# Manuelle Software-RAID Konfiguration

## Einleitung

invis Server verfügen in aller Regel über zwei Festplatten, die über ein Software-RAID miteinander verbunden sind. Ziel dabei ist den Server gegen den Ausfall einer Festplatte abzusichern. Es gibt eine Vielzahl so genannter RAID-Level (mehr dazu hier), von denen in der Praxis vorwiegend die Level 0, 1, 10 und 5 Anwendung finden.

Dabei stellt RAID0 eine Ausnahme dar. RAID0 dient nicht der Erhöhung der Ausfallsichereit eines Computers, ganz im Gegenteil. RAID0 verbindet zwei oder mehr Festplatten so mit einander, dass Schreib- und Lesezugriffe auf den Verbund auf alle Platten verteilt werden. So erhöht sich mit jeder hinzugefügten Platte der Datendurchsatz um die Bandbreite mit der auf die einzelne Platte geschriben werden kann. Da die geschriebenen Daten gleichmäßig auf alle dem Verbund angehörenden Platten verteilt werden, bedeutet der Ausfall einer Platte den vollständigen Datenverlust des gesamten Verbundes. Das Risiko des Datenverlustes erhöht sich also mit der Anzahl der beteiligten Platten.

RAID1 ist eine einfache Spiegelung von zwei oder mehr Platten, d.h. auf allen Platten befindet sich der vollständige Datenbestand. Die Kapazität des Verbundes entspricht der Größe der kleinsten beteiligten Platte. Der Datendurchsatz bei Schreib und Lesezugriffen beschränkt sich den der langsamsten Platte und verschlechtert sich mit der Anzahl der am Verbund beteiligten Platten, da alle Daten auf alle Platten geschrieben werden müssen.

Die RAID-Level 0 und 1 setzen mindestens zwei beteiligte Festplatten voraus.

RAID10 ist eine Kombination der Level 1 und 0 (RAID 0 Verbund über mehrere RAID1 Verbünde) mit dem Ziel die hohe Datensicherheit von Level 1 ohne die damit verbundenen Performance-Nachteile zu realisieren. Die Kapazität des Gesamtverbundes entspricht dabei 50% der gesamten Plattenkapazität. (Auch die Umkehrung RAID01 - ein RAID1 über mehrere RAID0 Verbünde ist möglich, bietet aber eine etwas geringere Ausfallsicherheit.) RAID10 und RAID01 benötigen minimum 4 beteiligte Festplatten.

RAID5 hat ebenfalls die Erhöhung von Ausfallsicherheit und Datendurchsatz (bezogen auf ein RAID1) zum Ziel. Im Vergleich zu RAID10 wird allerdings die zur Verfügung stehende Plattenkapazität besser genutzt. RAID5 setzt mindestens 3 Festplatten voraus, die Kapazität des Verbundes berechnet sich (identische Platten vorausgesetzt) zu „Plattenkapazität * (Anzahl der Platten - 1)“.

Die Ausfallsicherheit aller RAID Level (0 ausgenommen) kann durch die Verwendung so genannter „Hotspare“ Platten erhöht werden. Dies sind einfach Ersatzfestplatten, die beim Ausfall einer Platte automatisch in den Verbund aufgenommen werden.

***Achtung:*** *Eine wie auch immer geartete RAID-Konfiguration ersetzt keine Datensicherung! Es gibt genügend Gründe dafür, warum gleich alle in einem Server verbauten Festplatten auf einmal ausfallen.*

## Linux Software-RAID einrichten

Ziel dieses Howtos ist die Einrichtung eines einfachen RAID-Level 1 Verbundes zweier Festplatten.

Grundsätzlich gilt, alle oben beschriebenen RAID-Level - inklusive Hotspare-Platten - lassen sich unter Linux ohne entsprechenden RAID-Controller auf Software-Ebene nachbilden. Weiterhin gilt, dass in einem Software-RAID nicht gesamte Platten sondern lediglich Partitionen in RAID-Verbünden organisiert werden.

Aus letzterem ergibt sich folgende Problemstellung. Nehmen wir an, dass wir für unsere Server-Installation für /boot, /, /var, /home und /srv eigene Partitionen benötigen. Dies würde für die RAID-Konfiguration bedeuten, dass insgesamt 5 RAID1 Verbünde angelegt werden müssten, was einen Festplattentausch nicht gerade einfach gestalten würde.

Um dem zu entgehen wird im allgemeinen LVM, eine weitere Technik zur logischen Datenträgerorganisation auf Software-Ebene, genutzt. Mehr dazu hier.

## Festplatten partitionieren

Starten Sie ihr System im ersten Schritt mit einem Linux-Rettungssystem. Für welches Sie sich hier entscheiden sollte keine Rolle spielen, solange dessen Kernel Software-Raid unterstützt. Der Einfachheit halber verwende ich meist das Rettungssystem der openSUSE Net-Install CD.

Zunächst müssen die am RAID beteiligten Festplatten identisch partitioniert werden. Am schnellsten geht das mit dem Klassiker ***fdisk***.

```
Komandozeile: fdisk /dev/sdx
```

***fdisk*** ist relativ leicht zu bedienen. Die verschiedenen Funktionen verstecken sich hinter verschiedenen Buchstabentasten. So liefert die Taste „m“ einen Überblick über die verfügbaren Kommandos. Sie können das Programm auf zwei Wegen wieder verlassen. Dass Kommando „w“ (write) schreibt die vorgenommen Änderungen auf die Platte und „q“ verlässt das Programm ohne zu speichern.

***Achtung:*** *Wenn Sie gebrauchte Platten verwenden, sollten Sie zunächst eine leere DOS-Partitionstabelle (Kommando: „o“) anlegen und das System vorsichtshalber neu starten.*

Legen Sie auf jeder Platte drei Partitionen an: /dev/sdx1 125MB, /dev/sdx2 512MB (oder 1024MB) und eine letzte Partition (/dev/sdx3) über den gesamten noch verfügbaren Plattenplatz. Das Kommando zum Anlegen neuer Partitionen ist „n“ (new). Kontrollieren können Sie Ihre Änderungen mit dem Kommando „p“ (print).

Jetzt müssen den einzelnen Partitionen nich die entpsrechenden Partitions-IDs (Partitionstypen) zugeordnet werden. Dies erreichen Sie jeweils über das Kommando „t“ (toggle). Partition 1 und 3 erhalten den Typ „fd“, die Kennung für automatisch erkannte Linux RAID Partitionen und Partition 2 erhält den Typ „82“ für swap.

Abschließend wird noch die jeweils erste Partition mit dem Kommando „a“ (active) als aktive Bootpartition markiert.

Prüfen Sie Ihre Partitionierung und speichern Sie sie mit „w“ in die Partitionstabelle der Festplatte.

Sollte Kernel noch die vorherige Partitionstabelle geladen haben (***fdisk*** gibt beim Verlassen einen

entsprechenden Hinweis), sollten Sie Ihr System vor dem Anlegen der RAID-Verbünde neu starten.

## RAID Verbünde anlegen

Ausgehend davon, dass das Partitionieren fehlerfrei funktioniert hat, verfügt jede Platte mit sdx1 und sdx5 jetzt über zwei als „Linux auto RAID“ gekennzeichnete Partitionen. Um diese jetzt zu organisieren kommt das Tool ***mdadm*** zum Einsatz:

```
Kommandozeile: mdadm --create /dev/md0 --level=1 --metadata=0.90 --raid-devices=2 /dev/sda1 /dev/sdb1
```

und für den zweiten Verbund:

```
Kommandozeile: mdadm --create /dev/md1 --level=1 --raid-devices=2 /dev/sda3 /dev/sdb3
```

Damit werden die beiden RAID1-Verbünde /dev/md0 und /dev/md1 angelegt. Die gezeigten Befehlszeilen sehe ich als weitestgehend selbsterklärend an. Die Option „create“ erwartet die Angabe des md-Devices (md = multiple disk) - die Zählung beginnt bei 0. „level“ definiert den RAID-Level und „raid-devices“ erwartet als Parameter die Anzahl der beteiligten Partitionen, die abschließend noch genannt werden müssen. RAID-Verbünde lassen sich (etwa wenn es gerade schnell gehen muss und nur eine Platte zur Hand ist) auch „untermotoriesiert“ zum Leben erwecken, dabei wird einfach anstelle einer beteiligten Partition das Schlüsselwort „missing“ angegeben. Vergessen Sie aber nicht den RAID-Verbund später zu komplettieren, sonst war der Arbeitsaufwand für die Katz. Damit sind die Möglichkeiten des Tools ***mdadm*** bei weitem nicht erschöpft, wie ein Blick in dessen manpage schnell verdeutlicht.

***Achtung:*** *Dass bei der Erstellung des RAID-Devices /dev/md0 die Option* ***metadata*** *auf den Wert* ***0.90*** *gesetzt wurde, liegt darin begründet, dass die unter openSUSE (bis min. 11.4) verwendete Grub-Version RAID-Verbünde der Metadata-Version 1.00 nicht erkennt.*

Der Linux-Kernel in dessen Hoheitsbereich die Verwaltung der Software-RAID-Verbünde liegt, beginnt jetzt unmittelbar diese zu synchronisieren. Beim kleinen /dev/md0 geht dies so schnell, dass es sich kaum beobachten lässt. Da /dev/md1 bei aktuellen Platten und der zugrunde liegenden Partitionierung vermutlich um die 500GB (oder größer) sein wird sieht hier die Sache etwas anders aus. Hier kann die Synchronisation durchaus eine halbe Stunde oder länger dauern. Beobachten lässt sich dies mit:

```
Kommandozeile: watch cat /proc/mdstat
```

Wird der Rechner vor beendeter Synchronisation neu gestartet, beginnt die Prozedur nach dem Neustart einfach von vorne. Noch während der Synchronisation können bereits Daten auf das RAID geschrieben werden, wobei ich dies bei der Erstsynchronisation üblicherweise vermeide.

Wenn Sie die Ausfallsicherheit Ihres Servers weiter erhöhen möchten empfiehlt sich die Verwendung einer „Hotspare“ Platte, sprich einer weiteren Festplatte, die beim Ausfall einer aktiven Platte sofort als Ersatz einspringt.

Formatieren Sie diese Platte genau so wie auch die beiden aktiven Platten. Die Erstellung eines RAID1 Verbundes mit Hotspare-„Partition“ sieht dann so aus:

```
Kommandozeile: mdadm --create /dev/md1 --level=1 --raid-devices=2 /dev/sda3
/dev/sdb3 --spare-device=1 /dev/sdc3
```

Dies müssen Sie selbstverständlich für alle RAID-Verbünde vornehmen.

# Konfiguration des Bootmanagers

***Achtung:*** *die nachfolgenden Erläuterungen beziehen sich noch auf Grub1 – bzw. Grub-Legacy wie er heute gerne genannt wird – und sind somit veraltet.*

Grundsätzlich kann der verbreitete Bootmanager „Grub“ nicht auf Software-RAID Devices zugreifen. Trotzdem ist es möglich mit einem RAID-Verbund für das /boot Verzeichnis zu arbeiten und daraus auch Vorteile zu ziehen. Voraussetzung dafür ist, dass es sich definitiv um einen Level 1 Verbund handelt.

Da Level 1 einer einfachen Spiegelung entspricht, ist gewährleitset, dass auf allen beteiligten Partitionen der gesamte Datenbestand des RAID-Verbundes vorhanden ist. Aus Sicht des Bootmanagers sind es dann einfach mehrere /boot Partitionen mit gleichem Inhalt, die er als reguläre Partition und nicht als RAID-Device anspricht.

Daraus lässt sich für Grub ein Fallback-Mechanismus realisieren. Sprich: Ist die /boot-Partition auf Platte 1 nicht ansprechbar, nimm die von Platte 2. Erst, wenn der Linux-Kernel geladen ist wird aus den einzelnen Partitionen ein RAID-Verbund, der etwa im Falle eines Kernel-Updates dafür sorgt, dass das neue Kernel-Image auf allen beteiligten Partitionen landet.

Voraussetzung dafür ist, dass alle beteiligten Partitionen für Grub auch als Start-Partitionen nutzbar sind.

Dafür, dass alle /boot-Partitionen der beteiligten Platten als „active“ gekennzeichnet sind wurde bereits beim Partitionieren Sorge getragen. Jetzt muss noch Grub in den Masterbootrecord aller beteiligten Platten installiert werden. Dies kann mit ***grub*** selbst erledigt werden.

```
Kommandozeile: grub
grub> root (hd0,0)
grub> setup (hd0)
grub> root (hd1,0)
grub> setup (hd1)
grub> quit
```

Damit steht die erste Stufe von grub in jedem MBR der vorhanden Festplatten. Dieser Code verweist lediglich auf die Partition der Festplatten auf der die zweite Stufe von ***grub*** installiert wurde; Die boot-Partition des Systems.

Jetzt gilt es noch den versprochenen Fallback-Mechanismus einzurichten. Fügen Sie in die Datei /boot/grub/menu.lst einfach folgende Zeile ein:

```
fallback 2
```

In der Grub Konfiguration wird üblicherweise auf einen default-Eintrag verwiesen. Die fallback Option definiert einen zweiten Bootmenü-Eintrag, der dann verwendet wird, wenn kein Zugriff auf den ersten möglich ist.

Nach einer openSUSE Standard-Installation sind im Grub-Menü üblicherweise zwei Einträge (default und failsafe) vorhanden. Verdoppeln Sie diese Menüeinträge einfach mit einem Editor Ihrer Wahl und ändern Sie in den neuen Einträgen die Festplattennummern so ab, dass diese auf die zweite Festplatte im System zeigen. So könnte dies aussehen:

```
#0
###Don't change this comment - YaST2 identifier: Original name: linux###
title openSUSE 11.1
    kernel (hd0,0)/vmlinuz root=/dev/system/Root resume=/dev/system/Swap
splash=silent showopts vga=0x317
    initrd (hd0,0)/initrd
#1
###Don't change this comment - YaST2 identifier: Original name: failsafe###
title Failsafe -- openSUSE 11.1
    kernel (hd0,0)/vmlinuz root=/dev/system/Root showopts ide=nodma apm=off
acpi=off noresume nosmp noapic maxcpus=0 edd=off x11failsafe vga=0x317
    initrd (hd0,0)/initrd
#2
###Don't change this comment - YaST2 identifier: Original name: linux###
title openSUSE 11.1 Fallback
    kernel (hd1,0)/vmlinuz root=/dev/system/Root resume=/dev/system/Swap
splash=silent showopts vga=0x317
    initrd (hd1,0)/initrd
#3
###Don't change this comment - YaST2 identifier: Original name: failsafe###
title Failsafe -- openSUSE 11.1 Fallback
    kernel (hd1,0)/vmlinuz root=/dev/system/Root showopts ide=nodma apm=off
acpi=off noresume nosmp noapic maxcpus=0 edd=off x11failsafe vga=0x317
    initrd (hd1,0)/initrd
```

Grub nummeriert die Einträge beginnend mit 0 durch, bedeutet, dass „fallback 2“ auf den dritten Eintrag und somit auf die zweite Platte im System verweisst.

Wer das Setup-Script nutzt muss sich um die Grub-Konfiguration keine Gedanken machen, sie wird automatisch erledigt. Nachteilig an der Lösung ist, dass YaST nach jedem Kernel-Update neue Einträge, dummerweise am Anfang der Konfiguration hinzufügt. Löschen Sie diese einfach und alles funktioniert wie gehabt.

From:
https://wiki.invis-server.org/ - **invis-server.org**

Permanent link:
**https://wiki.invis-server.org/doku.php?id=invis_server_wiki:installation:diskprep&rev=1488283057**

Last update: **2017/02/28 11:57**