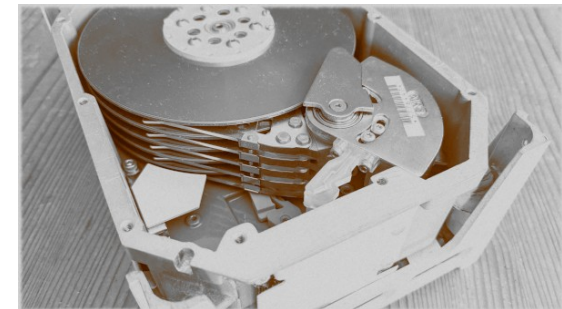


Ein vollständiges Backup

- LinuxDay 2018
 - **Sicheres privates Netz in unsicheren Zeiten**
 - SSH
 - universelle sichere Verbindung
- LinuxDay2019
 - **Backup: rsync, SSH, LUKS im Team**
 - Ergebnis:
 - rsync mit rsnapshot sind ideal für ein Backup
 - schnell, platzsparend
 - schwer wartbar, wenn viele Szenarien
- LinuxDay 2024
 - **Warum sollten wir Btrfs verwenden?**
 - Unterbau für ein Backup
- heute
 - ein komplettes Backup
- Demo am Ende des Vortrages

Physiker in Halle
Fernstudium Theologie
seit 1988 in München
Physiker am Max-Planck-Institut
IT Berater und Entwickler
Senior in Görlitz
Vorträge in
Augsburg, Graz, Dornbirn,
Wien, Kiel, Chemnitz, ...

<https://görlinux.de>



einfaches Backup (privates Foto)



Warum noch ein anderes Backup?

- Beispiel aus dem Vortrag zur Ubucon 2016
 - King's College London
 - “irretrievable data loss”
 - Youtube: „Ein Server ist kaputtgegangen. Sie haben alles verloren“.
 - Bericht 2017: **IT INFRASTRUCTURE RESILIENCY REVIEW**
 - 99.9999% Verfügbarkeit
 - Fallback mit Bandlaufwerken
- alle Datendesaster, die ich erlebt habe, waren menschliche Fehler
 - mangelnde Kommunikation
 - mangelnde Tests
 - ungeeignete Mittel: RAID, NAS, ...
 - zu komplex



- Lösung

- einfach, transparent, automatisch
- K.I.S.S.
Keep It Simple, Stupid
- *All the philosophy really boils down to one iron law, the hallowed 'KISS principle' of master engineers everywhere:*
Eric S. Raymond <https://catb.org/esr/writings/taoup/html/ch01s07.html>
- alles mit Linuxwerkzeugen aufbauen
- rsync, rsnapshot, bash, SSH, LUKS uvam.

- „Three Sysadmin Rules You Can't (And Shouldn't) Break

- Rule #1: Backup Everything (and validate the backup regularly)
- Rule #2: Master the Command Line
- Rule #3: Automate Everything (and become lazy)“

Ramesh Natarajan, The Geek Stuff
<https://www.thegeekstuff.com/2010/07/three-sysadmin-rules/>
<https://www.thegeekstuff.com/copyright/>

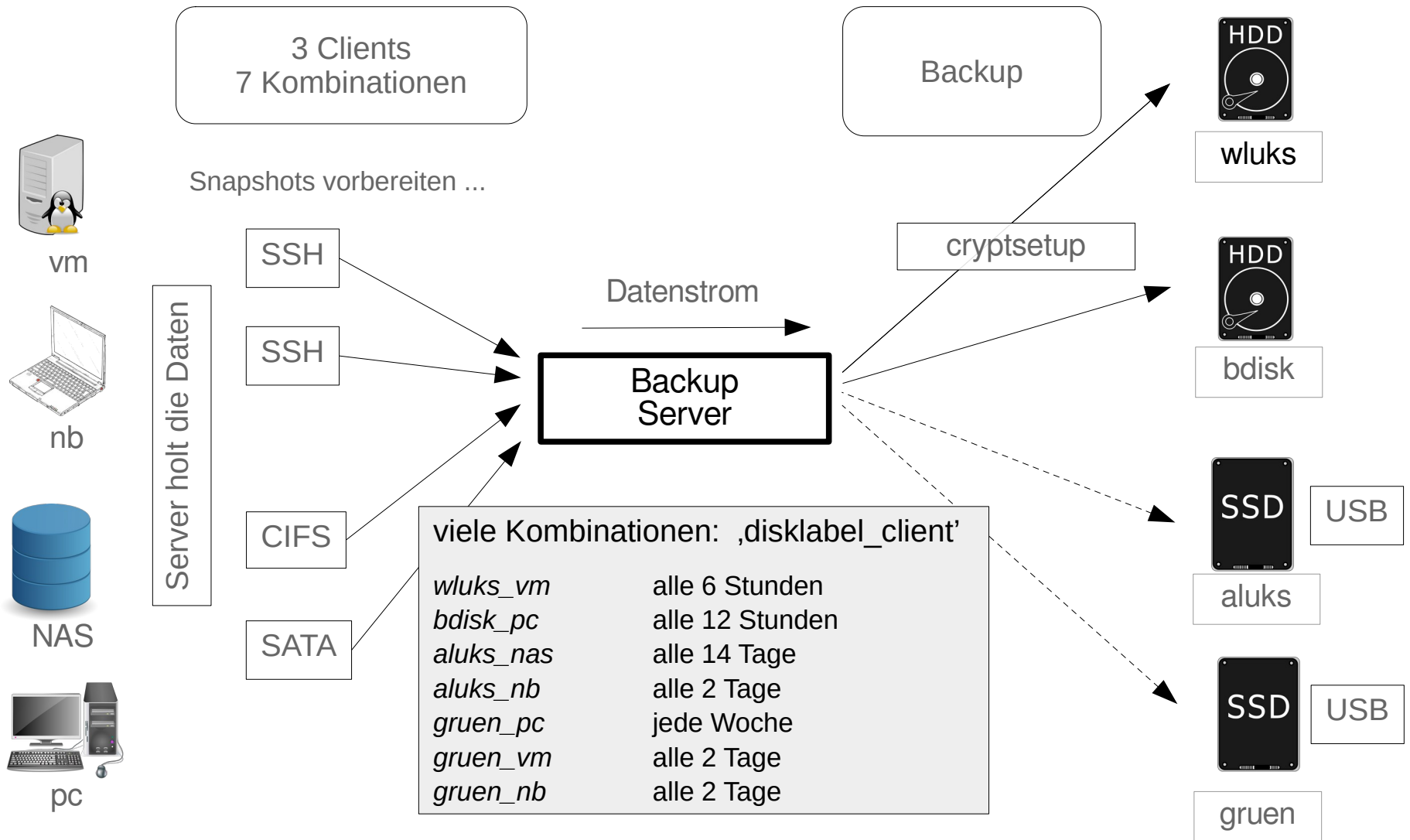


Linux-Kenntnisse nötig

- Warum?
 - zur Navigation im Filesystem
 - zum Testen
 - für Fehlersuche
 - zum Verifizieren (!)
 - uvam.
- feststellen, was in das Backup soll und was nicht
 - Files auf den Clients kennen
 - SSH einrichten, Snapshots, uvam.
- Vorarbeit **kostet viel Zeit**
 - Testdaten anlegen, ca. 1GB
 - Logfiles einrichten
 - Erfolg / Misserfolg loggen
 - und testen, testen, testen
- eine **,one click'-Lösung** für ein Backup gibt es nicht
 - Was ist, wenn das Backup nicht ,erledigt' ist?
 - Wie erfolgt die Validierung des Backup?



Backup kann komplex werden ...



<https://openclipart.org/>
<https://creativecommons.org/publicdomain/zero/1.0/>



Lösung (pseudo code)

start

some checks

while true
do

begin script

for each disk
do

lookup for dirty projects

check and mountHD

disk begin script

for each dirty project at disk
do

project begin script

do rsnapshot

project end script

done

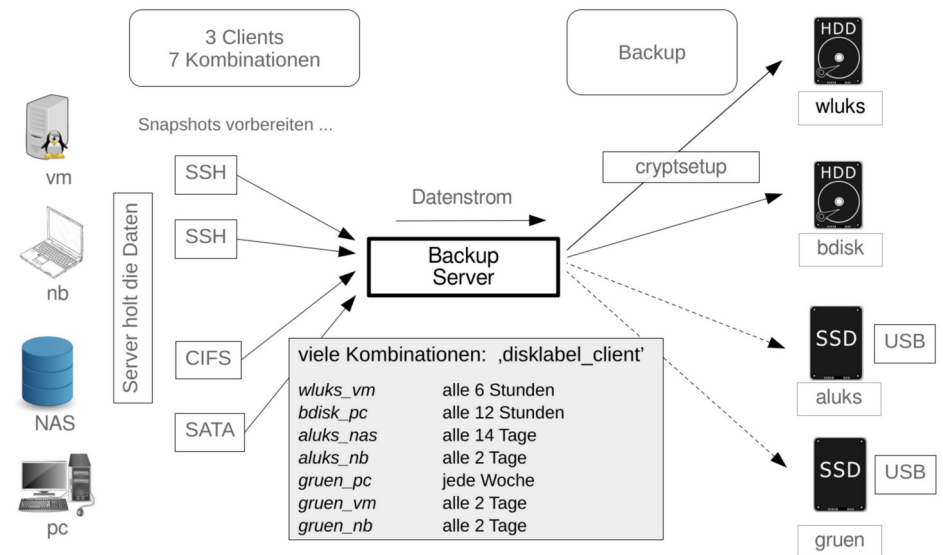
disk end script

umount HD
done

success message

end script

check stop → **end**
done



- 2 externe Platten
 - label ,cluks' und ,gruen'
 - Projekte: ,nb' und ,snap'
 - einmal nur ,copy'
 - danach mit ,snapshot' und 'copy'
 - disk_client Kombinationen
 - 'cluks_nb'
 - 'cluks_snap'
 - 'gruen_nb'
 - 'gruen_snap'
- Backup-Programm kann mitgenommen werden
 - oder hier abholen <https://rleofield.de/backup/index.html>
 - Konfiguration anschauen und anpassen
 - Logfiles anschauen
 - wissen, wie bash, rsync, ssh, btrfs, ext4, rsnapshot funktionieren
- Ist alles korrekt
 - läuft das Backup im Hintergrund
 - automatisch
 - für immer



Take Home Message

Ein Backup mit Linuxwerkzeugen ist mit
rsync, rsnapshot, SSH und LUKS flexibel automatisierbar.

Und läuft und läuft und läuft ...

Vielen Dank für Eure Aufmerksamkeit.

Richard Albrecht

LUG-Görlitz, LUG-Ottobrunn



Richard Albrecht

Linux in Görlitz

