Unix-AG

Zinching Dang

08. Juli 2015



Lokales Storage im Allgemeinen

- ▶ Datenträger, die direkt am Host angeschlossen sind
- ▶ Anbindung über verschiedene Bus-Systeme möglich, z. B. SAS, SATA
- ▶ Verwaltung durch das Betriebssystem oder durch einen Hardware-Controller

Partitionierung

- ▶ Aufteilen des gesamten Speicherbereichs einer Festplatte in mehrere, kleinere Bereiche
- ▶ notwendig, um mehrere Betriebssysteme oder Dateisysteme zu verwenden
- ▶ Aufteilung steht in der sog. **Partitionstabelle**
 - ▶ verschiedene Standards für Partitionstabellen, u. a. MBR und GPT
 - ▶ befindet sich im Anfangs-/Endbereich des Datenträgers

Partitionstabellen (1)

► Master Boot Record

- ▶ belegt die ersten 512 Bytes des Datenträgers
- ▶ enthält den Code des Bootloaders und die Partitionstabelle
- ▶ primäre und erweiterte Partitionen (max. 4 Partitionen insgesamt, max. 1 erweiterte Partition)
- ▶ innerhalb einer erweiterten Partition: beliebig viele logische Partitionen
- ▶ max. Datenträgergröße auf 2 TiB beschränkt (bei 512-Byte-Sektoren)

Partitionstabellen (2)

- ▶ **GUID (Globally Unique Identifier) Partition Table**
 - ▶ MBR-Nachfolger, Teil des UEFI-Standards
 - ▶ beliebig viele Partitionen, max. Datenträgergröße 8 ZiB
 - ▶ GPT-Header im Anfangsbereich (Primary GPT-Header) und im Endbereich (Secondary GPT-Header) des Datenträgers

Partitionierungstools

- ▶ *fdisk (nur MBR, kein GPT) und parted
- ▶ Partitionierung per Kommandozeilen-Befehl oder interaktiv möglich
- ▶ auch als GUI-Tools verfügbar: gparted, qtparted
- ▶ hier: nur parted

parted

- ▶ Aufruf: `parted /dev/sda`
 - ▶ bearbeitet die Partitionstabelle des Datenträgers sda
- ▶ wichtigsten Kommandos:
 - ▶ `help` und `help <CMD>`: allgemeine Hilfe und Hilfe zu einem bestimmten Kommando
 - ▶ `print`
 - ▶ `select <DEVICE>`
 - ▶ `mktable`
 - ▶ `mkpart`
 - ▶ `set`

Lab: Anlegen und Einbinden virtueller Festplatten

- ▶ mit dem `virt-manager` zwei virtuelle Festplatten anlegen und an den eigenen Server anbinden
- ▶ überprüfen, ob die neuen Festplatten erkannt wurden
- ▶ Festplatten partitionieren

RAID

- ▶ **Redundant Array of Independent Disks**
- ▶ meist werden mehrere Festplatten zusammen im Verbund betrieben
 - ▶ (höhere) Redundanz
 - ▶ größere Gesamtkapazitäten
 - ▶ höherer Datendurchsatz
- ▶ Hardware-Realisierung als spezieller RAID-Controller
- ▶ Software-Realisierung als Treiber
- ▶ bei sog. Fake-RAID ist nur ein Datenträger sichtbar, die RAID-Operationen werden jedoch in der CPU ausgeführt

Hardware- und Software-RAID

- ▶ Hardware-RAID ist performanter, da der RAID-Controller viele Aufgaben übernimmt
- ▶ bei einem Controller-Defekt wird ein identischer RAID-Controller benötigt
- ▶ Software-RAID ist sehr flexibel und unabhängig von der verbauten Hardware
- ▶ langsamer und Ressourcen-intensiver, da alle RAID-Operationen in der CPU ausgeführt werden

RAID-Level

- ▶ viele Möglichkeiten, Festplatten im RAID anzurichten
- ▶ verschiedene RAID-Level bieten unterschiedlich hohe Daten-Sicherheit
- ▶ am Gebräuchlichsten:
 - ▶ (RAID 0)
 - ▶ RAID 1
 - ▶ RAID 5
- ▶ Kombinationen verschiedener RAID-Level auch möglich (z. B. RAID 10, RAID 50)

RAID-Level

- ▶ RAID 0: Striping
 - ▶ bei n Festplatten werden die Daten auf alle n Festplatten verteilt
 - ▶ Gesamtkapazität entspricht Summe der Einzelkapazitäten
 - ▶ jedoch keine Redundanz und Defekt einer Festplatte führt meist zu Verlust **aller** Daten
- ▶ RAID 1: Mirroring
 - ▶ bei n Festplatten werden die Daten auf alle n Festplatten gespiegelt
 - ▶ sehr hohe Redundanz, Ausfall von $n-1$ Festplatten ohne Datenverlust möglich
 - ▶ keine Erhöhung der Gesamtkapazität, Gesamtgröße hängt von der kleinsten Festplatte ab

RAID-Level

- ▶ RAID 5
 - ▶ Kompromiss bzgl. Redundanz und Gesamtkapazität
 - ▶ benötigt mindestens 3 Festplatten
 - ▶ bei n Festplatten werden die Daten zusammen mit Paritäts-Informationen auf alle n Festplatten verteilt
 - ▶ $(n-1)$ -fache Gesamtkapazität und Ausfall einer Festplatte möglich

MD-RAID

- ▶ Software-RAID-Lösung unter Linux
- ▶ wird durch das Paket `mdadm` bereitgestellt
- ▶ u.a. werden RAID 0, RAID 1, RAID 5 und RAID 10 unterstützt
- ▶ aktuelle Informationen über das RAID befinden sich in `/proc/mdstat`

mdadm

- ▶ Aufruf: `mdadm <MODE> <DEVICE> <OPTION> <COMPONENT>`
 - ▶ MODE: Bearbeitungs-Modus
 - ▶ DEVICE: MD-RAID-Device
 - ▶ OPTION: Optionen für den Modus
 - ▶ COMPONENT: Komponenten des MD-RAID
- ▶ wichtige Modi:
 - ▶ `--create`: neues MD-RAID erzeugen
 - ▶ `--grow`: ein vorhandenes MD-RAID ändern
- ▶ wichtige Optionen:
 - ▶ `--raid-devices=<N>`: Anzahl der Datenträger im MD-RAID festlegen
 - ▶ `--level=<X>`: RAID-Level festlegen
 - ▶ `--add`: neue Datenträger als Spare zu einem bestehenden MD-RAID hinzufügen

Lab: MD-RAID

- ▶ RAID 1 über die beiden Partitionen erzeugen
- ▶ Status des RAID überprüfen