

# **MODUL 143**

## **BACKUP- U. RESTORE- SYSTEME IMPLEMENTIEREN**

Gerhard.Beutler

# 1 Einführung rsync

## Lernziele

- Mit rsync eine Vollsicherung machen können.

## 1.1 Vorbereitung: Dateiserver einrichten



Bevor wir mit den Sicherungen weiterfahren können, bauen wir den Applikationsserver zu einem Dateiserver aus. Dann haben wir mehr Daten zum Sichern! Das Einrichten des Servers ist nicht prüfungsrelevant. Es ist aber Voraussetzung um die Übungen machen zu können.

Zum Bereitstellen eines Dateiservers halten wir uns an ein möglichst einfaches Beispiel. Sicherheitsüberlegungen und -einstellungen sollen hier nicht im Vordergrund stehen. Als Technologie kommt der Linux-Samba-Server zum Einsatz, da dieser für Linux und Windows-Clients ausgelegt ist.



### Aufgabe 1)

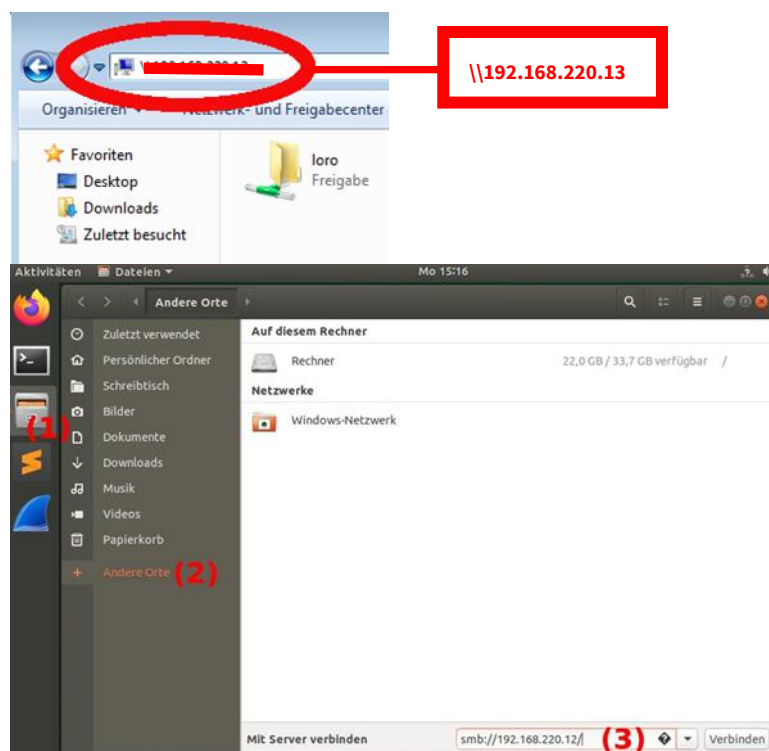
Zur Installation steht Ihnen auf dem Modulshare das Skript `m143-05_LS3_sambaShare.sh` zur Verfügung. Führen Sie das Skript auf dem Applikationsserver (im Heimverzeichnis) aus. Das Skript installiert den Dienst und richtet ihn ein. Im neu angelegten Ordner `smb_share` werden zukünftig die Share-Daten abgelegt. Wenn Sie gerne wissen möchten, was genau passiert: Im hinteren Teil dieses Kapitels ist eine manuelle Anleitung abgelegt.

### 1.1.1 Dateiserver einbinden

Vom Büro-PC kann das Einbinden wie folgt gemacht werden: Starten Sie den Dateimanager und konfigurieren Sie das Share gemäss Grafik.

Das funktioniert auch von Windows aus. Starten Sie `vmWP1` und tippen Sie die Adresse gemäss dem Bild unten im Dateibrowser ein.

Gratuliere! Sie sollten nun auf der Freigabe `loro` Dateien anlegen können. Testen Sie es doch gleich und legen Sie ein `HalloWelt.txt` an.





### Aufgabe 2)

Nun sollen noch ein paar Daten auf den Server. Ein Mitarbeiter hat Fotos des Fuhrparks von «Lorraine Rollt» gemacht, diese sollen für alle erreichbar abgelegt werden.

Holen Sie vom Modul-Netzlaufwerk im Ordner 05\_Software die Datei m143-05\_Velos.zip.

Entpacken Sie diese auf Ihrem lokalen Host-Rechner. Erstellen Sie in der Loro-Freigabe einen Ordner Fotos und legen Sie die entpackten Dateien von einem Büro-PC aus dort ab (vmLP1 oder vmWP1).

*Natürlich können Sie auch eigene Fotos für Ihr individuelles Fallbeispiel nehmen, verwenden Sie aber nicht zu viel Zeit um diese zusammenzusuchen!*



## 1.1.2 SSH-Root auf Applikationsserver

Seit Ubuntu 20.04 hat Root per Default keinen SSH-Zugriff mehr. Da wir diesen aber für die Übung benötigen, richten wir den Zugriff ein:

**vmadmin@vmLS3:~\$**

`sudo -i # root werden`

`passwd # passwort setzen`

`nano /etc/ssh/sshd_config # "PermitRootLogin yes" einfügen`

`systemctl restart sshd`

## 1.2 Einführung in rsync

rsync ist ein ausgereiftes Backup-Programm. Es ist für Einsätze in der Industrie, bzw. produktiven Umgebungen geeignet. Das Programm ist unter einer offenen Lizenz (GNU GPLv3) frei erhältlich und erweiterbar.

Hier wird rsync eingeführt. Die später folgenden Arbeitsblätter setzen das Wissen voraus!

Mit rsync lassen sich komplexe Ordnerstrukturen sichern. Der Sicherungsvorgang kann mit vielen Einstellungen beeinflusst werden, einige davon werden Sie kennenlernen.

Es gibt viele verschiedene Programme welche auf rsync aufbauen oder es erweitern (rsync ist einerseits ein Programm, zugleich aber auch der Name zum zugehörigen Protokoll). Ein paar Beispiele:

- **rsync:** Erstellt ein Backup eines Verzeichnisses. Kann sicher übers Netzwerk arbeiten. Kopiert bei Bedarf nur neue oder veränderte Dateien.
- **rsnapshot:** Ein Wrapper um rsync, geschrieben in Perl. Erlaubt einfaches Erstellen von «Schnappschüssen», woraus sich eine Art «timeline» (bzw.
- «time machine», wie sie Mac-Benutzern bekannt ist) ergibt. Vereinfachte Bedienung im Vergleich zu rsync.
- **rdiff-backup:** Ähnlich wie rsnapshot. In C und Python geschrieben.
- **Windows:** acrosync ist eine eigenständige, kostenpflichtige Implementation des Protokolls für Windows. Das normale rsync kann aber auch «gratıs» über Cygwin auf Windows genutzt werden.
- **Grafische Oberflächen:** Achtung: Dafür muss ein Desktop installiert sein, also nichts für Server!
  - **grsync:** Eine grafische GTK-Oberfläche für rsync.
  - **luckyBackup:** QT-GUI für KDE-basierte Desktops.
  - **BackInTime:** Eine bekannte und einfach zu bedienende Anwendung auf Basis von rsync mit GUI für Linux.



Sie sehen: Bereits um rsync herum wimmelt es nur so von Programmen. Nimmt man alle existierenden Backup-Programme überhaupt, ergibt sich eine kaum überblickbare Liste! Sie können nicht jedes Programm kennen. Die meisten funktionieren aber nach ähnlichen Prinzipien.

*Daher ist es wichtig, dass Sie die Prinzipien verstehen lernen!*

### Aufgabe 3)



Installieren Sie zur Vorbereitung der Übung das Programm rsync auf beiden Servern (Backup und Applikation).

## 1.2.1 Vollbackup mit rsync



### Aufgabe 4)

Erstellen Sie mit rsync eine Vollsicherung des kompletten Heimverzeichnisses von vmadmin auf dem Applikationsserver! Die Sicherung wird auf dem Backupserver abgelegt.

(Sie finden weiter unten auf dieser Seite Hinweise zum Vorgehen.)  
Kommando:

.....

.....

Nicht vergessen: Prüfen Sie ob die Sicherung geklappt hat!

rsync hat sehr viele Optionen. Der Einfachheit halber benutzen wir lediglich zwei Hauptoptionen: -v und -a. Dank -v sehen wir welche Dateien kopiert werden:

-v zeigt während des Synchronisierens  
alle ausgeführten Schritte an

rsync hat sehr viele Optionen. Der Einfachheit halber benutzen wir lediglich zwei Hauptoptionen: -v und -a. Dank -v sehen wir welche Dateien kopiert werden:

-a fasst folgende Optionen zusammen:

- r kopiert Unterverzeichnisse
- l kopiert symbolische Links
- p behält Rechte der Quelldatei bei
- t behält Zeiten der Quelldatei bei
- g behält Gruppenrechte der Quelldatei bei
- o behält Besitzrechte der Quelldatei bei (nur root)
- D behält Gerätedateien der Quelldatei bei (nur root)

Daraus ergibt sich folgendes Kommando auf dem Backupserver um eine Vollsicherung des Applikationsservers zu erzeugen:



**root@vmLS5:~\$**

**rsync -av \ # Archiv- & Verbose-Modus**

**\$USER@192.168.220.13:\$PFAD \ # Quelle**

**\$PFAD # Ziel**



Wir arbeiten hier als root um sicherzustellen, dass Datei-Metadaten erhalten bleiben.

## 1.3 Zusatz: Server manuell einrichten



Dies ist das manuelle Vorgehen zur Einrichtung des Dateiservers. Wenn Sie das Skript auf Seite 1 genutzt haben, müssen Sie diese Schritte nicht machen

### 1.3.1 Samba-Server

Installieren Sie zuerst auf dem Applikationsserver das Paket samba. Danach ist für den Fileserver ein Ordner mit entsprechenden Rechten zu erstellen:



```
vmadmin@vmLS3:~$  
sudo apt update  
sudo apt install samba  
mkdir smb_share
```

Um den Samba-Server zu konfigurieren wird die Datei /etc/samba/smb.conf editiert. Den existierenden Eintrag workgroup

setzen wir auf einen passenden Wert, z.B.: workgroup = LORO. Zuunterst in der Datei fügen wir folgenden Eintrag hinzu:

```
[loro]  
comment = Lorraine Rollt  
path = /home/vmadmin/smb_share  
browsable = yes  
guest ok = yes  
read only = no  
create mask = 0755
```

Zum Schluss muss der Dienst neu gestartet werden:



```
vmadmin@vmLS3:~$  
sudo systemctl restart smbd && sudo systemctl restart nmbd
```

Nun sollten wir uns von Linux und Windows-Maschinen her auf den Server verbinden können!

### 1.3.2 Share einbinden

Das Netzlaufwerk des Servers lässt sich bei Clients per Kommandozeile einbinden. Hierfür erstellen wir einen Ordner und

«mounten» dann die Freigabe auf diesen Ordner (mounten heisst einbinden):



```
vmadmin@vmLS5:~$  
mkdir -p m143/ab05/vmLS3_loroshare  
sudo apt update  
sudo apt install cifs-utils  
sudo mount -t cifs \  
//192.168.220.13/loro m143/ab05/vmLS3_loroshare -o guest
```

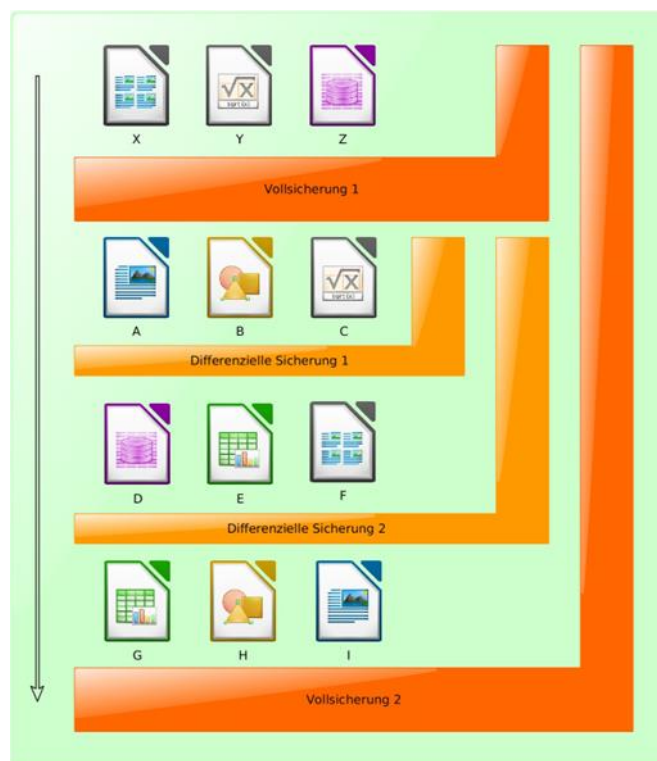
## 2 Sicherung 2: Differenzielle Sicherung

## Lernziele

- Differenzielle Sicherung beschreiben, unterscheiden und erkennen.
- Vorteile, Nachteile und Unterschiede der Differenziellen zu anderen Sicherungen nennen.
- Differenzielle Sicherung mit rsync durchführen.
- Hardlinks erklären und im Kontext von rsync nutzen.

## 2.1 Theorie: Differenzielle Sicherung

Die differenzielle Sicherung ist eine Ergänzung zur Vollsicherung. Dabei soll aber durch trickreiches Vorgehen Speicherplatz gespart werden. Anstatt jedes mal alle Datei- en zu sichern (Vollsicherung), werden nur die Dateien gesichert, welche seit der letzten Vollsicherung verändert wurden oder hinzugekommen sind (Differenz (lat): Unter- schied (dt)). Die Grafik rechts veranschaulicht dies.



Zeichnen Sie ein, welche Daten-Änderungen in welchem Backup enthalten sind:

[illegible]

D2												
V2												



Schauen Sie sich bei Verständnis-Problemen allenfalls noch einmal kurz die Grafik zur Vollsicherung als Vergleich an. Die beiden Verfahren sind sich ähnlich

## 2.1.1. Redundanzen in Backups

Sie sollten mit dem Begriff «Redundanz» bereits in Kontakt gekommen sein (zum Beispiel bei Datenbanken). Aber was ist mit Redundanz bei Backups gemeint? Schauen Sie sich das Beispiel in der folgenden Grafik an.



### Aufgabe 1)

Markieren Sie die redundanten Dateien in der Tabelle unten. Gewichten Sie die Redundanz mit der Angabe einer Zahl: 2 für doppelt, 3 für dreifach.

	A	B	C
V1	x		
V2	2	x	
V3	3	2	x



### Aufgabe 2)

Was ist das «Problem» (wegen der Redundanz), wenn die Sicherungen so wie in der oberen Abbildung durchgeführt werden?

Die Redundanz ist nicht regelmässig, die ältesten dateien werden am häufigsten gespeichert.

.....

.....

.....



Mit einer Differenziellen Sicherung möchte man Redundanzen verringern!



Bevor ein differenzielles Backup gemacht werden kann, wird eine Vollsicherung benötigt! Schauen Sie dazu nochmals kurz die Grafik zum Vorgehen an: Jede differenzielle Sicherung bezieht sich immer auf das vorangehende Fullbackup. Sprich: Ohne vorangehende Vollsicherung keine differenzielle Sicherung.

## 2.2 Vertiefung rsync

### 2.2.1 Backup einer Differenz

Um herauszufinden, wo Daten redundant sind, muss ein Ordner zum Vergleichen angegeben werden. Dies kann mit dem Parameter `--compare-dest` gemacht werden. `rsync` vergleicht dann die Quelle zur letzten Vollsicherung und schreibt die neuen oder veränderten Dateien in den Ziel-Ordner



**root@vmLS5:~\$**

```
rsync -av --compare-dest=$VOLLSICHERUNG $QUELLE $ZIEL
```



`--compare-dest` braucht immer einen absoluten Pfad als Angabe:

**falsch:** fullback\_share

**richtig:** /home/vmadmin/fullback\_share

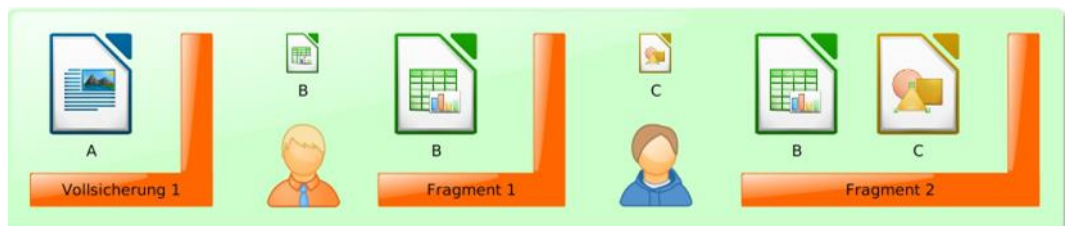
Geben Sie die Ordnernamen jeweils ohne abschliessenden Schrägstrich an:

**falsch:** smb\_share/

**richtig:** smb\_share

### 2.2.2 Problem: «Fragmentierung»

Würden wir das Beispiel der Vorderseite mit dieser Technik als Differenzielle Sicherung nachbauen, kämen wir zu folgendem Ergebnis:



Die Sicherungen nach der ersten Vollsicherung sind nur noch Fragmente (Teile), welche die Änderungen enthalten.



Da nur noch die Differenz gesichert wird, gibt es zwar weniger Redundanzen. Die Sicherungen enthalten aber nicht mehr alle Dateien, was das Zurückspielen von Dateien aufwändiger macht!





### Aufgabe 3)

Gibt es noch Redundanzen? Begründung?

Nein, weil keine dateien mehrmals gespeichert werden sondern nur dessen änderungen.

## 2.2.3 Überbrücken der Fragmentierung mit Hardlinks

Zum Verhindern von Fragmentierung setzt rsync auf Hardlinks. Ein Hardlink ist eine «Verknüpfung» auf eine andere Datei. Im Gegensatz zum Soklink (der wie eine Windows-Verknüpfung ist), ist der Hardlink fest im Filesystem «verbaut» und daher nicht als Link erkennbar.



Jede Datei hat mindestens einen Hardlink, sonst ist sie gar nicht sichtbar.

Macht man einen zusätzlichen Hardlink auf eine Datei, bekommt diese quasi ein zweites «Gesicht», also einen zweiten Pfad über den die Datei erreichbar ist.

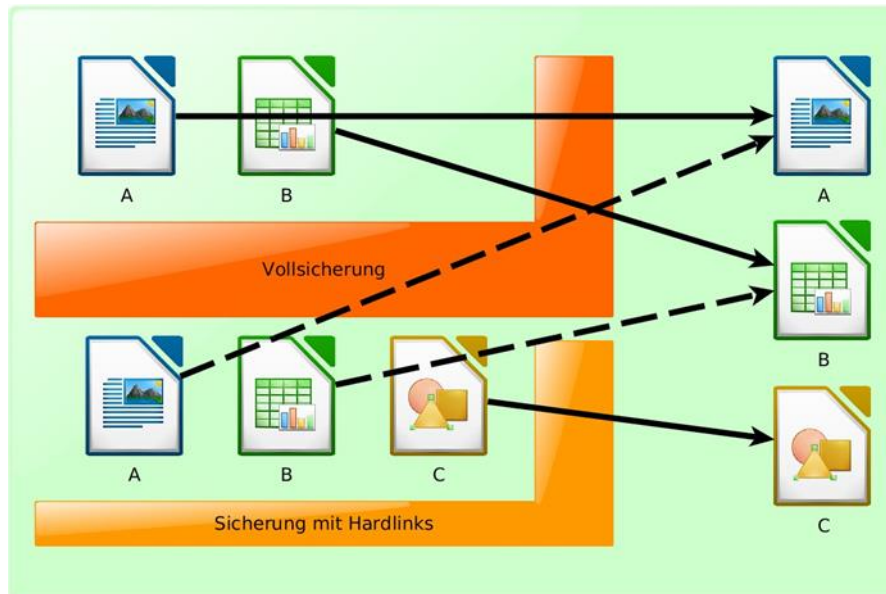
Hardlinks kann man sich mit `ls -l` anzeigen lassen. Die erste Zahl, direkt nach den Dateirechten steht für die Anzahl Hard- links. Die Dateien A.txt und B.txt haben eine 2, d.h. es gibt noch je einen anderen «Zeiger» auf diese Dateien! (Hardlinks können Sie mit dem Kommando `ln` selbst erzeugen. Das geht nur für Dateien, nicht für Ordner!)



### ls -l \*.txt

```
-rw-rw-r-- 2 usr grp 576 Mär 3 11:24 A.txt  
-rw-rw-r-- 2 usr grp 264 Mär 3 14:15 B.txt
```

Die folgende Abbildung soll dies verdeutlichen:



**Zur Abbildung:** Um Platz zu sparen, werden die Dateien A und B als Hardlinks mit eingefügt. Die gestrichelten Pfeile zeigen die neuen Hardlinks, die durchgehenden Pfeile den originalen Link auf die Datei (dies sind auch Hardlinks, einfach die «Ersten»).

**Der Vorteil:** Jede Sicherung enthält alle Daten, aber nichts ist doppelt gespeichert.

## 2.2.4 Links setzen mit Rsync

Durch das setzen von Hardlinks erreicht rsync, dass jede Sicherungen alle Daten beinhaltet. Trotzdem gibt es keine Redundanzen: Hinter den vermeintlich doppelten Daten liegt jeweils die gleiche Datei auf dem Speicher.



rsync benötigt wieder die Angabe des Vergleichsordners. Dieses Mal heisst der Parameter --link-dest.



root@vmLS5:~\$

```
rsync -av --link-dest=$VOLLSICHERUNG $QUELLE $ZIEL
```



Auch hier gilt: --link-dest immer mit absoluter Pfadangabe. Ordernahmen ohne abschliessenden Schrägstrich.

## 2.3 Differenzielle Sicherungen



Sie müssen die Vollsicherung vom letzten Arbeitsblatt gemacht haben um weiter zu arbeiten.



### Aufgabe 4)

Ein Mitarbeiter hat neue Fotos des Loro-Fuhrparks gemacht. Entpacken Sie die neuen Fotos und legen Sie diese vom Büro-PC her auf die Loro-Freigabe (m143-05\_Elektro.zip auf der Modulfreigabe unter Software).

Mit den neuen Daten ist das eben erstellte Fullbackup bereits wieder veraltet. Würde wieder eine Vollsicherung gemacht, wären die alten Fotos doppelt auf dem Backup-Server abgelegt. Dem soll nun eine differenzielle Sicherung entgegenwirken.



### Aufgabe 5)

Führen Sie eine Differenzielle Sicherung durch. Der Sicherungsordner soll **nur die neuen** Dateien beinhalten. Schreiben Sie das komplette funktionierende Kommando auf:

Vorhandene Ordner: /home/phonkd/backup /home/phonkd/testbgp (quelle) /home/phonkd/2

Full backup: `rsync -avz /home/phonkd/testbgp/ /home/phonkd/backup`

Diff: `rsync -av --compare-dest=/home/phonkd/backup /home/phonkd/testbgp /home/phonkd/2/`



### Aufgabe 6)

Prüfen Sie, ob in der eben gemachten Sicherung die korrekten Daten abgelegt sind! Erläutern Sie in eigenen Worten was der korrekte Inhalt der Sicherung ist und warum:

In der neuen sicherung sind nur die änderungen zur vollsicherung zu finden.



#### Aufgabe 7)

Führen Sie nun nochmals eine Differenzielle Sicherung durch. Der Sicherungsordner soll diesmal **alle** Dateien enthalten (bereits vorhandene Dateien sollen als Hardlinks zur alten Sicherung zeigen). Schreiben Sie das komplette funktionierende Kommando auf:

rsync -av --link-dest=/home/phonkd/2/ /home/phonkd/testbgp/ /home/phonkd/3/



#### Aufgabe 8)

Ein Mitarbeiter hat ein paar Rollbretter fotografiert: m143-05-Bretter.zip im Modulshare unter Software. Legen Sie diese neue Fotos auf dem Loro-Share ab. Führen Sie nun wieder eine differenzielle Sicherung durch (mit Hardlinks).

Vergleichen Sie die neu übertragenen Inhalte (nur ein Hardlink) der letzten und dieser Differenziellen Sicherung.. Erklären

dateien welche zum 2ten mal gesicher wurden haben 1nen hardlink mehr



#### Aufgabe 9)

Aktualisieren Sie auf dem Applikationsserver einige Velo-Fotos mit touch. Wie wirkt sich das auf eine nächste differenzielle Sicherung aus?