

# virt-install - oikeudet

- virt-install osaa luoda tiedoston levyimageksi jos sitä ei ole, mutta se vaihtaa sen omistajaa ja oletuksena jättää lukuoikeudet vain itselleen, joten sitä eivät muut voi kopioidakaan – käyttökelpoinen vain jos olet root. Mutta jos image on valmiiksi tehty, omistaja vaihtuu mutta suojaukset eivät. Imagen luonti etukäteen:

```
qemu-img -f qcow2 kone.img 3G
```

# levyformaatit

- KVM-qemu tuntee useita levyformaatteja. Parhaiten tuettu ja yleensä helpoimmin toimiva on *qcow2*; toinen yleinen on *raw*.  
(*qcow2* = qemu copy-on-write version 2,  
qemu = Quick EMUlator)

# virt-install/virsh etäkäyttö

- Kone, jossa virt-install ajetaan voi olla eri kuin se, johon virtuaalikone luodaan:

```
virt-install --connect qemu+ssh://lonkaX/system -n tunnus1  
--memory ...
```

- Myös virsh ja virt-viewer toimivat etänä samalla syntaksilla
- ssh-yhteydet on valmisteltava etukäteen (authorized\_keys)
- Levyimage luotava etukäteen (qemu-img create...)

# Varmuuskopiointi: image

- Sammuta virtuaalikone ensin - elävästä koneesta otettu kopio ei ole koherentti - ja:  
cp kone1.img kone1.img.bak
- Voidaan myös siirtää toiseen alustakoneeseen (xml-tiedosto siirrettävä myös, virsh dumpxml...)

# Varmuuskopiointi: image

- Tiedostojen omistaja muuttuu kun virtuaalikone käynnistetään, palautus esim.

```
rm -f kone1.img; cp kone1.img.bak kone1.img
```

*tai*

```
cp -f kone1.img.bak kone1.img
```

# Varmuuskopiointi: image

- ei säännöllinen varmuuskopiointitapa, vain poikkeustilanteissa, levyimagea siirrettäessä tai ennen versiopäivitystä tai jotain "tämä saattaa rikkoa kaiken" -säättöä
- polun vaihtuessa muutettava xml-tiedostoa (virsh edit, tai virsh dumpxml...define)

# Muistin kopiointi: virsh save

- Joskus halutaan kopio käynnissä olevan koneen tilasta (RAM &c), sen voi tehdä näin:  
virsh save kone tiedosto
- Kone "hibernoi", jatkaa komennolla  
virsh restore tiedosto

# Muistin kopiointi: virsh save

- Samasta tiedostosta voi palauttaa monta kertaa "samaan ajanhetkeen"; levyimage on tällöin talletettava ja palautettava erikseen
- Joskus kätevä ennen vaarallisia kokeiluja
- Voi vaihtaa alustakonetta, mutta käyttöjärjestelmän ja kirjastojen oltava kyllin samanlaisia kuin talletettaessa



# command expansion, eval

- Komennon tulos merkkijonoon:

```
year=$(date +%Y); month=$(date +%m)
```

```
day=`date +%d` # vanha tapa, ei suositeltava
```

```
files=$(ls -l $(grep -l kala *.txt))
```

# command expansion, eval

- Merkkijono komenoksi:

```
eval $(date '+year=%Y month=%m day=%d')
```

- `x=ls; $x # toimii`

```
x='y=z'; $x # ei toimi
```

```
x='y=z'; eval $x; echo $y # toimii
```

# loop device

Mountataan tiedosto kuin se olisi levy:

```
mount /tmp/disk.img /mnt/tmp -t ext4 -o loop=/dev/loop3
```

Käytetään tiedostoa kuin se olisi levy:

```
losetup -f /dev/loop0 file.img
```

(yleensä tehtävä roottina tai sudo ...)

# loop device 2

Partitiot näkyviin:

```
kpartx -a /dev/loop0
```

Mountataan em. tiedoston 1. partitio:

```
mount /dev/mapper/loop0p1 /mnt/tmp
```

Vapautetaan:

```
umount /mnt/tmp; losetup -d /dev/loop0
```

# Virtuaalilevyn suurentaminen 1

Edellyttää yleensä aina boottia. Vaarallinen!

Vaiheet:

- (1) Suurennetaan levyimage
- (2) Muutetaan partitointia
- (3) Suurennetaan tiedostojärjestelmä(t) ja swap

# Virtuaalilevyn suurentaminen 2

```
virsh shutdown kone # jatkuu alustakoneessa
```

```
mv kone.img kone.img.bak
```

```
cp kone.img.bak kone.img
```

(ensin mv jotta saadaan kirjoitusoikeudet  
kohdalleen)

```
qemu-img resize kone.img +2GB
```

# Virtuaalilevyn suurentaminen 3

virsh start kone # jatkuu virtuaalikoneessa

sudo fdisk -c -u /dev/vda1 # tai parted

- poistetaan kaikki partitiot ja luodaan uusiksi, root-partition alun pitää säilyä täsmälleen ennallaan (swap voi siirtyä, muut jos siirtää datankin), samoin tyyppien ja boot-lipun; voi myös lisätä uusia partitioita. Huom. vaikuttaa vasta bootin jälkeen!

# Virtuaalilevyn suurentaminen 4

```
sudo shutdown -r now
```

```
sudo resize2fs /dev/vda1; df
```

```
grep swap /etc/fstab
```

```
# cut'n'paste UUID= ... edeltä tai:
```

```
eval $(awk '/UUID.*swap/{print $1}' /etc/fstab)
```



# Virtuaalikoneen suurentaminen 5

```
sudo mkswap -U $UUID /dev/vda5
```

```
sudo swapon -a; cat /proc/swaps
```

```
shutdown -r now # varmuuden vuoksi
```

```
rm -f kone.img.bak # alustakoneessa
```

(kun on varmistettu että kaikki ok)

# Virtuaalilevyn suurentaminen 6

Jos on root-oikeudet (sudo) alustakoneessa,  
toinen tapa:

```
virsh shutdown kone
```

```
losetup -f # tulos (/dev/loop2 tms) talteen, käytetään alla
```

```
losetup -f /dev/loop2 kone.img
```

# Virtuaalilevyn suurentaminen 7

```
kpartx -a /dev/loop2 # partitiot näkyviin
```

```
eval $(blkid /dev/mapper/loop2p5 | awk '{print $2}')
```

```
# UUID talteen
```

```
kpartx -d /dev/loop2 # partitiot pois, muuttuvat alla
```

```
fdisk /dev/loop2 # partitiot, root-partition alku ennallaan
```

# Virtuaalilevyn suurentaminen 8

```
kpartx -a /dev/loop2 # uudet partitiot näkyviin  
e2fsck -f /dev/mapper/loop2p1 # varmuuden vuoksi  
resize2fs /dev/mapper/loop2p1  
mkswap -u $UUID /dev/mapper/loop2p5  
# swap UUID joka talletettiin edellä
```

# Virtuaalilevyn suurentaminen 9

```
kpartx -d /dev/loop2
```

```
losetup -d /dev/loop2
```

```
virsh start kone
```

```
# helpotuksen huokaus tai ...
```

# Virtuaalilevyn suurentaminen 10

Jos levyä joutuu suurentamaan tällä tavoin, jotain on mennyt pieleen suunnitteluvaiheessa. Vaihtoehtona kannattaa harkita uudelleenasetusta.

Jos levyjen ja osioiden ennakoitaan muuttuvan, LVM on yleensä hyvä idea.

# Ubuntun päivitys

- `sudo apt-get update; sudo apt-get dist-upgrade`
  - säännöllisesti, ja aina kun tietoturva-aukkoja tiedossa
  - automatisointi:  
`/etc/apt/apt.config.d/50unattended-upgrades`  
`dpkg-reconfigure unattended-upgrades`

# Ubuntun päivitys

- Käyttöjärjestelmäversion päivitys:  
`sudo do-release-upgrade`
  - muistia pitää olla riittävästi, samoin levytilaa
  - jos `do-release-upgrade` -komentoa ei löydy:  
`sudo apt-get install update-manager-core`



# Ubuntun päivitys

- Vain LTS-versioiden välillä vai kaikki:  
    /etc/update-manager/release-upgrades:  
    "Prompt=" Its, normal
- Päivityksen jälkeen apt-get autoremove

# for-loop, parameter expansion

```
for i in 1 3 5 7 ; do touch koe$i.txt ; done
```

```
for x in a b c ; do echo $x > ${x}koe.txt ; done
```

```
for f in *.txt ; do cp $f $f.bak ; done
```

# for-loop, parameter expansion

```
for f in *.txt ; do cp $f ${f%.txt}.bak ; done
```

```
for f in *.txt ; do cp $f ${f/txt/bak} ; done # bash
```

```
for f in a* ; do mv $f b${f#a} ; done
```

# for-loop, parameter expansion

```
for n in {0..7} ; do mkdir d$n ; done # bash
```

```
for d in d{0..7} ; do mkdir $d ; done # bash
```

```
touch koe{1..7..2}.txt # bash
```

# Uudelleensuuntaus, redirection

- Komento `>tiedosto 2>virheet`
- Komento `1>&2`
- `0=stdin, 1=stdout, 2=stderr`
- `3...9` vapaasti käytettävissä

# redirection, read

```
while read suku etu login demo ; do  
    echo $login $etu $suku; id $login  
done < kurssilaiset.txt
```

# redirection, read

```
while read suku etu login demo ; do # ei toimi!  
    echo ${login}1; ssh ${login}1 'dpkg -l | grep  
acpid'  
done < kurssilaiset.txt
```

# redirection, read

```
while read <&3 suku etu login demo ; do # toimii
    echo ${login}1; ssh ${login}1 'dpkg -l | grep
acpid'
done 3< kurssilaiset.txt
```



# 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 *laite*

# /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
```

# Run levels

- Järjestelmän "taso", järjestelmäprosessien joukko
  - SystemV: /etc/inittab
  - upstart: /etc/init/rc-sysinit.conf
  - systemd: /etc/systemd/system/\*target\*

# Run levels

- Perinteiset tasot (merkitykset vaihtelevat, etenkin 2-5):
  - 0 Halt
  - 1 (tai S) Single-user
  - 2 Multi-user
  - 3 Multi-user with networking
  - 5 Multi-user with networking and graphical UI
  - 6 reboot
- telinit *level*, telinit q (eivät nykyisin yleensä tarpeen)



# System startup & shutdown

- Useita erilaisia ja eri-ikäisiä menetelmiä:
  - Vanha Unixin init-script setup (SysV): `/etc/init.d`
  - Upstart (Ubuntu, Fedora, ChromeOS): `/etc/init`
  - Systemd (melkein kaikki nykyisin): `/etc/systemd`
- Vanhat käynnistysskriptit viritettävissä toimimaan uudempien kanssa => kaikkia näkee sekaisin

# System startup & shutdown

- SysV-init, "klassinen" init-script setup: RHEL -6 ym:  
/etc/init.d/proggis.sh [start|stop|restart...]  
/etc/rc?.d/{Snn,Knn}proggis  
– nn=runlevel

# System startup & shutdown

- upstart: Ubuntu 6.10-14.10, Fedora 9-14, ChromeOS:

`service proggis [start|stop|restart...]`

`start proggis`

`/etc/init/proggis`

# System startup & shutdown

- systemd: Fedora 15-, Debian 8-, RHEL 7-, Ubuntu 15.04- ...

init + udev + syslog + cron...

- /etc/systemd/system
- Käynnistysasetuksia: /etc/default/...