

# Verzeichnisse synchronisieren

Stefan Schumacher

`www.kaishakunin.com`

Chemnitzer Linux-Tage 2009

解  
积  
人

# Über mich

- freier Unternehmensberater (Kaishakunin.com) mit Schwerpunkt auf Social Engineering, Security Awareness, Counter Intelligence und Security Management
- Student der Bildungswissenschaft und Psychologie an der Uni Magdeburg
- Chefororganisator des Magdeburger Open-Source-Tages
- NetBSD-Entwickler (Stefan@NetBSD.org)  
Vorsitzender des NetBSD Deutschland e.V.
- Sysadmin in verschiedenen Netzen

解  
积  
人

# Table of Contents

- 1 Grundlagen
- 2 Kopieren
- 3 Netzwerk
- 4 Backup
- 5 Synch in mehrere Richtungen
- 6 Repository

解  
积  
人

# Table of Contents

- 1 Grundlagen
- 2 Kopieren
- 3 Netzwerk
- 4 Backup
- 5 Synch in mehrere Richtungen
- 6 Repository

解  
积  
人

# Warum Verzeichnisse synchronisieren

- Datenhaltung auf mehreren unabhängigen Geräten
- Desktop/Laptop/Netbook/Handy/PDA/Server
- Arbeit in Gruppen  
(Entwicklung/Hausarbeiten/Bücher/Organisationskomitee)
- Synchronisation der Daten via Netzwerk auf verschiedenen Rechnern

解  
积  
人

# Synchronisationsmethoden

- Dateiinhalt und Metadaten verteilen (ACL, File Flags, ExtAttributes)
- wichtig: Benutzer- und Zeitstempel (wer hat wann was geändert, was ist aktuell?)

解  
积  
人

# Table of Contents

- 1 Grundlagen
- 2 Kopieren**
- 3 Netzwerk
- 4 Backup
- 5 Synch in mehrere Richtungen
- 6 Repository

# cp/rcp/scp

Server/Client; 1:1

- cp kopiert Dateien vom Quellverzeichnis ins Zielverzeichnis
- -r: rekursiv; -f erzwungen
- rcp: im Netzwerk, aber unsicher
- scp: Teil von OpenSSH, kopiert im Netz, aber verschlüsselt
- Authentifikation via Schlüssel möglich (Skripte)
- Windows-Client: WinSCP
- cp/rcp/scp haben Probleme mit löchrigen Dateien, langen Dateinamen, Umlauten/UTF8

解  
积  
人

# dump/restore

- **Das** Backupwerkzeug in der BSD/Solaris-Welt
- arbeitet auf Inode-Ebene, unterhalb des Dateisystems
- extrem stabil, z.B. Plattenmigration
- leider dateisystemabhängig

# Rdist

## 1:n im Netzwerk (Stern)

- altes BSD-Tool
- verteilt Verzeichnisse im Netz
- Bsp: /etc/ im Klassenraum ausrollen
- distfile beinhaltet die Optionen (Quelle/Ziel)
- traditionelles Rdist ist unsicher, rdist6 ist besser

# Table of Contents

- 1 Grundlagen
- 2 Kopieren
- 3 Netzwerk**
- 4 Backup
- 5 Synch in mehrere Richtungen
- 6 Repository

# Rsync

- synchronisiert Verzeichnishierarchien auf Platten und im Netz
- verschlüsselt via SSH
- synced nur geänderte Dateien
- kann mit RegEx Dateien/Verzeichnisse ein-/ausschließen
- kann mit Prüfsummen Ergebnisse kontrollieren (!)
- cwRsync für Windows; MacOS X hat Rsync

```
rsync -exclude T[e]mp* --delete-after -v -r  
/home/www/MOST-WWW/  
stefan@192.168.1.1:/home/www/most/
```

# Rsync als Rdist-Ersatz

- Rsync in sternförmigem Netz kompliziert zu nutzen
- Ssync »emuliert« Rdist-Verhalten mit Rsync
- nutzt SSH, besser als Rdist

# Rsync und Backups

- zur Sicherung einfach die Daten auf einen anderen Rechner/ein anderes Medium kopieren
- `rsync -a /home /mnt/usbplatte/backup`
- Problem: `/brief.txt` kaputt, Rsync läuft
- Idee: Archiv

# Rsync und Backups

- zur Sicherung einfach die Daten auf einen anderen Rechner/ein anderes Medium kopieren
- `rsync -a /home /mnt/usbplatte/backup`
- Problem: `/brief.txt` kaputt, Rsync läuft
- Idee: Archiv
- ```
$heute=`date +%y%m%d%H%M`;  
mkdir /mnt/usbplatte/backup-$heute
```
- `rsync -a /home /mnt/usbplatte/backup-$heute`
- Problem: Platzverschwendung

# Rsync und Backups

- zur Sicherung einfach die Daten auf einen anderen Rechner/ein anderes Medium kopieren
- `rsync -a /home /mnt/usbplatte/backup`
- Problem: `/brief.txt` kaputt, Rsync läuft
- Idee: Archiv
- `$heute='date +%y%m%d%H%M';  
mkdir /mnt/usbplatte/backup-$heute`
- `rsync -a /home /mnt/usbplatte/backup-$heute`
- Problem: Platzverschwendung

# Table of Contents

1 Grundlagen

2 Kopieren

3 Netzwerk

**4 Backup**

5 Synch in mehrere Richtungen

6 Repository

# Rdiff-Backup

- Idee: Rsync und CVS kreuzen == Rdiff-Backup
- Rsync-Bibliotheken um Diffs erweitert
- im Repository liegen alle geänderten Versionen der Dateien vor

# Rdiff-backup

```
rdiff-backup /etc/ /usr/back/etc/
```

```
rdiff-backup -l /usr/back/etc/
```

```
Found 6 increments:
```

```
increments.2006-01-01T12:35:24+01:00.dir Sun Jan 1 12:35:24 2006
```

```
increments.2006-01-04T12:35:37+01:00.dir Wed Jan 4 12:35:37  
2006
```

```
increments.2006-01-28T12:35:31+01:00.dir Sat Jan 28 12:35:31  
2006
```

```
increments.2006-02-03T12:35:28+01:00.dir Fri Feb 3 12:35:28 2006
```

```
increments.2006-02-04T12:35:50+01:00.dir Sat Feb 4 12:35:50 2006
```

```
increments.2006-02-06T12:35:27+01:00.dir Mon Feb 6 12:35:27  
2006
```

```
Current mirror: Sun Feb 26 12:35:29 2006
```

```
rdiff-backup -r 2B -force /usr/back/etc/ /etc/
```

# Rdiff-backup

- `--compare-at-time Zeitstempel` Vergleicht das Quell- mit dem Zielverzeichnis, dabei wird das Zielverzeichnis passend zum Zeitstempel ausgewählt. Man erhält eine Übersicht, welche Dateien verändert wurden.
- `--exclude-regexp RegEx` Lässt getroffene Verzeichnisse und Dateien aus.
- `-r Zeitstempel` Restaurierung der Daten mit dem Zustand, der an *Zeitstempel* vorlag.
- `--remove-older-than Zeitstempel` Entfernt alle Sicherungen, die älter als *Zeitstempel* sind. Mehrere Sicherungen müssen mit `-force` entfernt werden.

解  
釈  
人

# Table of Contents

- 1 Grundlagen
- 2 Kopieren
- 3 Netzwerk
- 4 Backup
- 5 Synch in mehrere Richtungen**
- 6 Repository

解  
积  
人

# Unison

- Rsync arbeitet im Server-/Client-Modus, Änderungen im Client werden ignoriert oder überschrieben
- Unison synchronisiert und führt zwei Verzeichnisse zusammen
- nutzt dazu Profile z.B. eins für C eins für  $\text{\LaTeX}$
- bindet externe Programme zum mergen ein (vim ...)
- kann Dateien vor dem merge sichern  
Dateiname..laufendeNummer..unison.bak
- `merge = <PFADSPEZIFIKATION> ->`  
`<EXTRERNES_PROGRAMM>`

解  
積  
人

# Probleme

- erkennt Differenzen anhand der Inode-Nr und Modtime  
kann mit (re)touch hintertrieben werden
- Zugriffsrechte (außer SGID/SUID) werden  
mitsynchronisiert  
umask auf 022 setzen
- Unison ignoriert Hardlinks
- Umbenennung ignoriertes Verzeichniss kann zu deren  
Löschung führen
- es gibt keine Man-Page, `unison -doc` aber eine GUI :-)

解  
釈  
人

# Prüfen

- Vertrauen ist gut, Kontrolle ist besser ...
- mtree/aide/tripwire als Dateisystemintegritätsprüfer
- `find /home -printx | xargs md5`

# Table of Contents

- 1 Grundlagen
- 2 Kopieren
- 3 Netzwerk
- 4 Backup
- 5 Synch in mehrere Richtungen
- 6 Repository**

解  
积  
人

# Verteiltes Arbeiten

- Synchronisation kann komplex werden, gerade in Arbeitsgruppen
- Von Anfang an Revisionssoftware verwenden
- Bsp: CVS
- legt zentrales Repository an
- synchronisiert Clients mit dem Repository, kann Dateien sperren und Metadaten verwalten
- Nachteil: Zugang zum Server
- andere Programme: Subversion (SVN)

解  
积  
人

# verteilte Repository

- GIT, Monotone, GNU Arch etc. pp.
- ermöglichen Offline-Arbeit an den Repositories
- synchronisieren dann mit dem Server, wenn wieder Netzwerk da ist
- CVS + Unison ;-)

解  
积  
人

Fragen?  
Stefan.Schumacher@Kaishakunin.com

Folien: [www.Kaishakunin.com](http://www.Kaishakunin.com)

解  
积  
人