

Redundante Server mit freier Software

LiHAS – LinuxHaus Stuttgart
Adrian Reyer
are@lihas.de

Problematik

Ein Hardwareausfall bedeutet zunächst einmal
Datenausfall und damit einen Zeitausfall.

Backups schützen nur vor dem endgültigen
Datenverlust, die Daten müssen daraus aber zuerst
wieder hergestellt werden

klassische Server

- redundante Netzteile
- RAID-Systeme
- Unterbrechungsfreie Stromversorgungen

Problem: nicht alles ist redundant, es gibt genügend einzelne mögliche Fehlerpunkte

Lösungsansatz

- komplette Hardware doppelt vorhalten
- Daten auf beiden Systemen vorhalten
 - rsync / unison: klassisch, zeitverzögert, eher Backup als Redundanz
 - Shared Storage (SAN, Netzwerkdateisysteme): wieder einzelne Komponente
 - drbd: vergleichbar RAID-1 über Netzwerk

Folgeproblematik

- Wer führt die Umschaltung durch?
- Wer konfiguriert Dienste um?
- Wer passt Start-/Stoppskripte an die Clusterregeln an?
- Wie können Updates eingespielt werden ohne alles wieder von vorne konfigurieren zu müssen?

Umschaltung

- Wer führt die Umschaltung durch?
 - manuell: langsam
 - automatisch mit heartbeat: schnell, Fehlerkennungen eines Ausfalls möglich.
 - Verbindungen zwischen Rechnern mehrfach redundant auslegen.
 - Möglichst den anderen Server im Fall einer Übernahme abschalten lassen

Umkonfiguration

- zusätzliche Startskripte für Clusterdienste und Erweiterung bestehender Skripte / Konfigurationsdateien um die für den Clusterbetrieb notwendigen Unterscheidungen
 - mühsam und fehleranfällig
 - nach einem Update ist potentiell wieder alles kaputt
- besser: virtuelle Server als Dienst

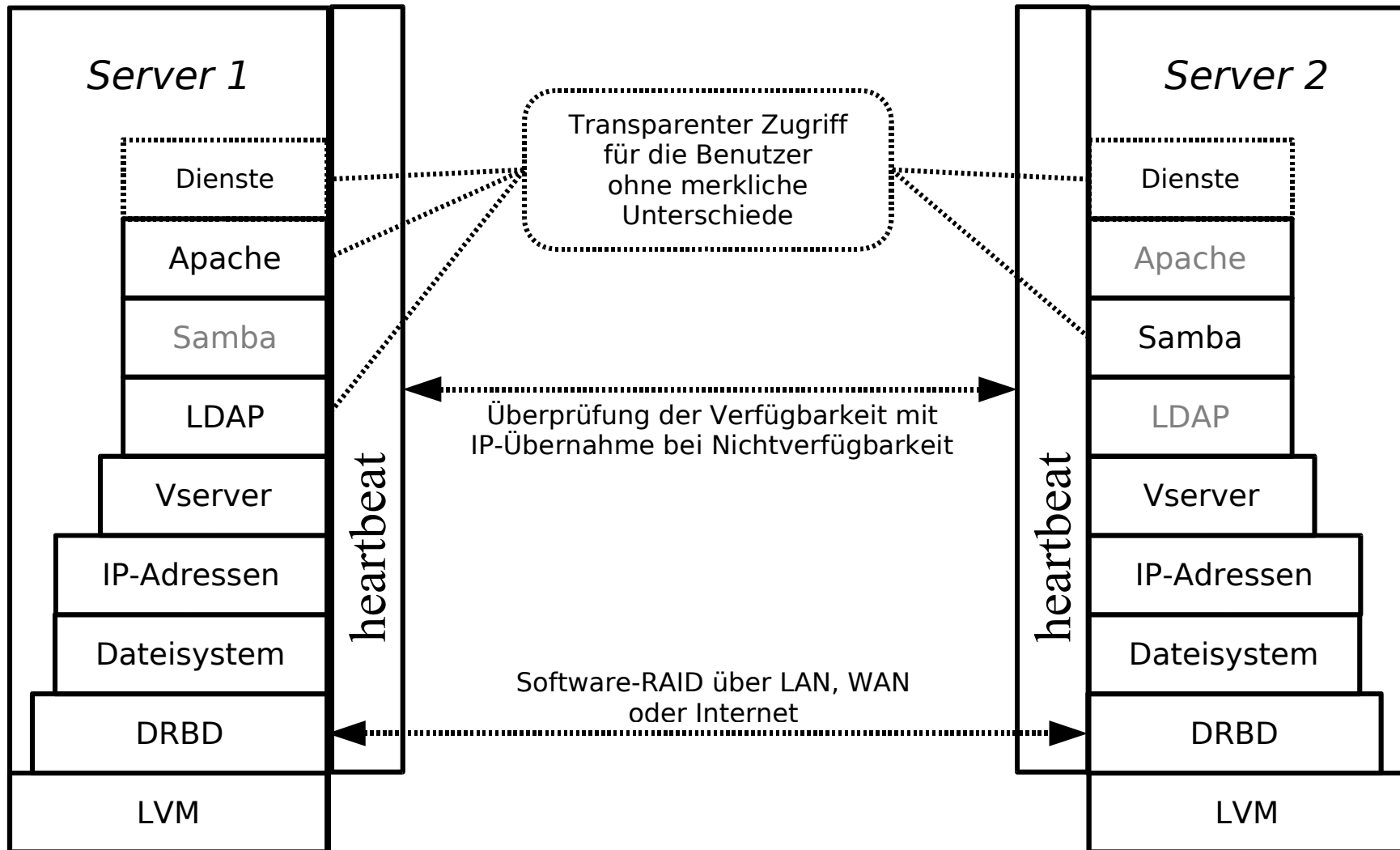
Virtuelle Server – vserver

- vergleichbar chroot, aber ausbruchsicher
- geringer Ressourcenverbrauch (~0.5% gegenüber Standardserver)
- Möglichkeit einen vserver in den Ressourcen einzuschränken
- vserver kommt komplett mit dem eigentlichen Dienst und den Helferdiensten wie z.B. cron, MTA, logrotate

Virtuelle Server – vserver (2)

- hat nur bestimmte vom Hauptserver vorgegebene Rechte
- Hardwareunabhängig (nicht Architekturunabhängig)
- Im vserver sind keine Kenntnisse über den Cluster notwendig
- Abzüglich der Hardware administrierbar wie ein Einzelserver

Aufbau



LVM

- dynamische Verwaltung von Festplattenplatz, auch im laufenden Betrieb
- keine (relevante) Beschränkung der Anzahl der Partitionen

DRBD

- vereinfacht: Netzwerk RAID 1
- schreibende Anwendung bekommt erst dann das OK, wenn beide Seiten die Daten geschrieben haben.
- Synchronisationsrichtung kann einfach gedreht werden.
- <http://www.drbd.org/>

IP Aliase

- Die Hardware im Cluster braucht eine IP
- In den meisten Faellen sind die redundanten Dienste Netzwerkdienste
 - sie brauchen damit eigene IPs
 - Switche müssen IPs mit mehreren MAC-Adressen auf verschiedenen Ports unterstützen

linux-vserver

- Virtualisierungslösung auf dem gleichen Kernel wie das Hardwaresystem
- Benutzung vergleichbar chroot() - im Inneren kann ein fast normales Linuxsystem administriert werden
- Berechtigungen Systemeigenschaften im Kernel zu ändern sind üblicherweise aus (Netzwerkkonfiguration, Mounts)
- <http://www.linux-vserver.org/>

heartbeat

- überprüft die Erreichbarkeit der anderen Maschine
- schaltet Dienste an und aus
- Dienste:
 - IPs
 - Richtung der DRBD-Synchronisation
 - vservers
- <http://www.linux-ha.org/>