

LINUX RAID 10

BESKYT DIN PC MOD  
DISKNEDBRUD

PRIVAT WEB PROXY MED DIGITAL  
SIGNATUR LOGIN

AND THE WINNER IS...

EKSTRAORDINÆR  
GENERALFORSAMLING

CSPLIT  
- DEN SMARTE SAKS



# MERE AKTIVITET I DKUUG

Vi har fortsat den ganske aktive linje i DKUUG, som vi indledte for et par år siden, med bl.a. mange medlemsmøder. Der er **videoer** og andet materiale fra de allerfleste møder på adressen <http://video.dkuug.dk>. Videoerne kan afspilles uden at downloade dem først, bl.a. med mplayer, som også kan spole vilkårligt frem og tilbage i videoerne.

I de forgangne måneder har vi holdt følgende arrangementer:

Foredrag om Suns **ZFS-filsystem** med Jørgen Olsen.

Introduktion til sikkerhedssystemet **SELinux** med Bryan Østergaard.

Møde om **privat www proxy med Digital Signatur login**, med Jon Bendtsen.

25-års jubilæum for den **første internet email** og begyndelsen til det danske internet, med Keld Simonsen.

Arbejds møder om DKUUGs indsats over for **UNIX- og Linux-konsulenter**, hvor vi er ved at lave noget fælles promovning af UNIX- og Linux-løsninger, samt en oversigt over hvilke firmaer der kan levere disse.

Møde om **hosting** i Aalborg. Dette var vores første møde i lang tid i **Jylland**, og Karsten Thygesen fortalte om de mange problemstillinger der er ved at tilbyde en professionel hostingsservice. En særlig tak til Jan Vestergaard for initiativet til jyske møder. Vi håber på mindst et møde til i det jyske i efteråret.

Seminar på ITU hvor **Mærsk Olie og Gas** fortalte om deres brug af UNIX og Linux, og CBS fortalte om brug af Linux i det **danske erhvervsliv**. DTU fortalte om brugen af Linux hos dem, og advokatfirmaet Horten fortalte om licensiering af frit programmel.

Vi har deltaget som medarrangør i et **valg møde** om valg til Europaparlamentet, hvor vi diskuterede europæisk IT-politik. DKUUGs videogrupper var ankermand for live streaming af mødet, som blev fulgt af mere end 50 udefra.

Vi har afholdt en velbesøgt workshop om hjemmesidesystemet **Joomla** med Carsten Thrane, hvor vi måtte lukke for tilmeldingerne, og vi gentager derfor dette arrangement.

Vi havde Jørgen Olsen fra Sun til at fortælle om **dtrace**, som kan bruges til fejlsøgning og at se hvad ens system laver.

I den kommende tid har vi **planer om møder** omkring tynde klienter, Apache, avanceret Postgresql, virtualisering, videregående Joomla, Drupal, Lego mindstorm, Sun teknologier som solaris, zfs og dtrace, og seminar omkring Linux i det offentlige.

Vi fortsætter med vores **Linuxdage**, som bl.a. vil forsøge at udbrede Linux til studerende. Dette er jo en kendt strategi: har man først fået overtalt de studerende, vil det gerne efterspørge det på deres arbejdspladser når de er færdiguddannede, eller i deres studentertjob.

Vi har fået en håndfuld nye **aktive medlemmer**, som er med i udvalgene og hermed med til at sætte deres præg på foreningens virke. Jeg glæder mig til den nye aktivitet, og opfordrer hermed andre medlemmer til at komme frem med forslag til aktiviteter i DKUUG, og evt. deltage i udvalgene. DKUUG har nogle midler og ekspertise og udstyr der kan hjælpe dig med at realisere idéer til fremme af UNIX og åbne standarder. Henvend dig gerne til bestyrelsen, [bestyr@dkuug.dk](mailto:bestyr@dkuug.dk) med dine idéer.

I det hele taget har vi øget den interne aktivitet i foreningen, ved at vores udvalg tilstræber hver at holde **et møde om måneden**. Det bliver til en del møder! Alle medlemmer er velkomne til at deltage i foreningens udvalg og møder. En ændring siden sidst er at Michael Lind Mortensen har overtaget formandskabet for NET-udvalget fra Kristen Nielsen, som har varetaget denne post i over 10 år.

## Indholdet i dette nummer

Jeg har selv bidraget med to artikler omkring RAID i Linux, hvor jeg har stået bag designet af en af de hurtigste RAID-typer, nemlig **RAID10,f2** eller far. Dette kan køre både spejlet og stripet med **kun 2 diske**, og har en ydelse på læsning der nærmer sig RAID0. Læs mere om RAID10 og hvordan man kan benytte det til at lave hurtig og billig sikring af ens arbejdsstation, på hhv. side 4 og 6. Jeg mener at alle bør bruge RAID på deres arbejdsstationer, for at forbedre deres ydelse og deres sikkerhed!

Jon Bendtsen har på side 10 skrevet om hvordan man kan stille sin intranets server tilgængelig for eksterne kunder, på en sikker måde via digital signatur.

På side 14 har Jon så skrevet om hvordan man kan bruge **cspilt** til tegnsætskonvertering af filer. Det bygger i øvrigt på recode.pl, hvor DKUUG via et EU-projekt for mange år siden har lavet de allerfleste tegnsætstabeller.

Side 15 har vi den heldige vinder, Trine Kjærsgaard, af en **Asus eee 900 mini-bærbar**, som vi udlovede i forbindelse med Open Source Days konferencen. Vi håber at Trine vil bidrage med en artikel i et af de kommende numre med hendes erfaringer med den lille maskine.

Til slut: fra side 16 har vi referater af DKUUGs ordinære og ekstraordinære generalforsamlinger. Vi har efterfølgende haft en urafstemning om vedtægtsændringerne, som blev vedtaget. Vi er derfor i den heldige situation, at vi nu kan tilbyde **personligt medlemskab til 300 kr** incl. moms.

*God læselyst  
Keld Simonsen*

# INDHOLD



DKUUG-Nyt er medlemsbladet for  
DKUUG, foreningen for  
Åbne Systemer og Internet

**Udgiver:**

DKUUG  
Fruebjergvej 3,  
2100 København Ø  
Tlf: 39 17 99 44  
Fax: 39 20 89 48  
email: dkuugnyt@dkuug.dk

**Redaktion:**

Keld Simonsen (ansvarshavende)  
Donald Axel  
Kristen Nielsen  
Jon Bendtsen  
Dennis D. Jørgensen

**Tryk:**

Typographic ApS

**Design og Layout:**

Seifert Design & Grafik

**Annoncer:**

Kontakt DKUUGs sekretariat

**Oplag:**

700 eksemplarer

Artikler m.v. i DKUUG-Nyt er ikke  
nødvendigvis i overensstemmelse  
med redaktionens eller DKUUGs  
bestyrelses synspunkter.

Eftertryk i uddrag med  
kilde-angivelse er tilladt.

Medlem af Dansk Fagpresse

DKUUG-Nyt  
ISSN 1395-1440



MERE AKTIVITET I DKUUG .....	2
LINUX RAID 10 .....	4
BESKYT DIN PC MOD DISKNEDBRUD .....	6
PRIVAT WEB PROXY MED DIGITAL SIGNATUR LOGIN .....	10
AND THE WINNER IS .....	14
DRAW.C .....	15
ORDINÆR GENERALFORSAMLING .....	16
EKSTRAORDINÆR GENERALFORSAMLING .....	18
CSPLIT – DEN SMARTE SAKS .....	19

# LINUX RAID10

## Hurtigere og mere fleksibel RAID.

RAID10 har nu i 5 år været en raid-type i Linux-kernen, men den er stadig rimelig ukendt uden for de indviede cirkler, og det er synd.

AF KELD SIMONSEN

RAID10 byder på øget fleksibilitet og op til dobbelt ydelse i forhold til andre gængse RAID-typer.

RAID står for Redundant Array of Inexpensive Disks, og er en måde til at bl.a. forøge ydelsen på et system, og have en vis sikkerhed mod fejl på diske. Dette sker ved at anvende flere diske, der arbejder sammen.

RAID10 er her et alternativ til den kendte RAID1-type, en spejlet array-type. Faktisk er en af de tilgængelige layouttyper, nemlig 'near' identisk i sin udlægning af data på diskene med RAID1 i Linux-implementationen. Der findes 2 andre layouttyper for RAID10, nemlig 'far' og 'offset' – og de kunne også godt være matchet i RAID1, da der normalt ikke ved RAID1 er angivet hvordan data er placeret på diskene.

Navnet RAID10 er lettere misvisende, da RAID10 ofte betegner RAID 1+0, hvor man først har lavet 2 RAID1 (spejlet RAID), som man derefter har lavet et RAID0 af. Til denne slags RAID10 skal der mindst benyttes 4 diske. hvor Linux RAID10 – som en anden slags RAID1 – kun behøver 2 diske for at fungere.

### Dobbelt så hurtig ved sekventiel læsning

Det er især layouttypen 'far' der her eksellerer. 'far' betegner at de spejlede blokke ligger langt væk målt som bloknumre på den spejlede disk ifht. den originale disk – hvor 'near' betegner at samme data på alle diske har samme bloknummer. 'far' er et layout, hvor data logisk er lagt ud som et RAID0, og de replikerende data er dernæst lagt ud som endnu et RAID0. Den tredje layoutmulighed er 'offset', hvor datablokkene ligger rimeligt tæt på hinanden, kun adskilt med en forskydning på nogle enkelte bloknumre mellem de indgående diske.

Ydelsen ved sekventiel læsning for 'far' er derfor som RAID0, som er den hurtigste raidtype. Den teoretiske værdi er antal diske gange hastighed på den langsomste disk, der indgår i arrayet, og praksis bekræfter dette. Dette er dobbelt så hurtigt som RAID1, og også dobbelt så hurtigt som RAID1+0 og de andre RAID10 layouttyper.

En yderligere fordel med 'far' er at kun de yderste sektorer benyttes ved læsning, disse er normalt en del hurtigere end de inderste, op til dobbelt hastighed. Det betyder også at læsehovederne på diskene skal bevæge sig mindre, nemlig kun over de yderste sektorer, som er lidt mindre end halvdelen af den fulde afstand mellem den inderste og den yderste sektor. Den gennemsnitlige søgetid reduceres dermed kraftigt til noget nær det halve.

### Ellers ydelse som RAID1

Ved tilfældig læsning er der teoretisk set ikke forskel mellem RAID1 og de forskellige RAID10-typer, bortset fra at RAID10, far benytter de hurtigste sektorer og har mindre søgetid.

Også ved sekventiel og tilfældig skrivning er der ikke teoretisk forskel mellem RAID1 og RAID10, idet elevatoralgoritmen for filsystemet afhjælper flytning frem og tilbage for far-layoutet.

I scenarier med blandede opgavetyper med meget tilfældig læsning og skrivning klarer RAID10, far sig pænt, bl.a. pga. brug af kun de hurtige yderste sektorer til læsning. RAID10, far giver endvidere god ydelse med sekventiel læsning af store filer på et belastet system. Dette er en væsentlig parameter, da læsningen som regel har en bruger der venter på at det bliver færdigt, og læsning af store datamængder er noget af det der tager længst tid. Læsning af små datamængder tager meget lidt tid, og er derfor ikke problematisk for en ventende bruger, og skrivning venter en bruger ikke på – de bliver bare lagt i en kø til senere udskrivning.

Det skal lige siges her at jeg selv er ophavsmanden til designet af RAID10, far – hovedvedligeholderen af Linux raid-koden Neil Brown har implementeret det i kernen, og jeg har lavet en lille patch til at sikre at kun de yderste sektorer bliver brugt.

### Større fleksibilitet

RAID10 har stor fleksibilitet mht. antal diske der kan indgå i arrayet. Det kan faktisk være et vilkårligt antal, også et ulige antal. Fx kan der indgå 3 diske i et RAID10-array, og på den måde kan man fx i mindre systemer få fuld valuta for brugen af en disk nr 3.

Antallet af spejlede datablokke er også vilkårligt. Det normale er 2 eksemplarer af hver datablok, også benævnt n2, f2 og o2 for hhv near, far og offset layoutvarianterne. Men man kan også have 3 eller 4 eksemplarer af hver datablok, og dette kan både forøge sikkerhed mod nedbrud, og læsehastighed. Det gør naturligvis skrivning langsomm-

mere, fordi data skal skrives flere gange, men ofte er der langt mere læsning end skrivning på et system, og læsning er en mere kritisk operation, fordi brugerne skal vente på færdiggørelsen af læsningen.

### Oprettelse af RAID10 array

Et RAID10 array oprettes med mdadm på normal måde, fx:

```
mdadm -C /dev/md2 --chunk=256
-l 10 -n 2 -p f2 /dev/sda2 /dev/sdb2
```

### Sammenligning af ydelse med andre raidtyper

Der er blevet lavet en del sammenligninger af ydelsen af forskellige RAID-typer i Linux. Ydelsen er både afhængig af de teoretiske værdier, og af kvaliteten af implementeringen i kernen. Har man n diske kan følgende tabel opstilles:

	læsning		skrivning	
	sekventiel	tilfældig	sekventiel	tilfældig
alm disk	1	1	1	1
RAID0	n	n	n	n
RAID1	n/2	n/2	n/2	n/2
RAID10,n2	n/2	n/2	n/2	n/2
RAID10,f2	n		n/2	n/2
RAID10,o2	n/2	n/2	n/2	n/2
RAID5	n-1		n-1	dårlig
RAID6	n-2		n-2	dårlig

### Referencer

<http://linux-raid.osdl.org> linux-raid wiki

<http://std.dkuug.dk/keld/stripemirror.html> Kelds design af RAID10,f2

<http://linux-raid.osdl.org/index.php/Performance> RAID ydelse

# BESKYT DIN PC MOD DISKNEDBRUD

## Brug RAID til at undgå at maskinen går ned ved disknedbrud

AF KELD SIMONSEN

Det følgende vil beskrive, hvordan man i Linux kan sætte et system op til at overleve, selv om en disk går ned midt i driften. Dette kan være vigtigt for en server, der er beregnet til altid at køre. Beskrivelsen er hovedsagelig rettet mod små servere, men det kan også bruges til arbejdsstationer for at beskytte mod at miste data, og sikre at systemet fortsat kører selv om en disk svigter. Nogle anbefalinger om større server-opsætninger er givet i slutningen af artiklen.

Det kræver nogle ekstra hardware, især diske, og beskrivelsen vil også berøre, hvordan du får mest ud af disse diske, det være sig i form af tilgængelig diskplads, eller input/output hastighed.

Teksten afspejler arbejde med Linux 2.6.12 og 2.6.24 kerner, men kan også gælde for kerner før og efter disse versioner.

### Oprettelse af partitioner

Vi anbefaler at oprette en partition hver til /boot, rod, swap og andre filsystemer. Dette kan fx gøres med fdisk, sdfisk, parted eller måske en grafisk grænseflade som Mandriva/PCLinuxos harddrake2. Det anbefales at bruge drev med samme størrelse og ydeevne.

Du kan starte med at lave opsætningen af partitioner på bare den første harddisk, som sættes op med de rigtige typer på hver partition osv., og derefter kan du kopiere det endelige layout til en anden disk med en enkelt sfdisk kommando.

Hvis vi bruger de 2 drev sda og sdb, så kan sfdisk bruges til at lave 4 partitioner (/boot, rod, swap, andet) i RAID-partitioner (type fd):

```
sfdisk -c /dev/sda 1 fd
sfdisk -c /dev/sda 2 fd
sfdisk -c /dev/sda 3 fd
sfdisk -c /dev/sda 5 fd
```

Kopier det endelige layout af sda til sdb:

```
sfdisk -d /dev/sda | sfdisk /dev/sdb
```

Brug følgende til at vise partitionslayoutet:

```
fdisk -l /dev/sda /dev/sdb
```

Partitionslayoutet kunne så se sådan her ud:

Disk /dev/sda: 1000.2 GB, 1000204886016 bytes  
 255 heads, 63 sectors/track, 121601 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1		1	37	297171	fd	Linux raid autodetect
/dev/sda2		38	1132	8795587+	fd	Linux raid autodetect
/dev/sda3		1133	1619	3911827+	fd	Linux raid autodetect
/dev/sda4		1620	121601	963755415	5	Extended
/dev/sda5		1620	121601	963755383+	fd	Linux raid autodetect

Disk /dev/sdb: 1000.2 GB, 1000204886016 bytes  
 255 heads, 63 sectors/track, 121601 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	37	297171	fd	Linux raid autodetect
/dev/sdb2		38	1132	8795587+	fd	Linux raid autodetect
/dev/sdb3		1133	1619	3911827+	fd	Linux raid autodetect
/dev/sdb4		1620	121601	963755415	5	Extended
/dev/sdb5		1620	121601	963755383+	fd	Linux raid autodetect

### Forbered opstart

Systemet bør være sat op til at kunne starte op fra flere enheder, således at hvis en disk fejler, så kan systemet starte fra en anden disk.

**BEMÆRK:** Hvis den første disk svigter, er der nogle BIOS'er som vil regne disk nummer 2 som hdc, mens andre vil bruge den fysiske placering. SATA-drev kan også være "flyttet", og udev kan anvende interessante og uindlysende navne for enhederne i disse tilfælde. Brug af UID-notation til at identificere RAID array medlemmer er derfor ønskeligt.

Nogle BIOS'er vil ikke tillade at starte fra mere end én disk. Hvis dette er tilfældet kan du fysisk flytte den anden disk til kablet på den første disk.

Det er muligt, at BIOS skal være sat op til at være i stand til at starte op for flere drev, også i tilfælde af at en af diskene ikke er til stede. Det anbefales, at du tester din opsætning, så snart du har lavet den færdig, og se om det virker hvis du afbryder strømmen til hver af de involverede diske. Gør kun dette med strømmen slukket til hele maskinen.

På Intel-kompatibelt udstyr er der to almindelige opstartindlæsere: grub og lilo. Både grub og lilo kan kun starte op fra en raid1-partition. De kan ikke starte op fra andre software RAID typer. Grunden til, at de kan starte op fra raid1 er, at de ser raid1-partitionen som en normal disk, og de bruger kun den ene af diskene ved opstarten. Opstartsfasen omfatter kun indlæsning af kernen med et initrd-billede som rod-filsystem, så der er ikke behov for mange data til dette. Kernen, initrd og andre opstarts-filer kan lægges i en lille /boot partition. Vi anbefaler omkring 200 MB til en ext3 raid1-partition.

Lav raid1 og ext3 filsystemet:

```
mdadm --create /dev/md1 --chunk=256
-R -l 1 -n 2 /dev/sda 1 /dev/sdb1
mke2fs -j /dev/md1
```

## lilo

Husk at have root indgangen i `/etc/lilo.conf` sat til `root=/dev/md1` og have rod-partitionen sat til `/dev/md1` i `/etc/fstab`

Gør hver af diskene opstartbare med lilo:

```
lilo -b /dev/sda
lilo -b /dev/sdb
```

Lav dette med `/dev/md1` monteret på `/boot` - og kernen, `initrd` etc flyttet til `/boot`-filsystemet. Denne metode gør det også muligt for os at bruge en RAID-type til rod-filsystemet, som ikke nødvendigvis er `raid1`.

## Grub

Gør hver af diskene opstartbare med grub:

Grub bruger en navngivning, hvor den kalder den første disk for `hd0`, også diske, der er kendt i filsystemet som `/dev/sda` og lignende. `hd0` er altid den disk, hvor grub starter op fra, og det er derfor nødvendigt at omdøbe hver af alternative boot-enheder via 'device'-kommandoen i følgende script.

Her er et script til at installere GRUB på Master Boot Record (MBR) på 2 diske: `sda`, `sdb`:

```
grub --no-floppy <<EOF
root (hd0,1)
setup (hd0)
device (hd0) /dev/sdb
root (hd0,1)
setup (hd0)
EOF
```

## Rod-filsystemet

Bemærk, at opstart af rod-filsystemet kræver, at `initrd`-filsystemet har de nødvendige drivere til SATA-diskkontrollere og RAID-typer, og det er ikke standard for alle `initrd`-filer, selv for nogle af de største distributioner. Du kan blive nødt til at bygge din egen `initrd`.

Rod-filsystemet kan ligge på en anden raid end `/boot`-partitionen. Vi anbefaler raid-typen `raid10,f2`, da rod-filsystemet vil blive brugt mest til læsning, og `raid10,f2` RAID-typen er den hurtigste til læsning, og også tilstrækkelig hurtigt til skrivning. Andre relevante RAID-typer kunne være `raid10,o2` eller `raid1`.

Det anbefales at bruge `udev`-filsystemet, da det kører i RAM, og du dermed kan undgå en række læsninger og skrivinger til disken.

Det anbefales, at alle filsystemer er monteret med `noatime` indstilling, hermed undgås skrivning af filsystem-inoder hver gang en fil er blevet læst eller skrevet.

Lav filsystem med `raid10,f2` og `ext3`:

```
mdadm --create /dev/md2 --chunk=256
-R -l 10 -n 2 -p f2 /dev/sda2 /dev/sdb2
mkfs -t ext3 /dev/md2
```

## Swap-filsystemet

Hvis en disk fejler, hvor processer bliver swappet til, så vil alle disse processer fejle. Dette kan være vitale processer på systemet. Du kan forhindre at processerne fejler ved at have swap-partitioner på et RAID. Pladsbehovet for swap-området er normalt relativt lille sammenlignet med den samlede diskplads til rådighed, så vi anbefaler de hurtigere RAID-typer fremfor de mere pladsøkonomiske. RAID-typen `raid10,f2` synes at være den hurtigste her, andre relevante RAID-typer kunne være `raid10,o2` eller `raid1`.

For at oprette et RAID-array, og lave swappartition direkte på det:

```
mdadm --create /dev/md3 --chunk=256
-R -l 10 -n 2 -p f2 /dev/sda3 /dev/sdb3
mkswap /dev/md3
```

I `/etc/fstab` skal du så have følgende linje:

```
/dev/md3 swap swap defaults 0 0
```



ADVARSEL: nogle "rednings"-cd'er vil ikke bruge raid10 som swap. Dette kan være et problem på systemer med lille hukommelse, og man bliver muligvis nødt til at have swap til at være startet og aktiveret manuelt.

### Resten af filsystemerne

Andre filsystemer kan også være beskyttet mod en fejlende disk. Hvilken teknik der kan anbefales afhænger af dit formål med diskpladsen. Du kan blande de forskellige RAID typer, hvis du har forskellige typer brug på samme server, for eksempel en database og servicering af store filer fra den samme server. Dette er en af fordelene ved software RAID fremfor hardware RAID: du kan have forskellige typer af RAID på en disk med software RAID, hvor en hardware RAID kun kan tage en type for hele disken.

Er diskkapacitet den vigtigste prioritet, og har du mere end 2 drev, så kan raid5 anbefales. Raid5 bruger kun pladsen for 1 drev til at sikre data, mens raid1 og raid10 bruger mindst halvdelen af kapaciteten til at duplikere data. For eksempel for 4 drev gør raid5 75% af den samlede diskplads anvendeligt, mens raid1 og raid10 højest (afhængigt af antal kopier) giver 50% udnyttelse af diskpladsen. Dette bliver endnu bedre for raid5 med flere diske, med 10 diske bruger du kun 10% til sikkerhed.

Er hastighed din hovedprioritet, så vil raid10, f2 raid10, o2 eller raid1 give dig mest hastighed under normal drift. Dette selv hvis du kun har 2 drev.

Er hastighed med en fejlet disk en bekymring, så kunne raid10, o2 være valget, da raid10, f2 er noget langsommere i drift, når en disk er brudt ned.

Eksempler:

```
mdadm --create /dev/md4 --chunk=256 -R
-l 10 -n 2 -p f2 /dev/sd[ab]5
mdadm --create /dev/md4 --chunk=256 -R
-l 10 -n 2 -p o2 /dev/sda5 /dev/sdb5
mdadm --create /dev/md4 --chunk=256
-R -l 5 -n 4 /dev/sd[abcd]5
```

### /etc/mdadm.conf

mdadm har en nem måde at generere filen /etc/mdadm.conf, bare gør:

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

Du bør have mdadm til at rapportere hvis der sker fejl. Dette kan ske ved at tilføje en linje med MAILADDR i /etc/mdadm.conf

```
echo "MAILADDR root" >> /etc/mdadm.conf
```

Eller du kan bruge en e-mail adresse for underrretelse i stedet for root.

Start overvågning af RAID fx ved:

```
mdadm --monitor --scan --daemonise
```

Afprøv at email-underrettelse fungerer ved:

```
mdadm --monitor --scan --test
```

### Anbefalinger for opsætning af større servere

Med en større serveropsætning med flere diske, er det muligt at overleve at mere end en disk svigter. Arraytypen raid6 kan bruges til at kunne overleve 2 disknedbrud, på bekostning af pladsen for 2 diske. /boot, root og swap partitionerne kan oprettes med flere diske, for eksempel en /boot partition, der består af en raid1 på 3 diske, og root og swap partitoner består af raid10, f3 arrays. Da raid6 ikke kan overleve mere end 2 disknedbrud, behøver systemdiskene heller ikke at være forberedt på mere end 2 disknedbrud, og du kan bruge resten af disk IO kapaciteten til at gøre systemet hurtigere.

Denne artikel er en oversættelse af min artikel på [http://linux-raid.osdl.org/index.php/Preventing\\_against\\_a\\_failing\\_disk](http://linux-raid.osdl.org/index.php/Preventing_against_a_failing_disk)

# PRIVAT WEB PROXY MED DIGITAL SIGNATUR LOGIN

AF JON.BENDTSEN@JBIT.DK

## Problem:

Internt website som visse eksterne partnere skal have meget nem adgang til.

## VPN er en dårlig løsning:

En kunde kom med ovenstående problem, dog med den tvist at hverken kunden eller partneren var særlig glad for at der skulle installeres en VPN på visse af partnernes computere, og min kunde ønskede ikke en central VPN da den ville give alle i partner firmaet adgang. Desuden så var min kundes VPN løsning lavet til at give min kundes medarbejdere adgang til alle serverne. Selvom at partner VPN adgangen kunne laves med firewall, så ville webserveren som udgangspunkt give adgang til hele intranettet, og ikke kun udvalgte dele.

## Apache proxy med digital signatur login:

Apache webserveren har mange spændende udvidelsesmoduler, blandt andet mod\_proxy og mod\_rewrite. Desuden kan Apache sættes op til at kræve et klientcertifikat for at tillade adgang. Så jeg lavede en kombination i Apache, og derved kan min kunde udstede et certifikat til de individuelle ansatte hos partneren som skal have adgang. Dette certifikat skal så bare importeres i en browser, og så gives der adgang. Hos min kunde bruger jeg vores egne certifikater, men løsningen virker også glimrende med digital signatur, som jeg vil vise herunder.

Proxy delen kan endda gøres forskelligt for hver eneste url, således at nogen ansatte hos partneren har flere beføjelser end andre.

Hos min kunde brugte vi en helt egen maskine til at køre den specielle Apache proxy, men egentlig kan en vilkårlig Apache installation bruges. Vi lavede

også et hul i firewallen således at partneren kan tilgå den specielt opsatte Apache. For en almindelig bruger hos min kundes partner så skal de således bare gå ind på dette website **httpS://min-kundes-partner-login.dk/** for at få adgang. Dette kan også bruges til webapplikationer som kræver et brugernavn og password for at logge ind, så skal partnerens ansatte bare også have brugernavn og password. Hos min kunde bruges løsningen primært til bugzilla og en flash applikation de har udviklet.

## 1 Opsætning

Nedenunder vil jeg gennemgå hvordan jeg har sat Apache op til at bruge httpS, kræve et klientcertifikat, kræve det rette klientcertifikat, og til sidst 2 forskellige måder at lave en proxy af et andet website. Jeg har anonymiseret opsætningen således at jeg bruger min egen Digitale Signatur og DKUUGs hjemmeside som eksempler.

### 1.1 Generer certifikat til Apache

Kør følgende kommandoer:

```
openssl genrsa -out jbit_dk.key 1024
openssl req -new -key jbit_dk.key -out jbit_dk.csr
openssl x509 -req -days 365 -in jbit_dk.csr
-signkey jbit_dk.key -out jbit_dk.crt
```

og gem filerne i /etc/Apache2/ssl/

Signatur login

Proxy

## 1.2 Bed Apache om at bruge det genererede server certifikat

I filen ports.conf tilføj

```
Listen 443
```

I biblioteket mods\_enabled så laves 2 symbolske links

```
In -s ../mods-available/ssl.conf
In -s ../mods-available/ssl.load
```

I biblioteket sites\_enabled kopierede jeg min gamle jbit.dk fil til ssl-jbit.dk og tilføjede følgende:

```
ErrorLog /var/log/Apache2/https_dk_error.log
CustomLog /var/log/Apache2/https_dk_access.log combined
SSLEngine on
SSLProtocol all
SSLCipherSuite HIGH:MEDIUM
SSLVerifyDepth 1
SSLCertificateFile /etc/Apache2/ssl/jbit_dk.crt
SSLCertificateKeyFile /etc/Apache2/ssl/jbit_dk.key
SSLVerifyClient none
SSLRequireSSL
```

Naturligvis bad jeg også den virtuelle host om at lytte til port 443. Efter at have verificeret at det virker, så er vi parate til at kræve klient certifikat, og at det er et digital signatur.

## 1.3 Client certifikat

Jeg formoder allerede at i har et Digital Signatur certifikat og at det er importeret i jeres browser. Men vi skal også bruge en CA crt fil. Derfor hiver vi lige CA crt ud af vores PKCS12 digital signatur fil. Har du ikke en PKCS12 certifikat fil, så kan du exportere den fra de fleste browsere. Husk at beskytte den med password, da den vil indeholde jeres private nøgle.

```
openssl pkcs12 -in Jon\ Bendtsen.pkcs12
-cacerts -nokeys -out TDC_OCES_CA.crt
```

Denne fil gemmer vi på Apache serveren i /etc/Apache2/ca/ og den er slet ikke hemmelig eller noget. Bemærk at serverens Certifikat ligger i ssl/, hvorimod client certifikat ligger i ca/. Desuden er serverens certifikat selfsigned, hvorimod at klientens er signeret

af TDC. De 2 certifikater behøver slet ikke have noget med hinanden at gøre, og der kan endda være flere forskellige CA som har signeret de forskellige klient certifikater.

## 1.4 Bed Apache om at kræve client certifikat

For at KRÆVE at klienten skal have et certifikat, så ændrer vi en linie:

```
SSLVerifyClient none
```

så der nu KRÆVES et klient certifikat

```
SSLVerifyClient require
```

Dernæst fortæller vi hvor Apache kan finde CA

```
SSLCACertificateFile
/etc/Apache2/ca/TDC_OCES_CA.crt
```

Lad os checke at det virker, genstart Apache og besøg dit website. Problemet er nu at alle med et digital signatur kan logge ind. Jeg kan fx. logge ind med både min private digitale signatur og med min virksomheds. Men Apache har mange muligheder for at stille krav til klient certifikatet. Det styres via SSLRequire, hvor der kan opstilles flere krav, se evt. nedenstående eksempel. SSLRequire skal iøvrigt ligge under et <Directory> Apache konfigurations direktiv.

```
SSLRequire %{SSL_CLIENT_S_DN_O} eq
Snake Oil, Ltd." and \
%{SSL_CLIENT_S_DN_
OU} in {"Staff", "CA", "Dev"}
```

Hvis vi nu skulle give adgang til alle ansatte i JBIT, så ville vi skrive

```
SSLRequire %{SSL_CLIENT_S_DN_O} eq
"JBIT v/Jon Bendtsen // CVR:28410948"
```

Vi genstarter Apache og prøver om det virker. Nu kan

den private signatur ikke få adgang. Hvis jeg prøver med den private signatur får jeg 403 - forbuden. Hvis jeg derimod bruger min firma signatur, så får jeg adgang.

Men det er jo kun udvalgte ansatte hos partneren som skal have adgang, derfor kræver vi også lige at kun Jon Bendtsen må komme ind, ingen andre ansatte.

```
SSLRequire %{SSL_CLIENT_S_DN_O} eq "JBIT v/
Jon Bendtsen // CVR:28410948" or \
    %{SSL_CLIENT_S_DN_N} eq
"Jon Bendtsen"
```

Vi genstarter og tester at det virker. Resultatet af det lille uskyldige or er at jeg nu også kan logge ind med min private signatur. På den måde kan i give adgang til en liste af personer. Lad os ændre det til and, og så kan kun firma signaturen logge ind.

```
SSLRequire %{SSL_CLIENT_S_DN_O} eq "JBIT v/
Jon Bendtsen // CVR:28410948" and \
    %{SSL_CLIENT_S_DN_CN} eq
"Jon Bendtsen"
```

Efter en genstart så er det kun min firma signatur som kan logge ind.

Man kunne liste samtlige ansatte på den måde, eller bruge nogle af de andre certifikat felter, fx. OU (Organisation Unit), som jeg bruger i de certifikater som jeg har lavet til min kundes partnere. Jeg har i OU feltet skrevet partnerens navn, men jeg har også nogle til mig og den anden systemadministrator, så vi ikke behøver starte VPN for at se nagios, UPS status, osv. I systemadministrator certifikaterne er OU sat til kundens eget navn.

### 1.5 Apache proxy

Efter at have fået styr på SSL og klient certifikater er det nu tid til at få Apache til at være proxy. Der findes 2 metoder, mod\_rewrite og mod\_proxy.

#### 1.5.1 mod\_rewrite

```
<Location /dkuug/*>
Order deny,allow
Allow from all
</Location>
RewriteEngine On
RewriteLog /var/log/Apache2/rewrite.log
RewriteLogLevel 0
RewriteRule ^/dkuug/(.*) http://www.dkuug.
dk/$1 [P]
```

Efter en genstart så vil **httpS://jbit.dk/dkuug/** indeholde **http://dkuug.dk**. Men desværre så bruger dkuug.dk hardlinks, dvs. at HTML sourcen skriver `http://www.dkuug.dk/` før alle links, og ikke bare `/foo/bar.png`. Måske mod\_proxy kan løse det problem?

```
cd /etc/Apache2/mods-enabled
ln -s ../mods-available/proxy.conf
ln -s ../mods-available/proxy.load
ln -s ../mods-available/proxy_html.load
ln -s ../mods-available/proxy_http.load
```

Herefter skal vi bare sætte proxy op i Apache

```
ProxyRequests off
<Proxy http://www.dkuug.dk/*>
    Order deny,allow
    Allow from all
</Proxy>
ProxyPass /dkuug/ http://www.dkuug.dk/
ProxyPassReverse /dkuug/ http://www.dkuug.dk/
ProxyPassReverseCookieDomain www.dkuug.dk
jbit.dk
ProxyPassReverseCookiePath / /dkuug
```

CLIENT

Privat Web  
SSL Require

Efter en genstart så indeholder **httpS://jbit.dk/dkuug/** stadig det forventede, men sourcen bruger stadig hardlinks, så vi skal have noget som kan søge i html sourcen og indsætte **httpS://jbit.dk** i stedet for **http://www.dkuug.dk/**. Apache indeholder selvfølgelig det ønskede modul, **libApache2-mod-proxy-html**, som kan erstatte i sourcen.

```
cd /etc/Apache2/mods-enabled
ln -s ../mods-available/proxy_html.load
```

Så skal vi bare tilføje følgende til proxy konfigurationen:

```
SetOutputFilter proxy-html
ProxyHTMLURLMap http://www.dkuug.dk/(.*)
/dkuug/$1 Ri
```

Efter en genstart virker det næsten optimalt. Alle **http://www.dkuug.dk/** er lavet om til **/dkuug/**, men desværre så indeholder sourcen denne dkuug link som ikke bliver lavet om:

```
@import "http://nyweb.dkuug.dk/templates/
dkuug_tema/css/template_css.css";
```

Men hvis vi nu bare tilføjer en extra **ProxyHTMLURLMap** linie?

```
ProxyHTMLURLMap http://nyweb.dkuug.dk/(.*) /
dkuug/$1 Ri
```

Næh, det hjælper heller ikke. Måske en der ikke er host specifik?

```
ProxyHTMLURLMap http://*.dkuug.dk/(.*) /
dkuug/$1 Ri
```

Den virker desværre heller ikke, så der er nogle begrænsninger, men for langt de fleste webapplikationer virker det fint. Jeg har brugt denne metode på følgende webapplikationer:

- Bugzilla, som kører i Apache på en Linux
- Nagios, som kører i Apache på en Linux
- Network UPS Tools upsstats CGI script, som kører i Apache på en Linux
- Smokeping, som kører i Apache på en Linux
- EJBCA - certificat management system, som kører sin egen tomcat java webserver
- kundes selvudviklet flash applikation som kørte på en windows 2003 server og IIS

og alle har virket ganske glimrende både fra Firefox 2.x, Safari, Opera og Internet explorer. Det har ingen betydning for opsætningen af den private web proxy om det website der skal proxy's kører på den ene eller den anden webserver. Det har heller ingen betydning om webapplikationen er lavet i PHP, flash, java, eller statisk html.

Min kunde har nu et internt intranet, og et semi eksternt partnernet som giver visse af min kundes partnere adgang til udvalgte dele af intranettet. Desuden så har min kunde et almindeligt eksternt website som alle kan besøge. Partnernettets forside kunne godt være en kopi af min kundes intranet forside, dog med en ændring at kun nogle links virkede. Men for at gøre det nemmere for partnerne at forstå forside, så er der kun links til de få applikationer de har adgang til. Desuden så er der ingen grund til at min kundes partnere har kendskab til hvilke andre webapplikationer min kunde kører. Partnerne har dog kendskab til hvilke andre partnere som har adgang, men ikke hvad de andre partnere har adgang til.

## 1.6 Læs mere

```
http://httpd.apache.org/docs/2.0/mod/mod_
_ssl.html
http://Apache.webthing.com/mod_proxy_html/
config.html
```

# HTML

# WEB application

# AND THE WINNER IS...

AF KELD SIMONSEN

Vinder af DKUUG konkurrencen på vores Open Source Days stand 2008 om en Asus eee 900 blev Trine Kjærsgaard. Hun er socialrådgiver i Frederiksberg kommune. Hun har ikke internet nu, men får det i begyndelsen af 2009, Så vil hun prøve maskinen.

Ellers vil hun måske prøve det hjemme hos sin bror, som havde lokket hende med på OSD. Hun skal bruge den til at skrive mails, se nyheder, og netbank og bestille biograf- og teaterbilletter.

Hun vinder ellers aldrig noget, sidst hun vandt noget var et tog da hun var 8 år. Hun havde helt glemt at tjekke sin

mail, fordi hun alligevel aldrig vandt noget. Så det var lige før vi var ved at trække lod igen om den.

Vi har overtalt hende til at forsøge

at skrive en artikel om brugen af den. Og hvordan den er i forhold til hendes anden pc, en windows-maskine.





# DRAW.C

```
/*
 * draw makes a random selection between 1 and the 1st argument
 * */

#include <sys/types.h>
#include <signal.h>
#include <stdio.h>
#include <errno.h>
#include <stdlib.h>

long int population,drawn;
unsigned short xsubi[3];
struct timeval t1;
struct timezone *tz = NULL;

int main(int argc, char **argv)
{
    if (argc > 1) {
        int i;
        for (i = 1; i < argc; i++) {
            population= atoi(argv[i]);
            gettimeofday(&t1,tz);
            xsubi[0] = t1.tv_sec & 0xffff;
            xsubi[1] = (t1.tv_usec >> 16) & 0xffff;
            xsubi[2] = t1.tv_sec & 0xffff;
            drawn = population * erand48(xsubi) + 1;
            printf("%ld %ld %lf\n",population,drawn,drand48());
            sleep(1);
        }
    }
    return(0);
}
```

Program til  
lodtrækning  
baseret på  
tilfældig tid.

# ORDINÆR GENERALFORSAMLING

Referat af ordinær generalforsamling i DKUUG, 27. november 2008

---

AF LARS JOSEPHSEN

---

## Dirigent Beretning

### 1. Valg af dirigent:

Dennis D. Jørgensen blev valgt som dirigent.

### 2. Valg af referent:

Lars Josephsen blev valgt som referent.

### 3. Beslutning om mødet er lovligt indkaldt:

Generalforsamlingen besluttede at den var lovligt indkaldt.

### 4. Fremlæggelse og behandling af beretninger:

Formandsberetningen:

Donald Axel berettede om DKUUGs historie og det forløbne år, indtil marts 2008.

Keld Simonsen berettede om resten af året.

Udvalgsberetninger:

Keld Simonsen berettede om administrations-, blad- og marketingsudvalgene.

risten Nielsen berettede om net-, ext-, medlemsmøde samt om The Camp 2008.

Keld Simonsen berettede om klubudvalget.

Eskild Nielsen berettede om standardiseringsudvalget.

Lars Sommer berettede om sikkerhedsudvalget.

Georg Sluyterman berettede om web- og videoudvalgene.

Generalforsamlingen godkendte beretningen, men bemærkede at de tog meget lang tid.

### 5. Fremlæggelse og godkendelse af Årsregnskab 2007:

Jon Bendtsen fremlagde årsregnskabet.

Generalforsamlingen godkendte årsregnskabet.

### 6. Fremlæggelse og godkendelse af budget og kontingenter (fra den nye medlemsstruktur) for det kommende år, samt maksimalt tilladt træk på foreningens formue:

Jon Bendtsen fremlagde budgettet og forslag til kontingenter (Uændret for stormedlemmer, organisationsmedlemmer og individuelle medlemmer. Kontingent for studerende sænkes til 150 kr. inklusiv moms.)

Generalforsamlingen godkendte budgettet og kontingenterne.

Jon Bendtsen fremlagde forslag til særlige regler for maksimalt træk på foreningens formue:

Bestyrelsen må maksimalt trække 1.034.000 kr. på formuen i 2009.

Generalforsamlingen godkendte de særlige regler for maksimalt træk på foreningens formue.



# Referat Beslutning Godkendelse

## 7. Valg af formand imellem:

Keld Simonsen  
Svenne Krap  
Svenne Krap blev valgt som formand.

## 8. Valg af 5 bestyrelsesmedlemmer imellem:

Dennis D. Jørgensen  
Donald Axel  
Keld Simonsen  
Kristen Nielsen  
Michael Lind Mortensen  
5 kandidater til 5 pladser gjorde at alle blev valgt.

Valg af bestyrelsessuppleanter:

Asbjørn Sloth Tønnesen  
Mads Chr. Olesen  
Begge bestyrelsessuppleanter blev valgt.

## 9. Valg af revisor og revisorsuppleant:

Bo Holst-Christensen blev genvalgt som revisor.  
Lars Josephsen blev genvalgt som revisorsuppleant.

## 10. Forslag fra bestyrelsen eller indsendt af medlemmerne.

- (a) Bestyrelsens forslag til vedtægtsændringer og formuestrategi:  
Generalforsamlingen var utilfreds med formuleringen af paragraf 7 og den skal ændres næste år.

Men bestyrelsen pålægges maksimalt at trække 1.034.000 kr. på formuen i 2009.

Dirigenten besluttede at Georg Sluyterman kan stemme på vegne af Lars Sommer som var gået uden at meddele det, trods kontrovers.

Generalforsamlingen pålagde bestyrelsen at indsende årets instruks til banken. Til næste års generalforsamling skal bestyrelsen fremlægge vedtægtsændringer, der pålægger bestyrelsen at indsende årets instruks, til banken, hvert år.

Generalforsamlingen vedtog vedtægtsændringerne, men ikke med 2/3 af alle medlemsstemmer, da mange medlemmer ikke var mødt op til generalforsamlingen.

Jon Bendtsen påpegede at bestyrelsens anbefalede investeringsstrategi medfører høj risiko. Den anbefalede investeringsstrategi er følgende blanding:

Aktier 0-100% (Dog kun børsnoterede fra almindeligt anerkendte børser)

Obligationer 0-100% (Dog kun børsnoterede fra almindeligt anerkendte børser)

Kontanter 0-100%.

Generalforsamlingen vedtog den nye investeringsstrategi.

- (b) Forslag om fjernelse af ubrugte bi-navne fra vedtægterne fra Eskild, Georg og Jon:

Generalforsamlingen vedtog forslaget med 69 stemmer for og 60 stemmer imod.

- (c) Forslag om inflationssikring og bevarelse af formuen fra Jon:

Forslaget blev nedstemt.

Evt.:

Der var ingen punkter under evt.

# Valg Forslag Fremlæggelse Bestyrelse

# EKSTRAORDINÆR GENERALFORSAMLING



Referat fra ekstraordinær generalforsamling i DKUUG den 6/1-2009

1. Valg af dirigent

Asbjørn blev valgt som dirigent.

2. Valg af referent

Lars Josephsen blev valgt som referent.

3. Beslutning om mødet er lovligt indkaldt

Generalforsamlingen besluttede at den var lovligt indkaldt.

4. Andenbehandling af vedtægtsforslag:

A: Afstemning om fjernelse af binavne  
51 stemmer for og 160 stemmer imod.  
Forslaget blev nedstemt.

Et ekstra medlem dukkede op med 5 stemmer.

B: Afstemning om andre vedtægtsændringer.

Forslaget blev enstemmigt vedtaget med 216 stemmer.

5. Eventuelt

Keld Simonsen præciserede at der nu er skabt to nye medlemsklasser:  
Privatpersoner med et årligt kontingent på 300 kr. inkl. moms.  
Pensionistmedlemskab med et årligt kontingent på 150 kr. inkl. moms.

Dette selvfølgelig kun såfremt den kommende urafstemning vedtager  
vedtægtsændringerne.

Dirigent: Asbjørn Jensen

Referent: Lars Josephsen

DKUUG : <http://www.dkuug.dk/> - Fruebjergvej 3, Boks 126, 2100 København Ø - CVR.: 15 07 64 53  
Sekretariatet : Telefon: 7027 9989 Telefax: 3920 8948 Email: sek@dkuug.dk

Kopi af referatet til  
DKUUG's ekstraordinære  
generalforsamling  
den 6. januar 2009

## CSPLIT – DEN SMARTE SAKS

AF JON BENDTSEN

Jeg har arvet en gammel bugzilla installation som ikke kører i UTF8, hvilket bugzilla anbefaler til nye installationer. Yderligere så er det en bugzilla som har deltagere fra Danmark, Norge, Rusland, Indien og sikkert også andre lande. Jeg fandt forskellige metoder til at konvertere mysql databaser, men da charset ikke altid var konsistent gik det ikke altid godt.

Bugzilla har dog også selv et re-code.pl tool hvilket jeg så prøvede. Dette tool gætter sig frem ved at bruge Encode::Detect. Nærmest alle entries (15810) er gættet til cp1252 ?windows-latin1?. Resten (95) er gættet til cp1251 (51), cp1255 (24), shiftjis (9), koi8-r (7). De sidste 4 er gættet til cp866, iso-8859-5, iso-8859-7 og big5-eten.

Desværre er de entries med anderledes encoding spredt ud over mange tabeller. Så for at sikre mig at data stadig er korrekt så vil jeg

udtage nogle stikprøver, mindst 1 i hvert charset, og sammenligne før og efter konverteringen.

```
recode.pl kan køres med --dry-run
```

som rapporterer for hver tabel. Det output er 52055 linier, men mange tabeller har ingen data der skal konverteres, og en enkelt tabel har 40981 linier, andre har ingen.

For at gøre opgaven mere overskuelig så vil jeg udtage hver tabel og kikke på den hver for sig. Den nemmeste metode er at bruge et værktøj som kan tage recode.pl output og splitte i hver tabel, og det er her at csplit kommer ind i billedet.

```
csplit ../recode-out.txt
"/^Converting */" {76}
```

Hver gang recode.pl starter på en ny tabel, så skriver den Converting \$table\_name, så det er jo ganske praktisk at bruge som PATTERN. De {76} er 1 mindre end antallet af linier som starter med Converting \$table\_name. Resultatet er 78 filer, xx00 til xx77. Langt de fleste filer indeholder kun 1 linie, ^Converting ... men nogle af dem indeholder flere linier, en linie for alle de rækker som er konverteret. Nu skal jeg så bare behandle alle de rækker og sikre mig at data ser identisk ud efter konverteringen i forhold til den før konverteringen.



Foto: StockXCHING

2009

1984 - 2009

# SUPERUSERS

## 25



Kurser · Konsulenttydelser · Certificering

Windows · Linux/UNIX · Netværk · TCP/IP · XML · C#/.NET · Java · C/C++ · Perl · SQL



Microsoft



PROMETRIC



Bestil det nye SuperUsers katalog 2009  
på tlf. 48 28 07 06 eller mail [super@superusers.dk](mailto:super@superusers.dk)