

# Backups mit rsync

Bernhard Trummer  
bernhard.trummer@gmx.at

26. April 2012

# Überblick

# Über mich

- TU / Telematik (1994 – 2001)
- Linux User Group Graz
- Angestellt bei Infonova (seit 2000)
- Technology Architekt, Bereich Infrastruktur und Testing

# Inhalt

- Wozu Backups?
- Was ist rsync?
- Beispiele
- Verschlüsselung
- Fokus: Heim-Anwender, Linux Ein-/Umsteiger

# Inhalt

- Wozu Backups?
- Was ist rsync?
- Beispiele
- Verschlüsselung
- Fokus: Heim-Anwender, Linux Ein-/Umsteiger

# Inhalt

- Wozu Backups?
- Was ist rsync?
- Beispiele
- Verschlüsselung
- Fokus: Heim-Anwender, Linux Ein-/Umsteiger

# Inhalt

- Wozu Backups?
- Was ist rsync?
- Beispiele
- Verschlüsselung
- Fokus: Heim-Anwender, Linux Ein-/Umsteiger

# Inhalt

- Wozu Backups?
- Was ist rsync?
- Beispiele
- Verschlüsselung
- Fokus: Heim-Anwender, Linux Ein-/Umsteiger

# Wozu Backups?

# Wozu Backups?

- Wikipedia: ... bezeichnet das Kopieren von Daten in der Absicht, diese im Fall eines Datenverlustes zurückkopieren zu können.
- Ursachen fuer Datenverlust:
  - Fehlbedienung
  - Hardwaredefekte
  - Äußere Einwirkungen

# Wozu Backups?

- Wikipedia: ... bezeichnet das Kopieren von Daten in der Absicht, diese im Fall eines Datenverlustes zurückkopieren zu können.
- Ursachen fuer Datenverlust:
  - Fehlbedienung
  - Hardwaredefekte
  - Äußere Einwirkungen

# Wozu Backups?

- Wikipedia: ... bezeichnet das Kopieren von Daten in der Absicht, diese im Fall eines Datenverlustes zurückkopieren zu können.
- Ursachen fuer Datenverlust:
  - Fehlbedienung
  - Hardwaredefekte
  - Äußere Einwirkungen

# Wozu Backups?

- Wikipedia: ... bezeichnet das Kopieren von Daten in der Absicht, diese im Fall eines Datenverlustes zurückkopieren zu können.
- Ursachen fuer Datenverlust:
  - Fehlbedienung
  - Hardwaredefekte
  - Äußere Einwirkungen

# Wozu Backups?

- Wikipedia: ... bezeichnet das Kopieren von Daten in der Absicht, diese im Fall eines Datenverlustes zurückkopieren zu können.
- Ursachen fuer Datenverlust:
  - Fehlbedienung
  - Hardwaredefekte
  - Äußere Einwirkungen

# Speichermedien

- USB-Sticks
- CDs und DVDs
- Externe Festplatten (USB, sATA)
- Streamerbänder
- Online-Festplatten

# Speichermedien

- USB-Sticks
- CDs und DVDs
- Externe Festplatten (USB, sATA)
- Streamerbänder
- Online-Festplatten

# Speichermedien

- USB-Sticks
- CDs und DVDs
- Externe Festplatten (USB, sATA)
- Streamerbänder
- Online-Festplatten

# Speichermedien

- USB-Sticks
- CDs und DVDs
- Externe Festplatten (USB, sATA)
- Streamerbänder
- Online-Festplatten

# Speichermedien

- USB-Sticks
- CDs und DVDs
- Externe Festplatten (USB, sATA)
- Streamerbänder
- Online-Festplatten

# Meine Erfahrungen

- Lesefehler bei gebrannten CDs nach weniger als 4 Jahren.
- Festplattendefekte können frühzeitig erkannt werden (Klappergeräusche, smartmontools).
- Festplatte können aber auch ohne Vorwarnung komplett ausfallen.
- Ein RAID-System ersetzt keine Backups!

# Meine Erfahrungen

- Lesefehler bei gebrannten CDs nach weniger als 4 Jahren.
- Festplattendefekte können frühzeitig erkannt werden (Klappergeräusche, smartmontools).
- Festplatte können aber auch ohne Vorwarnung komplett ausfallen.
- Ein RAID-System ersetzt keine Backups!

# Meine Erfahrungen

- Lesefehler bei gebrannten CDs nach weniger als 4 Jahren.
- Festplattendefekte können frühzeitig erkannt werden (Klappergeräusche, smartmontools).
- Festplatte können aber auch ohne Vorwarnung komplett ausfallen.
- Ein RAID-System ersetzt keine Backups!

## Meine Erfahrungen

- Lesefehler bei gebrannten CDs nach weniger als 4 Jahren.
- Festplattendefekte können frühzeitig erkannt werden (Klappergeräusche, smartmontools).
- Festplatte können aber auch ohne Vorwarnung komplett ausfallen.
- Ein RAID-System ersetzt keine Backups!

## Beispiel von mir

- 1,5 TB Systemfestplatte (davon 675GB belegt)
- Alle Daten die ich habe sind hier drauf.
- Zu sichernde Daten:
  - 50 GB MP3s bzw. OGGs
  - 20 GB Fotos
  - 400 GB Videos
  - 50 GB sonstiges
- In solchen Fällen sind (externe) Festplatten ideal.

## Beispiel von mir

- 1,5 TB Systemfestplatte (davon 675GB belegt)
- Alle Daten die ich habe sind hier drauf.
- Zu sichernde Daten:
  - 50 GB MP3s bzw. OGGs
  - 20 GB Fotos
  - 400 GB Videos
  - 50 GB sonstiges
- In solchen Fällen sind (externe) Festplatten ideal.

## Beispiel von mir

- 1,5 TB Systemfestplatte (davon 675GB belegt)
- Alle Daten die ich habe sind hier drauf.
- Zu sichernde Daten:
  - 50 GB MP3s bzw. OGGs
  - 20 GB Fotos
  - 400 GB Videos
  - 50 GB sonstiges
- In solchen Fällen sind (externe) Festplatten ideal.

## Beispiel von mir

- 1,5 TB Systemfestplatte (davon 675GB belegt)
- Alle Daten die ich habe sind hier drauf.
- Zu sichernde Daten:
  - 50 GB MP3s bzw. OGGs
  - 20 GB Fotos
  - 400 GB Videos
  - 50 GB sonstiges
- In solchen Fällen sind (externe) Festplatten ideal.

## Beispiel von mir

- 1,5 TB Systemfestplatte (davon 675GB belegt)
- Alle Daten die ich habe sind hier drauf.
- Zu sichernde Daten:
  - 50 GB MP3s bzw. OGGs
  - 20 GB Fotos
  - 400 GB Videos
  - 50 GB sonstiges
- In solchen Fällen sind (externe) Festplatten ideal.

## Beispiel von mir

- 1,5 TB Systemfestplatte (davon 675GB belegt)
- Alle Daten die ich habe sind hier drauf.
- Zu sichernde Daten:
  - 50 GB MP3s bzw. OGGs
  - 20 GB Fotos
  - 400 GB Videos
  - 50 GB sonstiges
- In solchen Fällen sind (externe) Festplatten ideal.

## Beispiel von mir

- 1,5 TB Systemfestplatte (davon 675GB belegt)
- Alle Daten die ich habe sind hier drauf.
- Zu sichernde Daten:
  - 50 GB MP3s bzw. OGGs
  - 20 GB Fotos
  - 400 GB Videos
  - 50 GB sonstiges
- In solchen Fällen sind (externe) Festplatten ideal.

## Beispiel von mir

- 1,5 TB Systemfestplatte (davon 675GB belegt)
- Alle Daten die ich habe sind hier drauf.
- Zu sichernde Daten:
  - 50 GB MP3s bzw. OGGs
  - 20 GB Fotos
  - 400 GB Videos
  - 50 GB sonstiges
- In solchen Fällen sind (externe) Festplatten ideal.

# Meine Backupmedien

- 1 TB USB-Festplatte
- 400 GB Festplatte (\*)
- 240 GB Festplatte (\*)
- (\*) waren meine vorigen Systemfestplatten

# Meine Backupmedien

- 1 TB USB-Festplatte
- 400 GB Festplatte (\*)
- 240 GB Festplatte (\*)
- (\*) waren meine vorigen Systemfestplatten

# Meine Backupmedien

- 1 TB USB-Festplatte
- 400 GB Festplatte (\*)
- 240 GB Festplatte (\*)
- (\*) waren meine vorigen Systemfestplatten

# Meine Backupmedien

- 1 TB USB-Festplatte
- 400 GB Festplatte (\*)
- 240 GB Festplatte (\*)
- (\*) waren meine vorigen Systemfestplatten

# Backup-Tools

- mit KDE/Gnome Frontend:
  - kbackup
  - backintime
  - ...
- Commandline:
  - rsnapshot
  - duplicity
  - ...

# Backup-Tools

- mit KDE/Gnome Frontend:
  - kbackup
  - backintime
  - ...
- Commandline:
  - rsnapshot
  - duplicity
  - ...

# Backup-Tools

- mit KDE/Gnome Frontend:
  - kbackup
  - backintime
  - ...
- Commandline:
  - rsnapshot
  - duplicity
  - ...

# Backup-Tools

- mit KDE/Gnome Frontend:
  - kbackup
  - backintime
  - ...
- Commandline:
  - rsnapshot
  - duplicity
  - ...

# Backup-Tools

- mit KDE/Gnome Frontend:
  - kbackup
  - backintime
  - ...
- Commandline:
  - rsnapshot
  - duplicity
  - ...

# Backup-Tools

- mit KDE/Gnome Frontend:
  - kbackup
  - backintime
  - ...
- Commandline:
  - rsnapshot
  - duplicity
  - ...

# Backup-Tools

- mit KDE/Gnome Frontend:
  - kbackup
  - backintime
  - ...
- Commandline:
  - rsnapshot
  - duplicity
  - ...

# Backup-Tools

- mit KDE/Gnome Frontend:
  - kbackup
  - backintime
  - ...
- Commandline:
  - rsnapshot
  - duplicity
  - ...

# rsync

# rsync

a fast, versatile, remote (and local) file-copying tool

# Einfaches Beispiel

```
rsync -av src/ dst/
```

- -a ... archive mode (-rlptgoD)
- -v ... verbose
- src/ ... Quellverzeichnis
- dst/ ... Zielverzeichnis

# Einfaches Beispiel

```
rsync -av src/ dst/
```

- -a ... archive mode (-rlptgoD)
- -v ... verbose
- src/ ... Quellverzeichnis
- dst/ ... Zielverzeichnis

# Einfaches Beispiel

```
rsync -av src/ dst/
```

- -a ... archive mode (-rlptgoD)
- -v ... verbose
- src/ ... Quellverzeichnis
- dst/ ... Zielverzeichnis

# Einfaches Beispiel

```
rsync -av src/ dst/
```

- -a ... archive mode (-rlptgoD)
- -v ... verbose
- src/ ... Quellverzeichnis
- dst/ ... Zielverzeichnis

# Einfaches Beispiel

```
rsync -av src/ dst/
```

- -a ... archive mode (-rlptgoD)
- -v ... verbose
- src/ ... Quellverzeichnis
- dst/ ... Zielverzeichnis

# Einfaches Beispiel

```
rsync -av --delete src/ dst/
```

- Wie voriges Beispiel, aber
- Ohne `--delete` bleiben Dateien in `dst/` erhalten, die in `src/` nicht mehr existieren (weil sie umbenannt, verschoben oder gelöscht worden sind)

## Einfaches Beispiel

```
rsync -av --delete src/ dst/
```

- Wie voriges Beispiel, aber
- Ohne `--delete` bleiben Dateien in `dst/` erhalten, die in `src/` nicht mehr existieren (weil sie umbenannt, verschoben oder gelöscht worden sind)

## Einfaches Beispiel

```
rsync -av --delete src/ dst/
```

- Wie voriges Beispiel, aber
- Ohne `--delete` bleiben Dateien in `dst/` erhalten, die in `src/` nicht mehr existieren (weil sie umbenannt, verschoben oder gelöscht worden sind)

# rsync über Netzwerk

- Push: `rsync -av --delete src/ user@hostname:dst/`
- Pull: `rsync -av --delete user@hostname:src/ dst/`

# rsync über Netzwerk

- Push: `rsync -av --delete src/ user@hostname:dst/`
- Pull: `rsync -av --delete user@hostname:src/ dst/`

# Demonstration

- `rsync -av src/ dst/`
- `rsync -av --delete src/ dst/`
- `rsync -av --delete src/ user@hostname:dst/`

# rsync des ganzen Filesystems

- rsync-Aufruf mit / statt src/
- Gewisse Verzeichnisse sollten/müssen weglassen werden:
  - /tmp, /proc (kcore)
  - Mount-Point von Backupmedium (z.B. /mnt)

# rsync des ganzen Filesystems

- rsync-Aufruf mit / statt src/
- Gewisse Verzeichnisse sollten/müssen weglassen werden:
  - /tmp, /proc (kcore)
  - Mount-Point von Backupmedium (z.B. /mnt)

# rsync des ganzen Filesystems

- rsync-Aufruf mit / statt src/
- Gewisse Verzeichnisse sollten/müssen weglassen werden:
  - /tmp, /proc (kcore)
  - Mount-Point von Backupmedium (z.B. /mnt)

# rsync des ganzen Filesystems

- rsync-Aufruf mit / statt src/
- Gewisse Verzeichnisse sollten/müssen weglassen werden:
  - /tmp, /proc (kcore)
  - Mount-Point von Backupmedium (z.B. /mnt)

# Filterregeln

- `--include-from=my.filter`
- `--delete-excluded`

# Filterregeln

- `--include-from=my.filter`
- `--delete-excluded`

## Filter-Syntax (man page: FILTER RULES)

- Format: RULE [PATTERN\_OR\_FILENAME]
- RULE: z.B. + oder -
- PATTERN: kann Wildcards enthalten (z.B. \*, \*\*, ?)

## Filter-Syntax (man page: FILTER RULES)

- Format: RULE [PATTERN\_OR\_FILENAME]
- RULE: z.B. + oder -
- PATTERN: kann Wildcards enthalten (z.B. \*, \*\*, ?)

## Filter-Syntax (man page: FILTER RULES)

- Format: RULE [PATTERN\_OR\_FILENAME]
- RULE: z.B. + oder -
- PATTERN: kann Wildcards enthalten (z.B. \*, \*\*, ?)

## Beispiele aus der man page

```
+ /some/path/this-file-will-not-be-found  
+ /file-is-included  
- *
```

- Warum wird das erste File nicht gefunden?
- Weil rsync den Verzeichnisbaum rekursiv abarbeitet und jedes einzelne Unterverzeichnis in den Filterregeln nachschlägt.

## Beispiele aus der man page

```
+ /some/path/this-file-will-not-be-found  
+ /file-is-included  
- *
```

- Warum wird das erste File nicht gefunden?
- Weil rsync den Verzeichnisbaum rekursiv abarbeitet und jedes einzelne Unterverzeichnis in den Filterregeln nachschlägt.

## Beispiele aus der man page

```
+ /some/path/this-file-will-not-be-found  
+ /file-is-included  
- *
```

- Warum wird das erste File nicht gefunden?
- Weil rsync den Verzeichnisbaum rekursiv abarbeitet und jedes einzelne Unterverzeichnis in den Filterregeln nachschlägt.

## Beispiele aus der man page

```
+ /some/  
+ /some/path/  
+ /some/path/this-file-is-found  
+ /file-also-included  
- *
```

## Praxisbeispiel

```
+ /etc/  
+ /etc/**  
  
+ /home/  
+ /home/slash/  
- /home/slash/.thumbnails/  
+ /home/slash/**  
  
+ /usr/  
+ /usr/local/  
+ /usr/local/**  
  
- *
```

# Snapshots

- wenn man auf den Datenstand eines älteren Backups zurückgreifen will.
- d.h. jedes Backup liegt in einem eigenen Zielverzeichnis.
- `--compare-dest=DIR`
- `--copy-dest=DIR`
- `--link-dest=DIR`

# Snapshots

- wenn man auf den Datenstand eines älteren Backups zurückgreifen will.
- d.h. jedes Backup liegt in einem eigenen Zielverzeichnis.
- `--compare-dest=DIR`
- `--copy-dest=DIR`
- `--link-dest=DIR`

# Snapshots

- wenn man auf den Datenstand eines älteren Backups zurückgreifen will.
- d.h. jedes Backup liegt in einem eigenen Zielverzeichnis.
- `--compare-dest=DIR`
- `--copy-dest=DIR`
- `--link-dest=DIR`

# Snapshots

- wenn man auf den Datenstand eines älteren Backups zurückgreifen will.
- d.h. jedes Backup liegt in einem eigenen Zielverzeichnis.
- `--compare-dest=DIR`
- `--copy-dest=DIR`
- `--link-dest=DIR`

# Snapshots

- wenn man auf den Datenstand eines älteren Backups zurückgreifen will.
- d.h. jedes Backup liegt in einem eigenen Zielverzeichnis.
- `--compare-dest=DIR`
- `--copy-dest=DIR`
- `--link-dest=DIR`

# Demonstration

- `rsync -av src/ dst1/`
- `rsync -av src/ dst2/ --link-dest=../dst1/`
- `rsync -av src/ dst3/ --link-dest=../dst2/`

# Sonstiges

# Verschlüsselung

- Interessant für USB-Medien (z.B. Verlust, Diebstahl)
- Oder Daten auf Online-Festplatten
- wichtig: sichere Passphrase
- und: Passphrase nicht vergessen!

# Verschlüsselung

- Interessant für USB-Medien (z.B. Verlust, Diebstahl)
- Oder Daten auf Online-Festplatten
- wichtig: sichere Passphrase
- und: Passphrase nicht vergessen!

# Verschlüsselung

- Interessant für USB-Medien (z.B. Verlust, Diebstahl)
- Oder Daten auf Online-Festplatten
- wichtig: sichere Passphrase
- und: Passphrase nicht vergessen!

# Verschlüsselung

- Interessant für USB-Medien (z.B. Verlust, Diebstahl)
- Oder Daten auf Online-Festplatten
- wichtig: sichere Passphrase
- und: Passphrase nicht vergessen!

# LUKS

- einmalig:

- `cryptsetup -c aes -s 256 luksFormat /dev/sdX`
- `cryptsetup luksOpen /dev/sdX my-disk`
- `mke2fs ... /dev/mapper/my-disk`

- danach:

- `cryptsetup luksOpen /dev/sdX my-disk`
- `mount /dev/mapper/my-disk /mnt`
- `rsync ...`
- `umount /dev/mapper/my-disk`
- `cryptsetup luksClose my-disk`

# LUKS

- einmalig:
  - `cryptsetup -c aes -s 256 luksFormat /dev/sdX`
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mke2fs ... /dev/mapper/my-disk`
- danach:
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mount /dev/mapper/my-disk /mnt`
  - `rsync ...`
  - `umount /dev/mapper/my-disk`
  - `cryptsetup luksClose my-disk`

# LUKS

- einmalig:
  - `cryptsetup -c aes -s 256 luksFormat /dev/sdX`
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mke2fs ... /dev/mapper/my-disk`
- danach:
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mount /dev/mapper/my-disk /mnt`
  - `rsync ...`
  - `umount /dev/mapper/my-disk`
  - `cryptsetup luksClose my-disk`

# LUKS

- einmalig:
  - `cryptsetup -c aes -s 256 luksFormat /dev/sdX`
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mke2fs ... /dev/mapper/my-disk`
- danach:
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mount /dev/mapper/my-disk /mnt`
  - `rsync ...`
  - `umount /dev/mapper/my-disk`
  - `cryptsetup luksClose my-disk`

# LUKS

- einmalig:
  - `cryptsetup -c aes -s 256 luksFormat /dev/sdX`
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mke2fs ... /dev/mapper/my-disk`
- danach:
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mount /dev/mapper/my-disk /mnt`
  - `rsync ...`
  - `umount /dev/mapper/my-disk`
  - `cryptsetup luksClose my-disk`

# LUKS

- einmalig:
  - `cryptsetup -c aes -s 256 luksFormat /dev/sdX`
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mke2fs ... /dev/mapper/my-disk`
- danach:
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mount /dev/mapper/my-disk /mnt`
  - `rsync ...`
  - `umount /dev/mapper/my-disk`
  - `cryptsetup luksClose my-disk`

# LUKS

- einmalig:
  - `cryptsetup -c aes -s 256 luksFormat /dev/sdX`
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mke2fs ... /dev/mapper/my-disk`
- danach:
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mount /dev/mapper/my-disk /mnt`
  - `rsync ...`
  - `umount /dev/mapper/my-disk`
  - `cryptsetup luksClose my-disk`

# LUKS

- einmalig:
  - `cryptsetup -c aes -s 256 luksFormat /dev/sdX`
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mke2fs ... /dev/mapper/my-disk`
- danach:
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mount /dev/mapper/my-disk /mnt`
  - `rsync ...`
  - `umount /dev/mapper/my-disk`
  - `cryptsetup luksClose my-disk`

# LUKS

- einmalig:
  - `cryptsetup -c aes -s 256 luksFormat /dev/sdX`
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mke2fs ... /dev/mapper/my-disk`
- danach:
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mount /dev/mapper/my-disk /mnt`
  - `rsync ...`
  - `umount /dev/mapper/my-disk`
  - `cryptsetup luksClose my-disk`

# LUKS

- einmalig:
  - `cryptsetup -c aes -s 256 luksFormat /dev/sdX`
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mke2fs ... /dev/mapper/my-disk`
- danach:
  - `cryptsetup luksOpen /dev/sdX my-disk`
  - `mount /dev/mapper/my-disk /mnt`
  - `rsync ...`
  - `umount /dev/mapper/my-disk`
  - `cryptsetup luksClose my-disk`

## Praxisbeispiel

```
#!/bin/bash

cryptsetup luksOpen /dev/sdb cs-1000 || exit
mount /dev/mapper/cs-1000 /home/backup/1000 || exit

rsync -a -H --delete --delete-excluded -v \
  --include-from=disc_1000_rsync.filter \
  / /home/backup/1000/freehome

umount /dev/mapper/cs-1000
cryptsetup luksClose cs-1000
```

# Demonstration

- Backup auf verschlüsselte USB-Festplatte

# Q & A