

Web-palvelinohjelmistot

- Apache: suurin ja kaunein, kaikki softat tukevat, mutta hieman resurssisyöppö: www.apache.org
- NginX ("engine X"): kevyempi mutta kuitenkin "full-featured", kaikki edes melko usein tarvittava kalusto löytyy: www.nginx.org
- Lighttpd: kevytversio, ominaisuuksia kuitenkin riittävästi useimpiin tarpeisiin: www.lighttpd.net
- Paljon muitakin erilaisiin erikoistarpeisiin

lighttpd: asennus

- apt-get install lighttpd
- nano /var/www/index.html
<http://s019.vm.it.jyu.fi>, <http://tt1.student.it.jyu.fi>
<http://130.234.209.19>, <http://172.21.209.19>
- /etc/lighttpd/lighttpd.conf
/etc/lighttpd/conf-{available,enabled}

lighttpd: kotihakemistot

- `lighty-enable-mod userdir`
- `nano $HOME/public_html/index.html` (ilman sudoa)
`http://s019.vm.it.jyu.fi/~tt0`
- `service lighttpd force-reload`

lighttpd.conf

```
server.modules = { ... }
```

```
server.document-root = "/var/www"
```

```
server.errorlog = "/var/log/lighttpd/error.log"
```

```
server.username = "www-data"
```

```
server.groupname = "www-data"
```

lighttpd.conf

server.port = 80

index-file.names = { "index.php", "index.html"... }

url.access-deny = { "~", ".inc" }

static-file.exclude-extensions = { ".php", ".pl", ".fcgi" }

lighttpd: name-based virtual hosts

- /etc/lighttpd/conf-available/95-local.conf:
\$HTTP["host"]=="s019.vm.it.jyu.fi" {
 server.document-root="/var/www/s019"
}
- mkdir /var/www/s019; nano /var/www/s019/index.html

lighttpd: name-based virtual hosts

- `sudo lighty-enable-mod local`
- `service lighttpd force-reload`
- `http://s019.vm.it.jyu.fi` on nyt eri kuin <http://130.234.209.19> tai `http://tt1.student.it.jyu.fi`

lighttpd: php

```
apt-get install php5-cgi
```

```
lighty-enable-mod fastcgi
```

```
lighty-enable-mod fastcgi-php
```

```
service lighttpd force-reload
```

```
nano ~/public_html/koe.php
```


lighttpd: access log, dir listing

- `lighty-enable-mod accesslog`
`tail -f /var/log/lighttpd/access.log`
- `lighty-enable-mod dir-listing # tietoturvariski!`
`mkdir /var/www/testi; touch /var/www/testi/kala`
`http://s019.vm.it.jyu.fi/testi`

grub

”GRand Unified Bootloader”: lataa käyttöjärjestelmän (kernel+initramfs)

- /etc/default/grub

```
GRUB_DEFAULT # oletusboottivalinta
```

```
GRUB_HIDDEN_TIMEOUT
```

```
# viive ennen menun näyttämistä Esc-näppäimellä
```

```
GRUB_HIDDEN_TIMEOUT_QUIET
```

```
# true=ei näytetä sekuntilaskuria
```

- HIDDEN-asetukset yleensä paras kommentoida pois

grub

GRUB_TIMEOUT # aika menusta oletusboottiin

GRUB_TIMEOUT_STYLE # menu, hidden, countdown

GRUB_CMDLINE_LINUX_DEFAULT

normaali boot komentorivi

GRUB_CMDLINE_LINUX

boot komentorivi, myös recovery)

- "quiet splash" pitää poistaa jos haluaa nähdä boottiviestit

grub

`GRUB_TERMINAL=console` # grub tekstikonsolille

- `/etc/grub.d/`, `/boot/grub/grub.cfg`, `/boot/grub/custom.cfg`
- Muutosten jälkeen komento **update-grub**

/boot, kernel, initramfs

- grub (tai muu bootloader) paikassa, josta BIOS (tai UEFI tai virsh start...) sen löytää
- grub tietää mistä kernel ja initramfs löytyvät, yleensä /boot
- jos grub pitää saada uudelle laitteelle:
grub-install /dev/xda

/boot, kernel, initramfs

- initramfs (initial RAM filesystem) on ramdisk, josta löytyvät tiedostot, joita tarvitaan bootissa ennen varsinaisen tiedostojärjestelmän mounttaamista (kernel modulit, RAID-konfiguraatiot jne); /etc/initramfs-tools/
- Ei välttämätön jos bootissa ei tarvita mitään mitä kernel ei jo osaa (ilman moduleita)

/boot, kernel, initramfs

- jos initramfs:ää pitää muuttaa (RAID-konfiguraatio muuttunut tms):

```
update-initramfs -u
```

- initramfs on gzip-pakattu cpio-archive, tutkittavissa esim. näin:

```
zcat /boot/initrd... | cpio -i
```

dd - copy & convert

- kopioi dataa "raakana" tiedostorajoista välittämättä
- poikkeuksellinen syntaksi:
dd keyword=value [keyword=value...]
- yleisimmät optiot:
 - if=syöttötiedosto (voi olla laite, /dev/sda tms)
 - of=tulostiedosto

dd - copy & convert

bs=lohkokoko (tavuja); myös ibs=... ja obs=...

count=lohkomäärä (syötteestä)

skip=ohitettava lohkomäärä syötteestä

seek=ohitettava lohkomäärä tulostiedostossa

oflags=... (joukko kopiointiin vaikuttavia asetuksia)

conv=... (joukko harvoin tarvittavia konversio-
optioita, big-endian vs. little-endian jne)

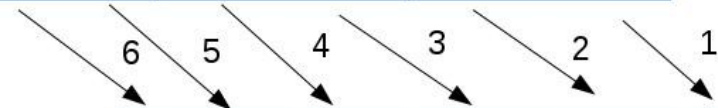
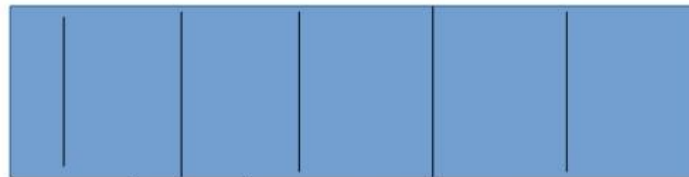
Esim: monen partition muutos

Levyllä kolme partitiota, root, swap ja /home.
Suurennettava root-partitiota

- suurennetaan levyä (qemu-img resize) niin että levyn lopussa on riittävästi tyhjää.
- poistetaan swap ja /home käytöstä (swapoff, umount)
- muutetaan partitiotaulua (fdisk tai parted)
- siirretään /home'n sisältö uuteen paikkaan dd:llä (useana palana jos osittain päällekkäin vanhan kanssa!), bootataan



Siirrettävä paloina lopusta alkaen:



Esim: monen partition muutos jatko

- suurennetaan root-tiedostojärjestelmä (fsck, resize2fs)
- alustetaan swap-osio uusiksi (mkswap)
- päivitetään boot loader (update-grub)
- päivitetään initramfs (update-initramfs -u)
- bootataan ja toivotaan että toimii

Hidasta, vaikeaa ja virhealtista.

LVM: Logical Volume Manager

- Ratkaisu levyjen partitiointiongelmaan
- Yhdistää joukon ”fyysisiä levyjä” (physical volumes, PV) levyryhmäksi (volume group, VG), josta sitten jaetaan ”loogisia levyjä” (logical volumes, LV) kuten partitioita
- PV:tä voi lisätä ja poistaa, LV:tä ja VG:tä voi luoda ja hävittää ja niiden kokoa muuttaa ”lennosta”, konetta boottaamatta

LVM: Logical Volume Manager

- snapshots, striping & mirroring (RAID), clustering
- Jotkin tiedostojärjestelmät (zfs, btrfs) tarjoavat itse osin samaa toiminnallisuutta, mutta eivät korvaa LVM:ää täysin
- Samassa koneessa voi olla sekä LVM että sen ulkopuolisia tiedostojärjestelmiä (usein /boot sen ulkopuolella)

LVM: PV-VG-LV

Physical volumes	hda		hdb		hdc	
Volume groups	vg00					
Logical volumes	/	/usr	/var	swap	/home	(unused)

LVM: PV-VG-LV

- PV:t voivat olla oikeita fyysisiä levyjä, partitioita, RAID-pakkoja, iSCSI-levyjakoja, kryptattuja tiedostoja...
"whatever looks like a disk", myös erilaisia sekaisin
- LV:t eivät välttämättä fyysisesti yhtenäisiä vaan niiden osat voivat olla eri puolilla VG:tä
- Samassa VG:ssä levyt, joita halutaan käsitellä kokonaisuutena (varautua siirtämään kerralla toiseen koneeseen tms)

LVM-komennoista

- komentojen alkukirjaimet kertovat mihin ne vaikuttavat:

pv* physical volume

vg* volume group

lv* logical volume

- kaikissa optio --help
- **man** *komento*

LVM-komennoista

- *lvm komento [optiot]*

”master”-komento, jonka alta löytyvät useimmat muut komennot

- yleensä sama kirjoittaako ”**lvm komento**” vai pelkästään *komento*, mutta esim. recovery-tilanteissa usein vain edellinen toimii, ja siinä on myös joitakin lisäkomentoja, joita ei erillisinä olekaan (esim. devtypes, dumpconfig)
- joitakin harvinaisia komentoja ei löydy lvm:n alta (mm. lvmdump)
- **lvm help**, **lvm help komento**

LVM-esimerkki: uusi VG

Esimerkki uuden VG:n käyttöönotosta:

- Alustetaan fyysiset levyt:

```
pvcreate /dev/vdb /dev/vdc
```

- Luodaan levyryhmä:

```
vgcreate vg00 /dev/vdb /dev/vdc
```

LVM-esimerkki: uusi VG

- Luodaan loogisia levyjä:

```
lvcreate --size 2T --name lvhome2 vg00
```

```
lvcreate -L 4G -n swap2 vg01
```

```
lvcreate --extents 50%VG -n big tt1-vg
```

```
lvcreate -l 100%FREE -n vara tt1-vg2
```

- Kaksi erilaista koonmäärittäytapaa `--size/-L` ja `--extents/-l`
- Paljon optioita erikoistilanteisiin

LVM-esimerkki: uusi VG

- Luodaan LV:lle tiedostojärjestelmät ja mountataan ne:
`mkfs -t ext4 /dev/vg00/lvhome2`
`mkdir /home2; mount /dev/vg00/lvhome2 /home2`
- Otetaan LV swap-käyttöön:
`mkswap /dev/vg01/swap2`
`swapon /dev/vg01/swap2`
- Pysyväksi muokkaamalla `/etc/fstab`'ia

LVM: laitepolut

- Loogisen levyn laitepolku on perinteisesti
/dev/<volume group>/<logical volume>
siis esim. */dev/vg00/lvhome*
- Nykyisissä Linuxeissa suositaan polkua
/dev/mapper/<volume group>-<logical volume>
siis esim. */dev/mapper/vg00-lvhome*
ja sitä on syytä käyttää */etc/fstab*'issa.

LVM: laitepolut

- ”oikea” polku on tyyliä /dev/dm-0, edelliset luodaan symbolisina linkkeinä siihen; dm-laitetta ei yleensä koskaan pitäisi käyttää suoraan
- VG:n nimet historiallisesti vg* ja LV:n lv*, nykyisin useimmin suoraan kuvaavia (esim. VG koneen nimi tai ”kone-vg”, LV ”usr” tai ”lv-usr” jne)

PV:n luonti ja hävittäminen

- Fyysinen volume, PV, pitää alustaa ennen käyttöönottoa:

```
pvcreate [options] laite[...]
```

- optiot harvoin tarpeen, joskus mm. --uuid, --force, --restorefile

- PV:n hävittäminen:

```
pvremove laite
```

- poistaa vain metadatan PV:ltä, harvoin tarpeen (yleensä vgreduce hoitaa)

VG:n luonti

- `vgcreate [options] nimi PV [PV...]`
 - vähintään yksi PV
 - paljon optioita mutta harvoin tarpeen, mm. `--maxlogicalvolumes`, `--maxphysicalvolumes`, `--physicalextentsize`, `--metadatacopies`, `--clustered...`

VG:n hävittäminen

- `vgremove [options] VGnimi [VGnimi...]`
 - jos VG ei ole tyhjä, varoittaa ja pyytää vahvistusta mutta suostuu poistamaan kuitenkin; LV:t voi myös poistaa yksitellen ensin
 - ainoa yleinen optio -f (--force)

VG:n suurentaminen

- `vgextend [options] VGnimi PVpath`
 - lisää VG:hen uuden PV:n
- `vgmerge VG1 VG2`
 - yhdistää (lisää) VG1:een VG2:n (kaikkine LV:neen)
 - VG2 ei saa olla aktiivinen (ensin `vgchange -a n vg2`)

VG:n pienentäminen

- `vgreduce [options] VGpath [PVpath]`
 - poistettavan PV:n oltava tyhjä (ks. `pvmove`)
 - `-a|--all` poistaa kaikki tyhjät PVt
 - `--removemissing` poistaa kadonneet (fyysisesti poistetut/särkyneet) PVt

Datan siirto VG:n sisällä

- `pvmove [options] sourcePVpath [destPVpath [destPVpath...]]`
 - siirtää PV:n sisällön vapaaseen paikkaan tai nimettyihin PV:hen
 - n *LVname* siirtää vain nimetyn LV:n sisällön
 - toimii taustalla, ei estä samanaikaista käyttöä (hidastaa kyllä)
 - voi keskeyttää (vaikka bootata), jatkuu automaattisesti
 - yleisin käyttö PV:n poisto (ennen `vgreducea`)

PV:n koon muuttaminen

- `pvresize [--setphysicalvolumesize koko] PVPath [...]`
 - Muuttaa PV:n koon (oletuksena maksimiin)
 - Ensin muutettava alla olevan laitteen (levyn) koko
 - Pienennettäessä ensin vapautettava lopusta riittävästi tilaa

VG:n siirtäminen

- **vgscan** etsii kaikilta levyiltä PV:t ja VG:t ja ottaa ne käyttöön, usein tarpeen levyjen (PV:den) siirtojen/muutosten jälkeen.

Harvemmin tarvitaan erikseen komentoja

- `vgexport [-a|--all] VGName [VGName...]`
poistaa VG:n määrittelyn järjestelmästä siirtoa varten
- `vgimport [-a|--all] VGname [VGname...]`
palauttaa VG:n määrittelyn ts. ottaa sen käyttöön
(mahdollisesti eri) järjestelmään
- `pvscan`, `lvmdiskscan` etsivät PV:tä/LV:tä (hyvin harvoin tarpeen).

VG:n nimen muuttaminen

- `vgrename OldVG NewVG`
 - samassa koneessa ei saa olla kahta samannimistä VG:tä
 - *OldVG* voi olla nimi, polku tai UUID
 - VG ei saa olla käytössä

VG:n konfiguraation tallennus

- `vgcfgbackup`, `vgcfgrestore`: VG:n metadatan backup ja palautus, yleensä automaattinen
- Joskus tarpeen särkyneen PV:n korvaamisessa (ensin `pvcreate --restorefile ... --uuid ...`)
- `/etc/lvm/backup`

LV:n luonti

- `lvcreate [optiot] volumegroup [PV[:PE[-PE]]...]`
 - n|--name *nimi*
 - L|--size *koko (kilo/mega/gita/tera/peta/exa)tavuina* **tai**
 - l|--extents *koko (LVM)lohkoina tai prosentteina*
(koko VG:stä %VG, vapaasta tilasta %FREE, PV:stä %VPS)

LV:n luonti

- Paljon muitakin harvemmin tarvittavia optioita, mm.
PV:n ja PE:t (physical extents) voi määrätä
- ei luo tiedostojärjestelmää, se pitää tehdä erikseen (mkfs, mkswap)

esim. **lvcreate -n home -L 1G vg00**

LV:n hävittäminen

- `lvremove [f|--force] LVpath`
 - ensin tarvittaessa `umount` tai `swapoff` (pois myös `fstab`'ista)
 - esim. **`lvremove /dev/vg00/lv2`**

LV:n koon muuttaminen

- `lvresize [-L|--size] [+_]koko [-r|--resizefs] LVpath [PVpath]`
- `lvresize [-l|--extents] [+_]koko [-r|--resizefs] LVpath [PVpath]`
 - koon tai sen muutoksen voi ilmaista monella tavalla, esim. `--extents 50%VG`, `-l 100%FREE`, `-L+1G` ...
 - suurennettaessa myös halutun PV:n voi määrätä
 - VG:ssä pitää olla tarpeeksi tilaa

LV:n koon muuttaminen

- Tiedostojärjestelmä pitää laajentaa tai suurentaa jälkeinpäin (resize2fs tms) tai käyttää -r -optiota (ei toimi aina, riippuu tiedostojärjestelmästä)
- Variantit vain suurentamiseen/pienentämiseen:
lvextend, lvreduce
 - soveltuvin osin samat optiot kuin lvresize
 - hiukan turvallisempia kuin lvresize (vähemmän tyoiluvaaraa)

LV:n nimen muuttaminen

- lvrename *OldLV NewLV*
- LV ei saa olla käytössä

Volume Groupin tila

- `vgdisplay [options] [VGname...]`

`vgs [options] [VGname...]`

- VG:n status, erityisesti vapaan tilan määrä
- `vgdisplay` ”puheliias”, `vgs` tiivistetty
- Paljon optioita tulostuksen muotoiluun ym

Loogisen Volumen tila

- `lvdisplay [options] [LVName...]`

`lvs [options] [LVName...]`

- LV:n status, mm. koko ja VG
- Ilman LV-nimeä listaa kaikki LV:t

Fyysisen Volumen tila

- `pvdisplay [options] [PVpath]`

`pvs [options] [PVpath]`

- PV:n status, erityisesti vapaan tilan määrä
- Ilman PVpath-argumenttia listaa kaikki PV:t

LVM: vgchange

- `vgchange [options] VGName`
 - muuttaa VG:n attribuutteja (melkein kaikkea mitä `vgcreate` asetti)
 - erityisesti VG:n aktiivisuus: `vgchange -a [n|y]` - yleensä tapahtuu automaattisesti, mutta `recovery`-tilanteissa muuttaminen käsin joskus tarpeen, samoin `vgmergen` kanssa

LVM: lvchange, pvchange

- `lvchange [options] LVPPath [LVPPath...]`
 - muuttaa LV:n attribuutteja, harvoin tarpeen
- `pvchange [options] PVPath`
 - muuttaa PV:n attribuutteja, harvoin tarpeen
 - `pvchange -x n` asettaa PV:n ”käyttökieltoon” esim. tulevaa poistoa ennakkoiden

LVM snapshots

- `lvcreate --snapshot -L size -n name [...] OrigLV`
- "jäädytetty" kopio LV:stä luontihetkellä, sisäisesti tallettaa erot
- vie tilaa vain (koko ajan kasvavan) eron verran, mutta lakkaa toimimasta jos snapshot-LV tulee täyteen

LVM snapshots

- mahdollistaa koherentin varmuuskopion (kaikki snapshot-LV:n osat edustavat samaa ajanhetkeä), tärkeää etenkin tietokantojen kanssa
- voi käyttää yksinäänkin tilapäisenä varmuuskopiona (ennen jotain potentiaalisesti katastrofaalista kokeilua), palautus komennolla

`lvconvert --merge snapshot-LV-path`

LVM snapshots

- snapshot-LV:tä ei normaalisti muuteta (mount -ro), mutta sitä voi muuttaa jos esim. halutaan säilyttää valikoituja muutoksia ennen palautusta
- yleensä hävitetään heti kun ei enää tarpeen (kun varmuuskopio tehty tms)

LVM RAID

- LVM sisältää (software-)RAID-toiminnallisuutta (striping & mirroring, nyttemmin myös RAID4/5/6), mutta sitä käytetään melko harvoin nykyisin: RAID yleensä toteutetaan erikseen mdadm:llä (monipuolisempi ja kypsempi, enemmän työkaluja) tai rauta-RAID-ohjaimella, ja siten luotua RAID-laitetta käytetään PV:nä LVM:lle.

LVM clustering

- Clustering (CLVM): Jos käytössä on jaettu levy (SAN tms), LV voidaan jakaa monelle koneelle yhtäaikaan. Synkronoinnista huolehtii clvmd (cluster lvm daemon), mutta toiminta yleensä edellyttää, että sovellukset ovat "cluster-aware".
- High-Availability -ympäristöön on RedHatin HA-LVM, joka myös jakaa levyn kahdelle koneelle mutta aktiivisena vain toisessa; optimoitu failover-scenarioon, ei edellytä cluster-säätöä sovelluksilta.

LVM-esimerkki: /home täynnä

- Olkoon /home (/dev/vg1/home) täynnä.
- Tarkistetaan onko VG:ssä tilaa:

```
vgdisplay vg1 | grep Free #tai#  
vgs vg1
```
- Jos VG:ssä ei ollut tilaa, lisätään sinne levy tai pienennetään tai poistetaan jokin vanha LV

LVM-esimerkki: /home täynnä

- Kun tilaa on, laajennetaan /home:

```
lvextend -L +1G -r /dev/vg1/home
```

- Jos -r -optio unohtui tai ei toiminut, laajennetaan tiedostojärjestelmä, tyypistä riippuen:

```
resize2fs /dev/vg1/home # ext*
```

```
xfs_growfs /home # xfs
```

```
mount -o remount,resize /home # jfs
```

LVM-esimerkki: levyn lisäys

- Lisätään levy ensin ”fyysisesti” (virtuaalikoneelle vain virsh attach-disk tms), olkoon se /dev/vdb, partitioitu yhtenä osiona. (Partitiointi on yleensä välttämätöntä boottilevyn kanssa, eikä haittaa muutenkaan.)

```
qemu-img create ...
```

```
virsh attach-disk ... vdb --persistent # alustakoneessa
```

```
fdisk /dev/vdb # virtuaalikoneessa
```

LVM-esimerkki: levyn lisäys

```
pvcreate /dev/vdb1      # alustetaan PV
vgextend vg1 /dev/vdb1 # lisätään PV VG:hen
# tarkistetaan miten meni:
pvs
vgs vg1
```

LVM-esimerkki: /usr/local erilleen

Halutaan tehdä /usr/local'ista oma tiedostojärjestelmänsä. Oletetaan, että VG:ssä on tilaa (tässä 1GB).

- Luodaan uusi LV ja alustetaan se:

```
lvcreate -L 1G -n local vg1
```

```
mkfs -t ext4 /dev/vg1/local
```

- Lisätään määrittely /etc/fstab'iin:

```
echo "/dev/mapper/vg1-local /usr/local ext4 defaults 0 2" >>/etc/fstab
```

LVM-esimerkki: /usr/local erilleen

- vanha /usr/local tilapäiselle nimelle, mountataan uusi:

```
mv /usr/local /usr/oldlocal
```

```
mkdir /usr/local
```

```
mount /usr/local
```

LVM-esimerkki: /usr/local erilleen

- Tarkistetaan ja tarvittaessa korjataan hakemiston oikeudet:

```
ls -ld /usr/oldlocal
```

```
chmod 755 /usr/local
```

```
chown root:root /usr/local
```


LVM-esimerkki: /usr/local erilleen

- Siirretään data ja hävitetään vanha hakemisto

```
mv /usr/oldlocal/* /usr/local
```

```
rmdir /usr/oldlocal
```

- /usr/local ei saa olla käytössä siirron aikana!

LVM-esimerkki: levyn vaihto

Vaihdetaan kokonaan LVM-käytössä oleva levy (vdb) toiseen (vdc), koska se on on menossa rikki tai käynyt pieneksi tms.

- Kytetään uusi levy "fyysisesti":

```
virsh attach-disk ... vdc
```

- Lisätään uusi levy VG:hen:

```
pvcreate /dev/vdc
```

```
vgextend vg1 /dev/vdc
```

LVM-esimerkki: levyn vaihto

- Tyhjennetään vanha levy:

```
pvmove /dev/vdb
```

- Poistetaan vanha levy VG:stä:

```
vgreduce -a vg1
```

- Poistetaan vanha levy:

```
virsh detach-disk ... vdb
```

- Levykirjaimet saattoivat vaihtua – tarkista!