

RAID og LVM på Linux

Gjennomgang av Linux MD-RAID og Linux' Logical Volume Manager

NUUG-foredrag på Rebell, nett og direkte på frikanalen.no

26.04.2022, 18:30 - 20:30

Malin Bruland og Roy Karlsbakk

Meld deg inn i NUUG!

Vurdér også å melde deg inn i NUUG slik at vi kan arrangere enda flere spennende foredrag. Meld deg inn her:

<https://www.nuug.no/innmelding.shtml>

Vi som har laget
foredraget!

Malin

- NUUG-medlem siden 2017 og medlem i styret. Hjemmeværende datahusmor, linuxgamer, hobbymusiker, hjemmekinoentusiast og historieinteressert.
- Fotograf med én utstilling
- Årsstudium i Psykologi og Informatikk. Bakgrunn fra Førskolelærerutdanning med vekt på estetiske fag. Master i autodidaktikk.
- Heltidsbruker av Linux i ca 15 år. Erfaring med Linux MD-Raid og LVM på egendriftet Debian-server, som kjører diverse kvm-vm-er med webserver, databaser, og noen private spillservere

Roy

- Videregående skole, yrkesfaglig, datateknisk, ellers autodidakt.
- Jobber som senioringeniør ved IKT-avdelinga på OsloMet.
- 28 års erfaring med Linux og 20+ års erfaring med Linux-drift.
- Fotograf, 3-4 utstillinger med “konsert-portretter”.
- Datalagringsnerd, hobbyornitolog, hobbyetymolog og egentlig bare omninerd.
- Veldig glad i dikt og leser dem gjerne høyt, men skriver akk så lite. Derav en ivrig diktresitant.

Linux MD-Raid og LVM

Agenda

- Oversikt over hva RAID og LVM er og hvordan det brukes i Linux.
- Hvorfor dette er lurt.
- Gå igjennom ulike RAID-nivåer.
- Opprette og utvide raid-1, -5, -6 og -10
- Lage logiske volum (LV), volumgrupper (VG) og fysiske volum (PV).
- Flytte data fra et fysisk volum til et nytt.

RAID er *ikke* Backup!

- Raid er ***ikke*** backup!
- Menneskelig feil er hovedårsaken til datatap. Folk slettet en fil de ikke skulle og forventer å få den tilbake. RAID løser ikke dette problemet.
- Feil på filsystem eller såkalte stille feil (“silent errors”) der en harddisk korrumpes «uten å vite det sjøl». Gjerne på grunn av ustabil strøm. F.eks. dårlig strømforsyning.
- RAID hjelper lite ved fysisk skade som f.eks. brann.
- Sørg for å ha backup utenfor hjemmet eller kontoret - helst noen mil unna.
- Har du ikke tilstrekkelig bredbåndstilgang, er også lokal backup på USB-disk eller minnepinne et alternativ.

Historikk Raid

- I 1976 ble begrepet RAID først tatt i bruk som en forkortelse for *Redundant Array of Inexpensive Disks*.
- Muliggjorde bruk av rimeligere diskere med vel så god sikkerhet som dyre diskere, gitt nok lagring i en klok konfigurasjon.
- I dag brukes forkortelsen normalt som *Redundant Array of Independent Disks*, kanskje siden leverandører av lagring ønsker at kundene skal henge seg opp i pris?

Historikk Raid

- RAID 1 eller speiling var det første som ble allment anerkjent på syttitallet av flere, for eksempel Tandem NonStop Systems.
- RAID 4 i 1977, patentert av IBM
- RAID 5 i 1986 med et nytt patent fra samme firma
- RAID 6 kom litt seinere, rundt årtusenskiftet, regnet som unødvendig da det kom pga diskpriser, etc.
- Snakk om patent på RAID 6, men innehaveren er (eller var) en aktiv utvikler av Linux-kjerna, anså patentet som i praksis verdiløst, siden innholdet var så åpenbart.

Raid-nivåer

- **Fordeler (+)** og **ulempes (-)** med ulike raid-nivåer

RAID-0 Striping

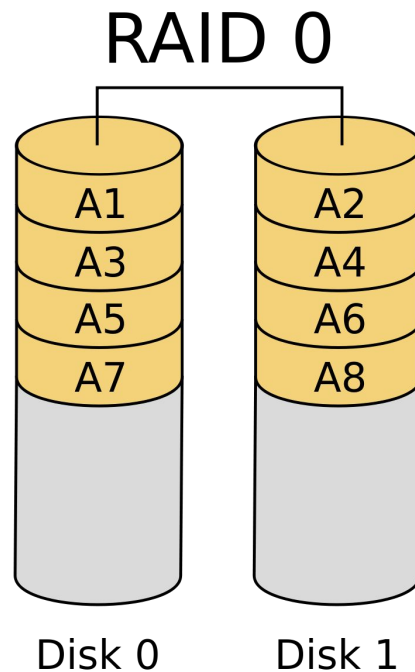
- Kombinerer flere diskker til en større

Fordeler

- + Rask IO, spesielt sekvensielt.
- + Utnytter all plass

Ulemper

- Ingen redundans, derav nivå null
- Dør en disk, dør alt



RAID-1 - Speiling

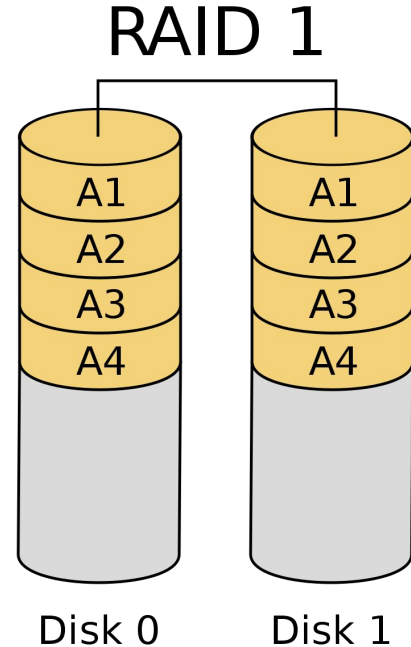
- To diskers i speil
- Alt skrives parallelt til begge diskene

Fordeler

- + Teoretisk dobling av iops ved lesing
- + Leser fra begge samtidig

Ulemper

- Prisspørsmål, da dobbelt antall diskers trengs for å oppnå påkrevd lagringsplass



RAID-2 og 3

- Ikke spesielt mye i bruk i dag og finnes ikke i Linux MD RAID
- Se wikipedia: https://en.wikipedia.org/wiki/Standard_RAID_levels for mer informasjon

RAID-4 - Striping pluss paritet

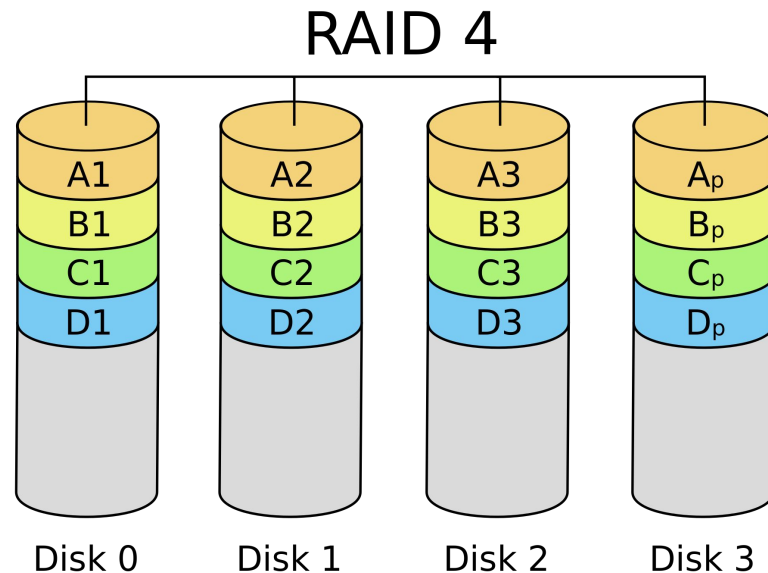
- RAID-0 med paritetsdisk

Fordeler

- + Egen disk for paritet
- + Gjenoppretting fra paritetsdisken om en vilkårlig disk ryker
- + Ryker paritetsdisken, finnes dataene på de andre diskene

Ulemper

- Skrivning til noen av datadiskene medfører at paritetsdisken må oppdateres.
- Paritetsdisken blir en flaskehals



RAID-5 Striping + Roterende paritet

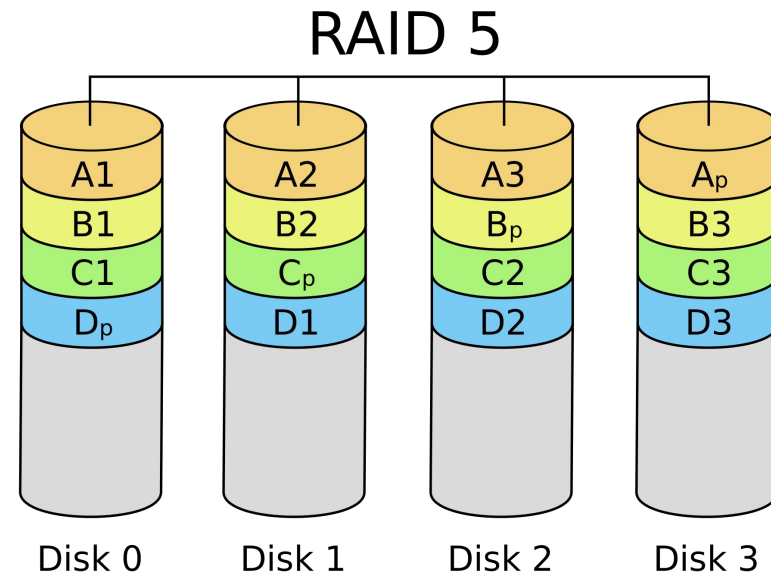
- Videreutviklet fra RAID-4
- Med 3 disker skrives Data, Data, Paritet (DDP), DPD og PDD i evig runddans

Fordeler

- + Fordeler paritet på alle diskene i raidet
- + Tåler å miste en vilkårlig disk

Ulemper

- Kan kun miste én disk

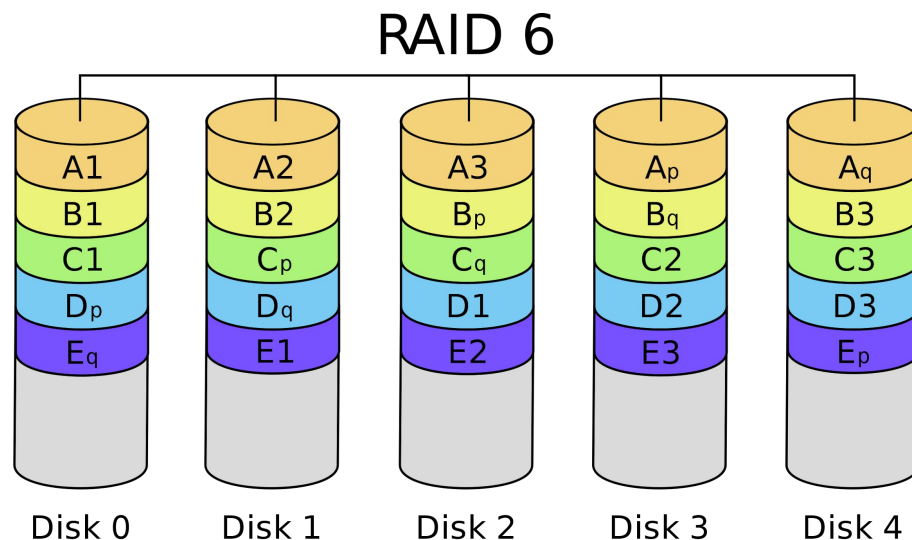


RAID-6 Striping + Dobbel roterende paritet

- Paritet i en ekstra dimensjon
- Lagrer data i DDDPxPy, DDPxPyD, osv

Fordeler

- + Kan miste to vilkårlige diskker uten å miste data
- + Ekstra sikkert om en disk til ryker mens raidet gjenoppbygges



Kombinasjons-RAID

- Kjent som Nested RAID på engelsk
- Mange måter å kombinere RAID-sett på
- Sjansen for å miste 3 disker i f.eks RAID-6 øker proporsjonalt med antall disker
- Man kan f.eks. stripe speil (RAID-0+1) eller speile striper (RAID-1+0)
- Dekker her de vanligste kombinasjons-RAID-ene



RAID-1+0 Striping av speil

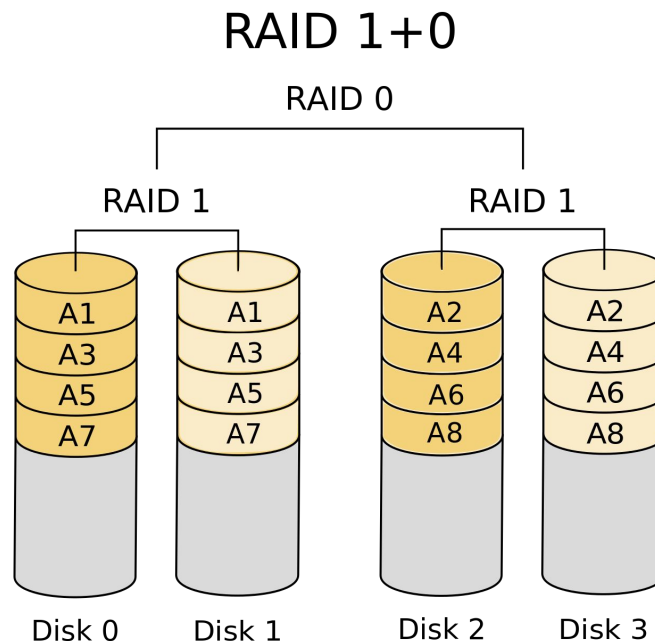
- Kombinerer to RAID-1 i ett RAID-0

Fordeler

- + Høy IOPS og høy sikkerhet
- + Kan miste én disk for hver side av speilet

Ulemper

- Mister som ved RAID-1 halvparten av potensiell lagringsplass
- Ved tap av to disker i samme speil, dør RAID-et

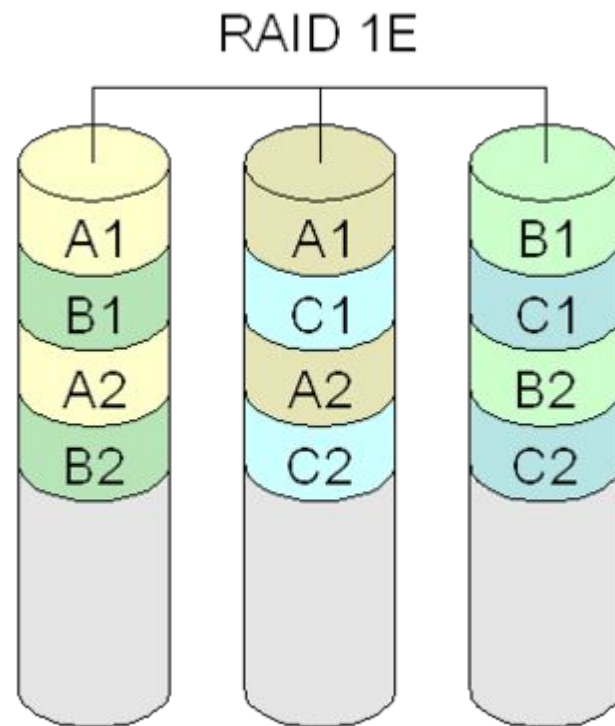


RAID-10

- Som raid 1+0

Fordeler

- + Antall diskker kan deles på annet enn 2



Hardware RAID

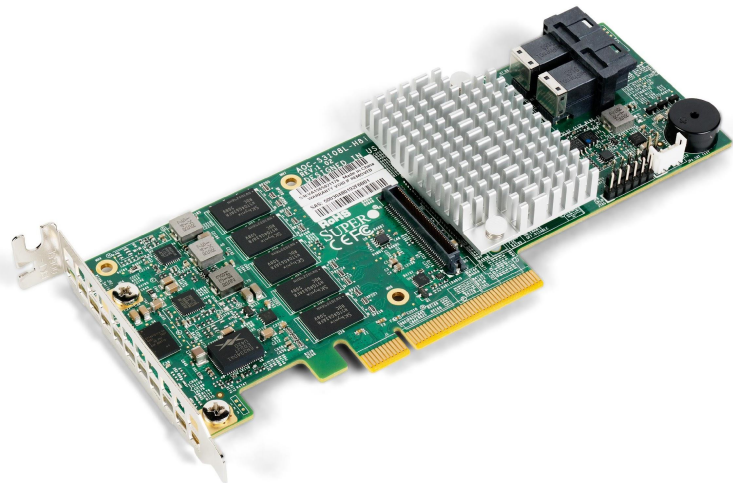
- Kontrollerkort med mikroprosessor og minne som tar seg av fordeling av blokker

Fordeler

- + Fungerer fint helt til noe går galt

Ulemper

- Feilskrivning av paritetsdata, oppdages først når dataene gjenopprettes etter diskfeil
- Dødt kontrollerkort må som oftest erstattes med tilsvarende kort, av og til også med samme revisjon og firmwareversjon



Linux md-RAID - mdadm

- **md** står for Multiple Device
- md-driveren kom inn i Linux-kjerna i 2001 sammen med verktøyet **mdadm**
- Ferdigbygde hylle-RAID-løsninger for hjemmebruk eller kontorfellesskap kjører med stor sannsynlighet ARM-baserte maskiner med Linux og md/mdraid.

Fordeler

- + Muliggjør avanserte RAID-løsninger uten dyr maskinvare
- + Sannsynligvis mest brukte RAID-løsning i dag i volum

Ulemper

- Ingen batteribackup
- CPU må gjøre paritetsgenerering

LVM

Logisk volumbehandler

LVM - Begreper

PV - Physical volume

En fysisk disk, en partisjon, et RAID, det være seg hardware eller software raid eller noe annet som presenteres som en blokkenhet ("block device") for Linux. En PV lagrer dataene i **physical extents** (PE), typisk 4MB hver.

VG - Volume group

Ei volumgruppe består av én eller flere PV-er, samlet i ei gruppe som kan holde på logiske volum.

LV - Logisk volum

Et logisk volum består av et antall **extents** allokert som en egen blokkenhet og eksponert til operativsystemet til bruk for å legge et filsystem på eller tilsvarende.

LVM - Logisk volumbehandler

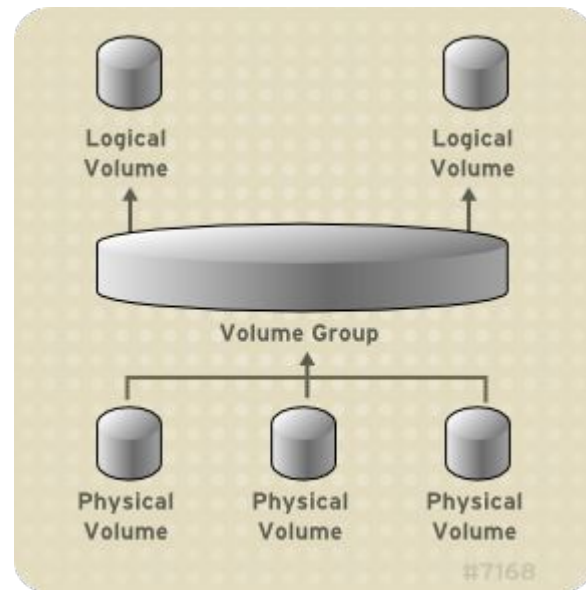
- Lage logiske volumer (LV) fra flere fysiske diskere (PV)
- Fungerer som raid0 om man striper en volumgruppe (VG) over flere Fysiske Volum (PV)
- Støtter tilsvarende redundans som **md**, men brukes i denne sammenhengen bare på toppen av **md**, siden **md** er ansett som mer stabilt

Fordeler

- + Dynamisk endring av størrelse
- + Legg til mer plass ved behov

Ulemper

- Teoretisk mer overhead og forsinkelse enn rene enheter



Valg av filsystemer

På Linux, kan man velge mellom mange filsystemer. Alle har sine styrker og svakheter. Eksempler på disse er

- ext4
- XFS
- JFS
- btrfs
- ZFS

Videre holder vi oss til ext4 og XFS, da disse er de mest brukte til normal bruk

ext4

Fordeler

- + Velprøvd og stabilt over 14 år
- + Er bygget på ext3 (2001), som er bygget på ext2 (1993)
- + Raskt og stabilt for mindre filsystemer
- + Raskt for aksess til millioner av småfiler
- + Kan økes og krympes etter behov

Ulemper

- Har i utgangspunktet en struktur basert på 32bit, så kan ikke vokse over 16TiB
- En filsystemsjekk (fsck) av et ext4 på 10TiB kan fort ta opp mot et døgn
- Antall inoder er statisk og bestemmes når det lages og kan ikke endres seinere

XFS

Fordeler

- + Skalerer godt opp til flere hundre TiB
- + Trenger normalt ikke fsck på grunn av effektiv bruk av journalen, men xfs_repair er rask når behovet melder seg

Ulemper

- Treigt på filsystemer med veldig mange (millioner++) småfiler
- Støtter ikke fsck/xfs_repair ved oppstart, men må gjøres offline i tilfelle korrupsjon
- Kan ikke krympes

Eksempler

- Gå gjennom ulike RAID-nivåer
- Opprette og utvide raid-1, 5, 6 og 10
- Lage fysiske volum (PV), volumgrupper (VG) og logiske volum (LV).
- Flytte data fra et fysisk volum til et nytt

Lage RAID-1

- Starter med å lage et degradert raid-1 med manglende disk, legger raidet til i konfigurasjonen og oppdaterer initramfs

```
mdadm --create --level=1 --raid-devices=2 /dev/md0 /dev/vdf  
missing
```

```
mdadm --detail --scan >> /etc/mdadm/mdadm.conf
```

- Obs. Hvis ikke mdadm.conf er oppdatert i initramfs, vil det oppdagede RAIDet gis et vilkårlig nummer fra 127 og nedover, så vi oppdaterer denne.

```
update-initramfs -u
```

Legge LVM og filsystem på MD-Raidet

- Lag fysisk volum på MD-Raidet

```
pvccreate /dev/md0
```

- Legg det fysiske volumet inn i en volumgruppe med navn *data*

```
vgcreate data /dev/md0
```

- Lag et logisk volum med navn data på toppen av volumgruppen data og bruker all tilgjengelig plass

```
lvcreate -n data -l +100%FREE data
```

- Opprett et filsystem xfs på volumet

```
mkfs -t xfs /dev/data/data
```

Automatisering av monteringspunkt

- Slipper å montere filsystemet manuelt ved oppstart
- Oppretter mappen /data, legger til monteringspunkt i fstab og monterer det nye volumet

```
mkdir /data
```

```
/dev/data/data /data xfs defaults 0 0
```

```
mount -a
```

Legg til disk for å få friskt speil

- Legger til ny disk for å friskmelde speilet

```
mdadm --add /dev/md0 /dev/vdg
```

Utvide fra RAID-1 til RAID-5

- Vi utvider RAID-1 slik at vi får RAID-5 på to disker

```
mdadm --grow --level=5 /dev/md0
```

- RAID-et har nå layout RAID-5, på to disker.
- Vi vil utvide RAID-5 til tre disker for å få mer lagringsplass

```
mdadm --add /dev/md0 /dev/vdh
```

```
mdadm --grow --raid-devices=3 /dev/md0
```

- Legg til en disk i reserve

```
mdadm --add /dev/md0 /dev/vdi
```


Utvide LVM

- Vi vil utnytte den nye plassen.
- Derfor må vi gi det fysiske volumet beskjed om utvidelsen.
- På den måten vil vi fysisk kunne utvide volumet med noe av den nye plassen
- Der vi avslutter med å utvide filsystemet

```
pvresize /dev/md0
```

```
lvresize -L +500M /dev/data/data
```

```
xfs_growfs /dev/data/data
```

Øke RAID-nivå fra 5 til 6

- Vi har nå tre disker i RAID-5 i tillegg til en reservedisk (“spare”).
- RAID-5 kan miste én vilkårlig disk, RAID-6 kan miste to
- Vi utvider til RAID-6, siden med RAID-5
- En reservedisk er bra, men å ha den bakt inn i et RAID-6, er sikrere enn RAID-5+reserve, siden man tåler å miste nok en disk, også under gjenoppbygging av RAID-et etter at den første røyk.

```
mdadm --grow --level=6 /dev/md0
```

Opprette og fjerne volum fra volumgruppa

- Vi lager noen nye volumer i volumgruppa data

```
lvcreate -n skolegreier -L 10M data
```

```
lvcreate -n filmer -L 500M data
```

- Fjernes igjen ved å kjøre

```
lvremove /dev/data/filmer
```

```
lvremove /dev/data/skolegreier
```

Opprette og fjerne volum fra volumgrupp

- Vi fyller i stedet opp hele volumet *data* med den nye plassen og utvider filsystemet

```
lvresize -l +100%FREE /dev/data/data
```

```
xfs_growfs /data
```

Konvertere Raid-6 til Raid-5

- Vi vil ha mer plass, på bekostning av redundans og nedgraderer raidet fra 6 til 5

```
mdadm --grow --level=5 /dev/md0
```

```
mdadm --grow --raid-devices=4 /dev/md0
```

```
pvresize /dev/md0
```

Bytte ut diskene med større

- Vi vil erstatte 1G med 2G-disker og begynner med å fjerne eventuelle reservedisker
- Vi legger først til 2G-disker som reservedisker, setter 1G-diskene i erstatningsmodus, fjerner 1G-disken og skriver over superblokken

```
mdadm --add /dev/md0 /dev/vdp
mdadm --replace /dev/md0 /dev/vdf
mdadm --remove /dev/md0 /dev/vdf
mdadm --zero-superblock /dev/vdf
```

- Regex for resten

```
mdadm --add /dev/md0 /dev/vd[qrs]
mdadm --replace /dev/md0 /dev/vd[ghi]
mdadm --remove /dev/md0 /dev/vd[ghi]
mdadm --zero-superblock /dev/vd[ghi]
```

Ny runddans med utvidelser

- Utvider raidet og lvm med all tilgjengelig plass

```
mdadm --grow --size=max /dev/md0
```

```
pvresize /dev/md0
```

```
lvresize -l +100%FREE /dev/data/data
```

```
xfs_growfs /dev/data/data
```

Flytte data til nytt fysisk volum PV

- Vi lager et Raid-10 med tre 8G-disker
- Utvider volumgruppen *data* med det nye fysiske volumet

```
mdadm --create --level=10 --raid-devices=3 \  
    /dev/md1 /dev/vd{z,aa,ab}
```

```
mdadm --detail --scan >> /etc/mdadm/mdadm.conf
```

```
pvcreate /dev/md1
```

```
vgextend data /dev/md1
```


Flytte data til nytt fysisk volum (PV)

- Flytter dataene over på det nye fysiske volumet
- Fjerner det gamle fysiske volumet fra volumgruppen og lvm

```
pvmove /dev/md0 /dev/md1
```

```
vgreduce data /dev/md0
```

```
pvremove /dev/md0
```

Sjekke at det nye setupet fungerer

- Stopper og starter raidet manuelt for å forsikre oss om at alt funker som det skal

```
umount /data
```

```
vgchange -a n data
```

```
mdadm --stop /dev/md1
```

```
mdadm --assemble --scan
```

```
mount -a
```

Oppsummering

- Vi har vært igjennom Raidets historie og ulike varianter
- Hva Raid brukes til
- Raid er ikke backup. Velg egnet system for det.
- Eksempler med MD-raid i form av Raid-1, -5, -6 og -10
- Hvordan LVM kan brukes i kombinasjon med et raid
- Legge til og fjerne Fysisk Volum (PV), Volumgruppe (VG) og Logisk Volum (LV)
- Flytte data fra fra ett fysisk volum til et nytt

Meld deg inn i NUUG!

Vurdér også å melde deg inn i NUUG slik at vi kan arrangere enda flere spennende foredrag. Meld deg inn her:

<https://www.nuug.no/innmelding.shtml>

Spørsmål og svar!

Meld deg inn i NUUG

Vurdér også å melde deg inn i NUUG slik at vi kan arrangere enda flere spennende foredrag. Meld deg inn her:

<https://www.nuug.no/innmelding.shtml>