Linux-ylläpito, kevät 2014
2. luentokalvosetti 17.1 – 5.2

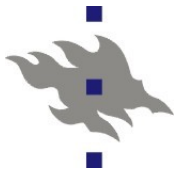
Jani Jaakkola
jjaakkol@cs.helsinki.fi
Viimeksi päivitetty 5.2.2014

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Komentorivi

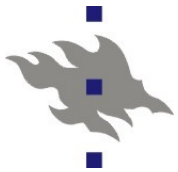
Terminaaliemulaattori

- Terminaaliemulaattoriohjelmisto:
  - Tulkitsee ja toteuttaa saamansa tekstidataan upotetut ohjauskoodit (escape sequences)
    - Esim. näytön tyhjennys: ”\033[H\033[2J”
  - Muuntaa samansa näppäinpainallukset upotetuiksi ohjauskoodeiksi
    - Esim. Home näppäin ”\033[H”
  - Ohjauskoodit on peritty muinaisista aidoista fyysisistä terminaalilaitteista (jotka nykyään on ainakin osittain standardoitu)
  - Esim, screen, xterm, gnome-terminal, kernelin virtuaalikonsoli
  - Ympäristömuuttuja *TERM* kertoo terminaaliohjelmiston tyypin
- Komentoriviohjelmistot ja shellin komentorivi toimivat virtuaalisessa päätelaitteessa (pseudo tty, pty)
  - */dev/pts/<nro>*
  - */usr/bin/tty* kertoo käytössä olevan päätelaitteen
  - Virtuaalinen päätelaite toteuttaa kaikki aidon sarjapiuhalaitteen rajapinnat (baud rate, flow control, stop bits, jne..)



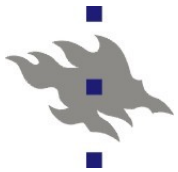
Shell

- Komentorivin ”toteutus”
  - Käyttää terminaaliemulaattorin ohjauskoodeja komentorivieditorin toteutukseen
  - Käynnistää lapsiprosesseina komentorivin komennot
    - Toteuttaa käynnistettyjen lapsiprosessien hallinnan ja syötetiedostojen ohjaukset
  - Toteuttaa ohjelmointikielen, jolla ylläpitoskriptit usein toteutetaan
    - Myös jakelupaketin mukana valmiina tulevat skriptit
    - Shelliskriptien kirjoitus ei ole välttämätön taito, mutta shelliskriptien lukutaito on
  - */bin/sh* on Bourne Shell skriptien tulkki, jonka toiminnallisuus on standardoitu. Standardiominaisuudet löytyvät myös muista Unixeista
- */bin/bash* – GNU-projektin shell
  - Ei välttämättä paras shell, mutta tämä tyypillisesti löytyy kaikkialta



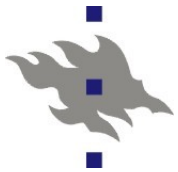
Screen

- Terminaaliemulaattori terminaaliemulaattorin sisällä
  - Daemoni, joka hallinnoi pty:illä olevia istuntoprosesseja
  - Asiakas, jolla daemonin hallinnoimiin istuntioihin pääsee käsiksi
- */usr/bin/screen*
  - uusi screen istunto
- */usr/bin/screen -r*
  - yhteys olemassaolevaan screen istuntoon
- */usr/bin/screen -rd*
  - katkaise aikaisemmat yhteydet olemassaolevaan istuntoon ja aloita uusi
- Näppäimistöoikotiet asiakkaassa näkyvän istunnon vaihtamiseen
  - **ctrl+a c** – uusi shell-istunto
  - **ctrl+a <numero>** - edustaistunnon vaihto
  - **ctrl+a d –** sulje screen-asiakas, jätä istunto taustalle (detach)
- Tuoreemmat vaihtoehdot: **tmux** ja **byobu**



Kernelin virtuaalikonsoli

- Kernelin sisään rakennettu terminaaliemulaattori
  - VGA-yhteensopivissa näytönohjaimissa käyttää näytönohjaimen tekstitilaa.
  - Käytettävissä myös GUI:n käynnistymisen jälkeen
    - X:ssä näppäinyhdistelmät ctrl+alt+f1, ctrl+alt+f2, …
    - Komentoriviltä vaihdettavissa komennolla chvt
  - Hyödyllinen minimaalisissa asennuksissa, X:n ongelmia ratkottaessa ja kernelin kaatuessa
    - Kernelin kaatuessaan tulostama stack trace näkyy konsolilla
- Kernelin Frame Buffer ajurit (fb):
  - Kernelin konsoli käyttää näytönohjaimen graafista tilaa
    - Fontteja ja näyttötilaa voi vaihtaa
    - Nykyisissä työpöytädistribuutioissa otetaan heti käyttöön
  - Näyttötila voi vaihtamaa myös BIOS-kutsuilla, jossa sama BIOS toimii kaikilla näytönohjaimilla
- Sarjakonsolit edelleen käytössä upotetuissa laitteissa

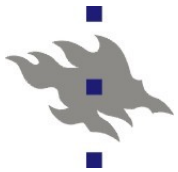


Etähallintakonsolit

- DRAC
  - Dell Remote Access Console
  - Dellin palvelimilla oleva fyysinen lisäkortti
  - Kortilla on kokonainen erillinen tietokone, verkkokortilla, tcp/ip-pinolla ja http-palvelimella varustettuna
  - Kaappaa palvelimen VGA-portin ja ohjaa sen edelleen verkkoon
  - Hiiren ja näppäimistön käyttö etänä
  - Sammutus, käynnistys ja resetointi etänä
  - Mahdollistaa verkossa sijaitsevan optisen media ja/tai usb-median liittämisen etähallittuun koneeseen, aivan kuin se olisi koneessa fyysisesti kiinni
  - Monitorointi
- Virtuaalikoneen konsoli
  - Virtuaalinen näytönohjain
  - Virtuaalisen näytönohjaimen kuva välitetään graafiselle työpöydälle (esim. vnc-protokollalla)

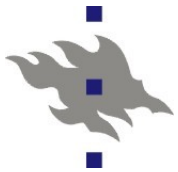
Ssh-protokolla

- Ssh-palvelin on ylläpitäjän tärkein etäylläpitotyökalu
  - Joskus jopa ainoa
  - Tatu Ylösen kehittämä perinteiset telnet, rlogin ja rsh komennot (ja protokollat) korvaava ohjelmisto
- Kryptaa kaiken verkkoliikenteen
- Palvelimen ja asiakkaan identiteetin varmistus
- Voi autentikoida myös salaisella avaimella
- Kerran autentikoidulla ssh-yhteydellä voidaan
  - Avata Login-istuntoja tai suorittaa komentoja
  - Tunneloida TCP-yhteyksiä
  - Tunneloida X-asiakkaita
  - Etäkäyttää tiedostoja (sftp)
- Monet komentorivin työkalut suunniteltu toimimaan ssh-yhteyden yli
  - *rsync*, *cvs*, *svn* ja *tar*



Ssh

- Ssh-ohjelmistot
  - ssh.com – se alkuperäinen, nykyään kaupallinen
  - f-secure/WRQ kaupallinen ssh-asiakas
  - Lsh – ssh GNU-lisenssillä
  - Putty – portattava OS-asiakas. Hyödyllinen windows- ja kännykkäkäyttäjille
  - Openssh – OpenBSD-projektin vanhasta ssh-v1:stä forkkaama versio. Tätä kaikki distribuutiot käyttävät
- Linux-verkkoa pystytettäessä mietittävää
  - Palvelinten julkisten avainten hallinta
  - Ylläpitoavaimet
  - Ssh-agent
  - Single sign on kerberos-lautentikoinnilla

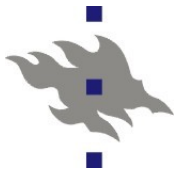


Openssh

- OpenBSD-projektin edelleen kehittämä versio vanhasta Tatu Ylösen OS ssh:sta
- Ominaisuudet
  - Tuki ssh-protokollan versioille 1 ja 2
  - Asiakas- ja palvelintuki
  - *scp* yksinkertaiseen tiedostojen kopiointiin
    - perintökalu joka ei suostu kuolemaan (rcp)
  - Sftp-palvelin
  - Yksinkertainen komentorivin sftp-asiakas
  - *ssh-agent* daemoni salaisten avainten säilytykseen
  - Kerberos-tikettien uudelleenohjaus
  - Valmiiksi autentikoidun yhteyden jakaminen
  - Nykyään myös VPN-tuki
    - Tosin IP-liikenteen tunnelointi TCP-putken yli ei ole välttämättä hyvä idea

Autentikointi ssh-avaimilla

- Ssh-protokolla osaa käyttää salaisia avaimia käyttäjän autentikointiin
  - Ja ohittaa niillä normaalin salasana-autentikoinnin
- Käyttäjä tai ylläpito luo avainparin
  - Julkinen avain talletetaan kohdekoneen käyttäjän kotihakemistoon *.ssh/authorized_keys* -tiedostoon
  - Salainen avain talletetaan levylle
    - Mielellään salasanalla kryptattuna
- Erillinen ohjelma säilyttää salaiset avaimet koneen RAM-muistissa
  - *Ssh-agent*
  - *gnome-keyring*
- Ssh-asiakas osaa käyttää levyltä tai agentilta löytyviä salaisia avaimia
  - Ssh-protokolla osaa myös tunneloida salaiset avaimet etäkoneilla (openssh -A vipu)



Ssh:n ongelmia

- Ei PKI-infrastruktuuria avainten jakamiseen ja allekirjoittamiseen
  - openssh:lle on X509-sertifikaattituen toteuttava paikka
  - Laitoksella 777 riviä */etc/ssh/ssh_known_hosts* -tiedostossa
  - Avaimet nimipalvelussa
- Salaisia ssh-avaimia pitää suojella yhtä hyvin kuin salasanoja
  - Levyllä salaiset avaimet pitäisi säilyttää kryptattuna
- X-yhteyksiä ei pitäisi tunneloida oletusarvoisesti
  - X on monipuolinen protokolla, jonka yli voi tehdä (ainakin) näppäinpainallusten lokia ja ruudunkaappauksia
- Vastaavasti kerberos tikettien ja *ssh-agent* yhteyksien tunnelointi voi olla vaarallista
- FUSE:n avulla kernel-tason tuki sftp-protokollalle
- Ssh-palvelinten ”koputtelu” on pysyvä ongelma
  - Löydettyjä salasanoja tai kryptaamattomia ssh-avaimia käytetään muiden palvelinten murtamiseen

Openssh:n konfigurointi

- Sshd:n konfiguraatiotiedosto */etc/ssh/sshd_config*
  - PAM:in autentikoinnin konfiguraatiotiedosto */etc/pam.d/sshd*
  - Openssh:n autentikointiprosessi ei oletusarvoisesti toimi rootin oikeuksin, vaan erityisen autentikointia varten olevan käyttäjätunnuksen alla
- Käyttäjän kotihakemistossa
  - *.ssh/authorized_keys*
  - Käyttäjän omat luotetut ssh-avaimet ja niiden parametrit
- Ssh-asiakkaan konfiguraatiotiedosto */etc/ssh/ssh_config*
  - Tai kotihakemiston *.ssh_config*
  - Kohdekonekohtainen konfigurointi

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Käyttäjätunnukset

Käyttäjien hallinta

- Työasemakäyttäjä tarvitsee:
  - Kotihakemiston
  - Rivin */etc/passwd* -tiedostossa
    - Kotihakemisto, UID, GID, SHELL
  - Rivin */etc/shadow* -tiedostossa
    - yksisuuntaisen hash-funktion läpi ajettu salasana ja salasanan ja tunnuksen voimassaoloaika
  - Ryhmät */etc/group* -tiedostossa
    - Nykylinuxeissa on tapana perustaa jokaiselle käyttäjällä oma ryhmä
- */usr/sbin/adduser* ja */usr/sbin/addgroup*
  - Käyttäjätunnusten ja ryhmien lisäys
- */usr/sbin/deluser* ja */usr/sbin/delgroup*
  - Vastaavasti tunnusten ja ryhmien poisto
- */usr/sbin/chage*
  - *Shadow-tiedoston käsittely*
- */usr/bin/passwd*
  - Salasanan vaihto

Shadow-tiedosto

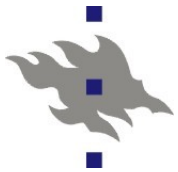
- *sp_lstchg:*
  - Päivä jolloin salasana edellisen kerran vaihdettiin
  - Mitattuna päivinä vuoden 1970 alusta
- *sp_min*
  - Minimimäärä päiviä, joiden on kuluttava ennen kuin salasanan voi vaihtaa uudelleen
- sp_max
  - Monenko päivän jälkeen salasanan vaihtoon pakotetaan
- *sp_warn*
  - Monenko päivän jälkeen varoitetaan salasanan vanhenemisesta
- *sp_inact*
  - Monenko päivän jälkeen salasanaa ei voi enää vaihtaa ja tunnus vanhenee
- *sp_expire*
  - Päivä jolloin tunnus vanhenee

Sisäänkirjautuminen

- Sisäänkirjautumisen toteuttavat erilliset root-oikeuksin pyörivät sovellukset
  - PAM-rajapinta: Linuxin autentikoinnin toteuttava kirjasto ja plugin-järjestelmä
- Display Manager
  - X-palvelimen alla toimiva graafisen autentikoinnin toteuttava sovellus
    - Ubuntulla lightdm, Gnome-projektilla *gdm*, KDE-projektilla *kdm* ja vielä vanha perinteinen *xdm*
- Ssh
  - Etäkirjautuminen shell-istuntoon
  - PAM-autentikoinnin lisäksi omat ssh-autentikointimekanismit
- getty ja */bin/login*
  - Kernelin virtuaalikonsolin tai sarjapäätteen alustus ja autentikointi
- */bin/su* ja */bin/sudo*
  - Käyttäjätunnuksen vaihto komentoriviltä

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Ohjelmistopakettitietokannat



Ohjelmistopakettitietokannat

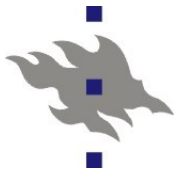
- Tyypillinen Linux-distribuutio koostuu useista sadoista erilaisista ohjelmistoista ja oheistiedostoista
- Ohjelmistojen välillä on riippuvuuksia
  - Ohjelmistot tarvitsevat toimiakseen kirjastoja, palveluita, tiedostoja, toisia ohjelmia, jne...
    - Tavallisesti vielä jonkin tietyn version
- Distribuution mukana asentuu pakettitietokanta
  - Jokainen tiedosto on osa jotain ohjelmistopakettia
  - Ohjelmistopakettitietokanta pitää yllä tietoa asennetuista ohjelmistopaketeista, versioista, pakettien sisältämistä tiedostoista ja pakettien välisistä riippuvuuksista
  - Pakettienhallintaohjelmat ylläpitävät tietokantaa ja voivat kieltäytyä asentamasta ohjelmistoja, jotka eivät puuttuvien riippuvuuksien vuoksi toimisi
  - Metapaketit eivät itse tee mitään, mutta sisältävät ohjelmistokokonaisuuksia riippuvuuksinaan

Ajoaikaiset päivitykset

- Unixeissa auki olevan tiedoston voi poistaa tiedostojärjestelmästä
  - Jos prosessilla on muistiavaruudessaan kirjastotiedosto, voi tiedoston poistaa näkyvistä tiedostoavaruudesta, ilman että prosessin muistiavaruudelle käy huonosti
  - Paketinasennusohjelma kirjoittaa asentamansa tiedostot tilapäisillä nimillä, sitten suorittaa *rename()* -systeemikutsun, jolla vanhat tiedosto mahdollisesti poistetaan tieltä
  - Vanhat tiedostot siivotaan pois viimeiseksi
  - Jos uudet tiedostot ovat yhteensopivia vanhan käynnissä olevan ohjelmiston kanssa tai jos vanha käynnissä oleva ohjelmisto ei edes yritä avata niitä, järjestelmä jatkaa toimintaansa normaalisti
  - Vanhat tiedostot voivat jäädä roikkumaan; lopullisesti ne poistuvat vasta kun kone käynnistetään uudelleen

Pakettivarasto (repository)

- Distribuution oletusasennuksen mukana tulee rajallinen määrä ohjelmistoja
- Distribuutio tarvitsee mekanismin ohjelmistopäivityksiin
- Ratkaisuna verkossa sijaitsevat pakettivarastot
  - Pakettivarasto sisältää oletusasennuksen ulkopuoliset ohjelmistot ja myöhemmin tehdyt päivitykset
  - Metatietona lista kaikista paketeista, pakettien riippuvuuksista ja versionumeroista
  - Distribuution mukana tulee ohjelmisto, joka osaa hakea ja asentaa ja päivittää ohjelmistopaketteja pakettivarastoista
    - Ohjelmisto myös ratkoo pakettien väliset riippuvuudet
  - Debian: apt, Fedora: Yum, Suse: Yast
- Distribuution ulkopuoliset pakettivarastot
  - Ohjelmistoille, joita ei voi jakaa distribuution mukana
  - Multimediakoodekit, 3. osapuolen ohjelmistot, omat paketit



RPM-pakettitietokanta

- RPM: Red Hat Package Manager
  - Alunperin RedHat-1.0 distribuutiolle vuonna 94
- RPM-pakettitietokannan osat
  - *.rpm* -tiedostoformaatti
  - Tietokanta asennetuista paketeista: */var/lib/rpm*
  - Komentoriviohjelmisto pakettien hallintaan: */usr/bin/rpm*
  - Kirjasto rpm-paketteja tai tietokantaa käyttävien ohjelmistojen käyttöön
- RPM-paketin sisältö:
  - Paketin varsinaiset tiedostot (pakattu cpio arkistotiedosto)
    - Konfiguraatio- ja dokumentaatiotiedostot on erikseen merkattu
  - Paketin metatiedot (tags): versio, arkkitehtuuri, jne
  - Paketin vaatimat ja tarjoamat riippuvuudet
    - Ja konfliktit: kaikki ohjelmistot eivät ole yhteensopivia
    - Lista vanhoista paketeista, jotka uusi paketti korvaa kokonaan
  - Asennuksen ja poiston yhteydessä suoritettavat skriptit

RPM: Digitaaliset allekirjoitukset

- Ohjelmistopaketteja asennettaessa on jotenkin voitava varmistua paketin sisällöstä ja alkuperästä
- RPM-paketit voivat sisältää osanaan digitaalisen allekirjoituksen
  - Distribuutioiden pakettivarastojen paketit sisältävät aina allekirjoituksen
- Järjestelmän pakettitietokantaan taas on talletettu distribuution julkinen avain
- Paketteja asennettaessa paketin sisäistä allekirjoitusta verrataan julkiseen avaimeen

Yum pakettivarastonhallinta

- Fedoran pakettivaraston ja päivitykset toteuttaa *yum*
- Yum hakee pakettivarastosta ajantasaisen listan ohjelmistopaketeista ja asentaa sen pohjalta päivitykset ja halutut uudet ohjelmistot
  - Yum ratkoo myös ohjelmistoriippuvuudet ja automaattisesti asentaa tarvitut ohjelmistot
  - Yum tallettaa pakettivarastosta haetun metadatan ohjelmistoriippuvuuksista välimuistiin
    - Jos metadata ei ole muuttunut, ei turhaan ladata sitä uudelleen
- Oletusarvoisesti yum suostuu asentamaan vain allekirjoitettuja paketteja
  - Kuka tahansa voi peilata (mirror) pakettivaraston: paketin allekirjoitus kertoo onko paketti luotettava
- Käytössä olevat pakettivarastot listataan konfiguraatiohakemistossa */etc/yum.repos.d/*

RPM-paketit komentorivillä

- Paketin asennus: *rpm -i <paketti>*
- Paketin päivitys: *rpm -U <paketti>*
- Paketin poisto: *rpm -e <paketti>*
- Tietoja paketista: *rpm -qi <paketti>*
- Paketin sisältämät tiedostot: *rpm -ql <paketti>*
- Mikä paketti sisältää tiedoston: *rpm -qf <tiedosto>*
- Paketin asennus- ja poistoskriptit: *rpm -q –scripts <paketti>*
- Asennettujen pakettien listaus: *rpm -qa*
- Järjestelmän päivitys ajan tasalle: *yum update*
- Ohjelmistopaketin asennus varastosta: *yum install <paketti>*
- Kaikkien saatavilla olevien pakettien listaus: *yum list available*
- Päivitysten listaus: *yum list updates*

.deb -paketit

- Debian-distribuution käyttöön kehitetty pakettiformaatti
- *.deb*-paketti on *ar* kirjastotiedosto, jonka sisältä löytyy kolme tiedostoa
  - d*ebian-binary:* pakettiformaatin tunnistus ja versionumero
  - *control.tar.gz:* paketin metadata
    - Asennuksessa ja poistossa suoritettavat skriptit
    - *Control:* paketitn kuvauksen ja metadatan sisältävä tekstitiedosto
    - Tarkastussummat
    - Listan konfiguraatiotiedostoista
      - Konfiguraatiotiedostoja ei ylikirjoiteta pakettia päivitettäessä
  - *data.tar.gz*: paketin sisältämät varsinaiset tiedostot

dpkg-paketinhallinta

- */usr/bin/dpkg* on *.deb*-pakettien matalan tason hallintatyökalu
  - Pakettien asennus
  - Pakettien poisto
  - Pakettitietokannan hallinta
- Dpkg osaa varmistaa, että pakettien asennus tai poisto täyttää pakettien väliset riippuvuusvaatimukset
  - Mutta ei automaattisesti hakea tarvittavia uusia paketteja
  - Kuten */usr/bin/rpm*
- Asennettujen pakettien tietokanta */var/lib/dpkg*
  - Metatiedot isoissa tekstitiedostoissa
  - Asennusskriptit ja listat talletettuna */var/lib/dpkg/info* -hakemistoon erillisiin tiedostoihin
- Daemonien hallinta
  - */sbin/start-stop-daemon* asennusskripteille

Debconf

- Monia deb-paketteja konfiguroidaan debconf -järjestelmällä
- Paketin mukana tulee tiedosto, joka sisältää kysymyksiä, joihin ylläpitäjän täytyy vastata
- Vastausten perusteella erillinen skripti konfiguroi paketin (tyypillisesti muokkaamalla konfiguraatiotiedostoja)
- Kysymyksiin voi kuitenkin vastata etukäteen: täysin automaattiset asennuksen ovat mahdollisia
- */usr/bin/dpkg-reconfigure* komennolla voi myöhemmin konfiguroida paketit uudelleen

APT-paketinhallinta

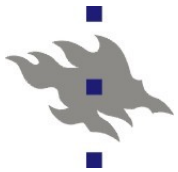
- Kuten *yum* (*apt* toki oli olemassa ensin)
  - Hakee pakettilistat ja pakettien metadatan debian-pakettivarastosta
  - Ratkoo riippuvuudet ja asentaa riippuvuuksien vuoksi tarvittavat lisäpaketit
  - Välimuistissa olevat tieto pakettien metadatasta pitää päivittää erikseen
  - Pakettivarastojen konfiguraatiotiedosto */etc/apt/sources.list*
- Käyttö:
  - *apt-get update:* pakettilistojen päivittys
  - *apt-get install <paketit>*: pakettien asennus
  - *apt-get remove <paketit>:* pakettien ja niistä riippuvien pakettien poisto
  - *apt-get -u upgrade:* päivitysten asennus ja niiden listaus

Apt-secure allekirjoitukset

- Apt-pakettivarasto sisältää *Release*-tiedoston, joka sisältää kaikkien pakettivaraston tiedoston tarkastussummat
  - Tiedosto R*elease.gpg* sisältää *Release*-tiedoston gpg-allekirjoituksen
  - Apt tarkastaa allekirjoituksen ja suostuu asentamaan vain paketteja, joiden tarkastussumma on sama kuin *Release*-tiedostossa listattu tarkastussumma
- *debian-archive-keyring* -paketti sisältää debian-pakettivarastojen luotetut julkiset avaimet
  - *ubuntu-archive-keyring* ubuntu-paketeilla
  - Mahdollistaa avainten automaattisen päivityksen

PPA: Personal Package Archive

- Ubuntun mekanismi varsinaisen jakelupaketin ulkopuolisten pakettivarastojen hallinnointiin ja ylläpitoon
  - Ubuntu-projekti tarjoaa palvelintilan ja pakettivaraston ylläpidon
- Jokaisella PPA-pakettivarastolla on oma allekirjoitusavain, joka on vain Ubuntu-projektin hallussa
- Ubuntu-asennuksen mukana tulee työkalut PPA-pakettivarastojen aktivointiin
  - *sudo add-apt-repository ppa:user/ppa-name*
  - Aktivointi lisää rivin */etc/apt/sources.list.d* hakemistoon ja asentaa allekirjoitusavaimen julkisen puoliskon



RPM vs DEB

- RPM käyttää indeksoituja binääritiedostoja, deb tavallisia tekstitiedostoja
  - Rpm on nopeampi
  - deb-ongelmia joskus helpompi ratkoa
- Molemmat tukevat symbolisia riippuvuuksia
  - RPM tukee riippuvuuksia myös tiedostoista
- RPM-paketit sisältävät digitaalisen allekirjoituksen
  - Debianin mallissa vain pakettivarastot allekirjoitetaan
- RPM-pakettien asennus ei (lähes) koskaan edellytä ylläpitäjän interaktiota
  - Debconf-järjestelmä voi alkaa kysellä tyhmiä kesken pakettien asennuksen

Debian pakettien luonti

- Debian paketin metatiedot sijaitsevat ohjelmiston lähdekoodihakemiston alihakemistossa *debian/*
- Komennolla d*pkg-buildpkg* rakennetaan metatietohakemiston konfiguraation perusteella *.deb* asennuspaketti
  - Myös debian lähdekoodipaketti ja *.dsc* kuvaustiedosto
- Paketin metatiedot tiedostossa *debian/control*
- Paketin rakennusprosessi tiedostossa *debian/rules*
- Paketin asennushakemisto *debian/<ohjelmannimi>*

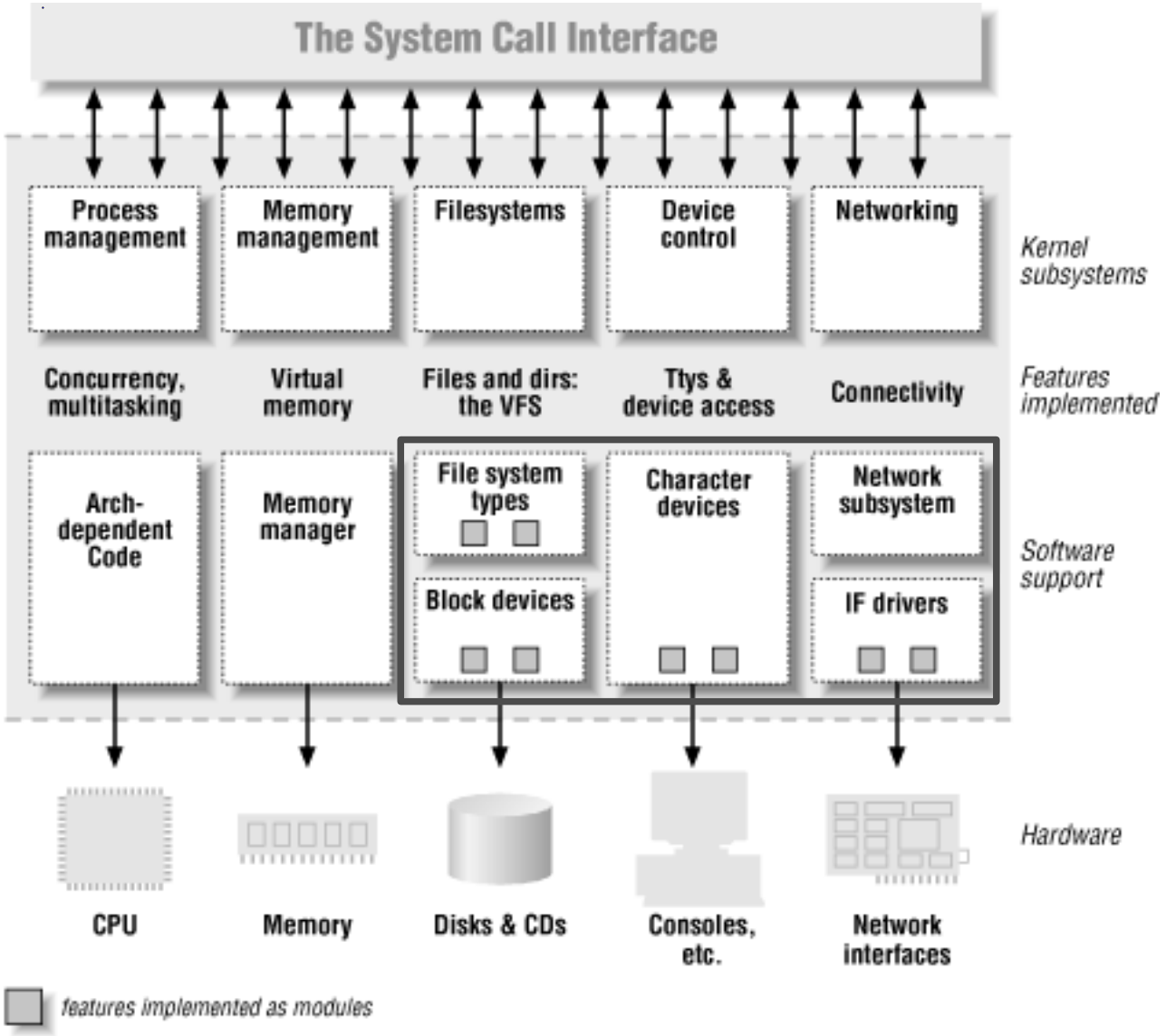
RPM-pakettien luonti

- Pakettien kokoamiseen tarvittava ympäristö:
  - *yum install @development-tools*
  - *yum install fedora-packager*
- Paketit rakentavan käyttäjän hakemistohierarkia
  - *rpmdev-setuptree*
- Paketin metatiedon ja kokoamisen *.spec* -tiedosto *SPECS/ -alihakemistoon*
- Lähdekoodipaketit *SOURCES/* alihakemistoon
- Paketin kokoaminen *rpmbuild -ba <tiedosto.spec>*
- Alihakemistoon *RPMS* ilmestyy valmis paketti
- Alihakemistoon *SRPMS* ilmestyy lähdekoodipaketti
- Paketin kokoamisprosessissa paketti asentuu *BUILDROOT* alihakemistoon

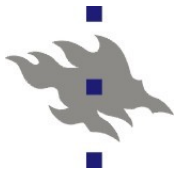
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Linux kernel

Käyttöjärjestelmän tehtävä



Linux Device Drivers, 2nd Edition, By Alessandro Rubini & Jonathan Corbet, 2nd Edition June 2001



Kernelin vastuualueet

- Laitteistoajurit
- Prosessit, säikeet ja skedulointi
  - Kernel-tason kevyet säikeet
  - SMP-suorituskyky
    - Prosessien keskeytys
    - “Pienet”-kernel lukot
    - NUMA: Non Uniform Memory Architecture
- Muistinhallinta
  - Tuki 64-bittiselle muistiavaruudelle (myös 32-bit kernelillä)
  - Tuki 32-bittisille prosesseille 64-bittisellä alustalla
- Tiedostojärjestelmät
  - Ext4, Brtfs
- Verkkotiedostojärjestelmät
    - NFS, Smbfs
- Fuse: Tuki käyttäjätason tiedostojärjestelmille (ssh, ntfs)

Lisää kernelin vastuualueita

- Verkko
  - IPv4 ja IPv6-protokollapino
  - Ethernet, wlan ja point2point-protokollapinot (esim. PPP)
  - Pistokkeet, palomuuri, reititys (ja siltaus), ipsec
  - Tuki user space -tason verkoille (tunnelointi, VPN, virtuaalikoneiden verkot)
- Virransäästö
  - ACPI rajapinnan käyttöjärjestelmäpuolen toteutus
  - Suspend to RAM, hibernointi
  - CPU:n kellotaajuuden säätö ja virransäästötilat
  - Dynaamiset kellokeskeytykset
- Virtualisointi
  - Rajapinnat, jolla käyttäjätasolla voi tehokkaasti emuloida laitteistoa

Kernel ja laitteisto

- Kernel tukee isoa joukkoja laitearkkitehtuureja
  - Intel IA32, IA64, ARM, Power, …
  - … toki vain yhtä kerrallaan
- Kernelin laiteajurit
  - Väylät: PCI, USB, SCSI, Firewire, Thunderbolt, i2c
    - laitteiden listaus, alustus, virransäästö
  - Verkkolaitteet
    - Langattomat ja langalliset verkkokortit
  - HID (Human Interface Device) rajapinta
    - Näppäimistöt, hiiret, peliohjaimet, kaukosäätimet
  - ALSA (Advanced Linux Sound Architecture)
    - Äänikorttiajurit ja kernelin audio API
  - Levyohjaimet
    - IDE/SATA, SCSI, USB, Firewire, FC (multipath), RautaRAID
  - ...

Kernel: modulit

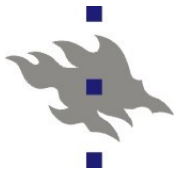
- Suurin osan ytimen toiminnoista käännetään moduleiksi, jotka ladataan vain tarvittaessa
  - Laiteajurit
  - Tiedostojärjestelmät
  - Kokonaiset alijärjestelmät (virtualisointituki, SCSI-protokolla)
  - On mahdollista rakentaa monoliittinen kerneli ilman modulitukea (tätä ei kai kukaan enää tee)
  - Mahdollistaa saman binääri ytimen käytön eri laiteympäristöissä ja konfiguraatioissa
- Modulit eivät ole siirrettäviä kernelin eri versioiden välillä
  - Jos käännösaikainen kernelin konfiguraatio vaihtuu, ei edes saman version välillä
  - DKMS on järjestelmä joka käynnistymisen yhdeydessä kääntää(!) moduleja uuteen kernelin kofiguraatioon
- Udev-daemoni automatisoi ajurimodulien latauksen
- Moduleja voi ladata ja poistaa myös manuaalisesti

Kernel: module-init-tools

- Module-init-tools sisältää työkalut modulien käsittelyyn
  - *modinfo*: modulin metadatan listaus
  - *modprobe*: modulin lataus ja argumentit
  - *rmmod:* modulin poisto
  - *lsmod:* listaa ladatut modulit
  - *depmod*: luo riippuvuushierarkian ja indeksit
- Konfiguraatiotiedosto */etc/modprobe.conf*
  - Muinoin täällä listattiin modulialiaksia, joiden avulla varsinainen haluttu ajuri löytyi
    - Esim. snd-card-0 oli alias ensimmäisen äänikortin ajurille
  - Nykyään lisätään mustalle listalle modulit (black list), joita ei haluta ottaa käyttöön
    - Esim. näytönohjainten framebuffer ajurit
  - Modulien parametrien konfigurointi
  - Voidaan konfiguroida skriptejä, joita suoritetaan ennen modulin latausta ja sen jälkeen

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Laitteisto ja laitteistolistaukset



Emolevy

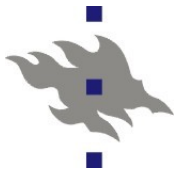
- Emolevyn piirisarja (chipset) toteuttaa PC:n laitteiston hallinnan
  - CPU, väylät, muistit
- Nykyään lisälaitteet ovat emolle integroituja
  - Äänikortti
  - Verkkokortit (myös langattomat)
  - Levyohjaimet (SATA-ohjain)
  - Integroidut näytönohjaimet
- Muistipaikat
- CPU-paikat
- BIOS
  - PC:n käynnistys ja emon laitteiden alustus
  - Virransäästöominaisuudet:
    - laitteiston unitilat
    - Suspend to RAM

Muistiavaruus

- Laitteet ovat nykyään 64 bittisiä
  - 32-bittisellä muistiavaruudella voi osoittaa vain 4G muistia
  - Alle 4G muistilla varustettua palvelinrautaa ei enää myydä
  - Toisaalta 64-bittinen muistiavaruus edellyttää 64-bit osoitinten käyttöä C- ja C++-ohjelmistoissa
    - 32-bit ja 64-bit koodi eivät ole binääriyhteensopivia: 32-bittisestä koodista ei voi tehdä kutsuja 64-bittiseen kirjastoon ja päinvastoin
    - Täsmälleen sama koodi käyttää 64-bittisenä hieman enemmän muistia
  - Pelkästään 32-bittisiä sovelluksia ei ole enää juurikaan jäljellä
    - Lähinnä pelejä
  - 32-bittiset sovellukset toimivat hyvin 64-bittisellä kernelillä, jos 32-bittiset kirjastot on asennettu
  - 64-bittisellä alustalla voi pyörittää 32-bittistä virtuaalikonetta

Muisti ja muistiväylä

- Muistiväylän nopeus ja latenssi on oleellisin koneen tehoon vaikuttava tekijä heti CPU:n jälkeen
  - Virheenkorjaava ECC muisti on kallista, mutta usein välttämätöntä palvelinkäytössä
- Vanha DDR2 SDRAM
  - double-data-rate synchronous dynamic random access memory
  - Taajuudet 400MHz - 1600Mhz
  - 800MHz tyypillinen nykykoneissa, 6400MB/s siirtonopeus
    - 5ns latenssi
- DDR3 SDRAM
  - double-data-rate type three synchronous dynamic random access memory
  - Enemmän kaistaa, latenssit yhtä pitkät, käyttää vähemmän virtaa
  - Taajuudet 800MHz – 1600MHz
    - 1.3GHz taajuudella 6 ns latenssi, 12800MB/s siirtonopeus



CPU

- Tärkein laitteen tehoon vaikuttava tekijä
- Nykyisissä CPU:issa on tyypillisesti ainakin 4 ydintä
  - Erityisesti palvelinraudassa
    - Palvelinraudassa myös useampi CPU / laite
  - Yhden ytimen CPU:t ovat jääneet historiaan
  - Virtualisointi on helpoimpia keinoja hyödyntää useampia ytimiä
  - Yksittäisen ytimen nopeus ei ole kasvanut juurikaan viime vuosina
  - Kehitys on tapahtunut virransäästössä ja ydinten lukumäärässä
- Intel CPU
  - Selkeästi nopeampia / ydin kuin AMD:n tuotteet
  - High-end ei kilpailijoita
  - Hidas mutta virtapihi näytönohjain mahdollisesti integroitu CPU:lle
- AMD CPU
  - AMD on panostanut ydinten lukumäärän kasvatukseen ja hintaan
  - AMD APU: CPU-sirulle integroitu (tehokkaampi) näytönohjain
- Intel ja AMD molemmat kehittävät itse Linux-ajurit
  - Ei yhteensopivuusongelmia Linuxin kanssa

Laitteistolistaukset: *dmidecode*

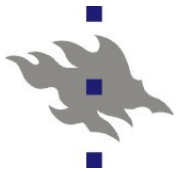
- Komento *dmidecode* listaa BIOS:in mielipiteen PC-laitteiston kokoonpanosta
  - Lukee tiedot BIOS:in muistialueelta: toimii vain root-käyttäjällä
  - Kernel käyttää samoja tietoja alustaessaan koneen muistia ja CPU:ita
- Listaus sisältää
  - Laitteen valmistajan, tyypin ja sarjanumerot
  - Emon valmistajan, tyypin ja sarjanumerot
  - CPU:n valmistajan, tyypin, tunnistetiedot, ominaisuudet ja sarjanumeron
    - CPU-ytimien lukumäärän
  - Listauksen integroiduista laitteista ja laitepaikoista
    - Myös niistä jotka eivät ole käytössä!
  - Muistin konfiguraation, muistikammoittain jaoteltuna
    - Kampojen tyypit, valmistajat ja sarjanumerot
  - BIOS-ohjelmiston tiedot (valmistaja ja versio)

Väylät

- Väylät toteuttavat tiedonsiirron laitteiden välillä
  - Väylä on itse laite: se tarvitsee ajurin
  - Väylän tiedonsiirtonopeus ja latenssi määrittelevät pitkälti laitteiston tehokkuuden
- Väylä ja sen ajuri toteuttavat
  - Tiedonsiirron
  - Laitteistolistaukset ja notifikaatiot laitteistomuutoksista
  - Virransäästön hallinnan
    - Väylällä olevien laitteiden nukutus ja sammutus
    - Itse väylän nukutus ja sammutus
- Muistiväylä: muistien ja CPU:iden välinen kommunikointi
- PCI: CPU:n näkymä laitteistoon
- (e)SATA: massamuistilaitteiden väylä (kovalevyt, flash)
- USB: ulkoisten laitteiden väylä
- SCSI: palvelinlaitteiston väylä konehuoneen sisällä
  - Fibre channel, iSCSI
- Ethernet: koneiden välinen verkko

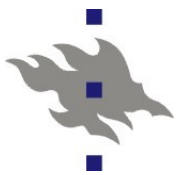
PCI-väylä

- Emolevyllä oleva väylä, jonka CPU näkee suoraan
  - Peripheral Component Interconnect
- Emolle integroidut lisälaitteet tyypillisesti ovat kiinni emon sisäisessä PCI-väylässä
- Lisäksi emolla on PCI-väyläpaikkoja erillisille lisälaitteille
  - Perinteisen 33MHz PCI-väyläpaikan nopeus 133MB/s tai 64-bittiselle väyläpaikalle 266MB/s
    - PCI 2.2 toi 66 MHz väyläpaikat
  - Mini PCI-korttipaikka kannettavien koneiden sisäisille lisälaitteille
    - WLAN-kortti tai 3G-kortti tyypillisesti Mini PCI-paikassa
  - Kannettavien PC-card korttipaikat ovat PnP PCI-väyläpaikkoja



PCI-väyläpaikkoja

- AGP väyläpaikat toivat lisää nopeutta muistista näytönohjaimelle
  - Vanhempia 3D-näytönohjaimia varten
- PCI-X väyläpaikkoja löytyi vanhoista palvelinlaitteista
  - 1064MB/s
- PCI-E (Express)
  - Moderni väylä paljon siirtotehoa tarvitseville laitteille
  - PCI-E 1.x nopeus 250MB/s
  - PCI-E 2.0 nopeus 500MB/s
  - PCI-E 3.0 nopeus 1000MB/s
  - Erityisesti 3D näytönohjaimille
  - Myös levyohjaimia ja verkkokortteja
  - PCI-E on vähitellen syrjäyttämässä perinteiset PCI-laitepaikat. Uusissa emoissa ei vanhoja PCI-paikkoja välttämättä ole
- PCI-väylän laitteiden listaus: *lspci*



USB, Firewire, E-sata

- USB ja Firewire ovat väyliä erilaisten ulkoisten laitteiden kiinnittämiseen
  - USB- ja Firewire ohjaimet taas ovat kiinni PCI-väylällä
  - Standardoidut ohjainrajapinnat: Tarvitaan vain muutama standardiajuri
- USB2 ja Firewire ovat hitaita:
  - USB2 väylän max nopeus 60MB/s
    - Käytännössä rajoittaa USB2-kovalevyjen nopeuden 25MB/s kohdalle
  - Firewire väylän nopeus 3200Mbit/s
- Uusissa laitteissa on USB3 väyläpaikat
  - Väylänopeus 4.8Gbit/s
  - Ei enää ole pullonkaula perinteisille kovalevyille
  - Voi olla pullonkaula flash-levyille
- USB-laitteiten listaus: *lsusb*
- External SATA ulkoisille kovalevyille
  - 3Gbit/s
  - Tämä varmaan häviää USB3:n myötä nopeasti

SCSI-väylä

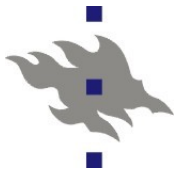
- Perinteiset fyysiset SCSI-väylät ovat jääneet historiaan
- SCSI-protokolla sen sijaan elää ja voi hyvin
  - SATA-levyt ja väylät käyttävät SCSI-protokollaa
  - Storage Area Network (SAN) on moderni versio SCSI-väylästä, missä konehuoneen sisällä jaetaan erillisen verkon ja kytkinten avulla laitteita konehuoneen sisällä
    - Levypalvelimet, nauhurit ja palvelimet ostetaan erikseen, kaikki kytketään yhteen samaan SAN-verkkoon
  - SCSI-protokollaa voi käyttää ethernet verkon yli
    - ISCSI-protokolla
- ESATA- ja USB-massamuistilaitteet käyttävät SCSI-protokollaa
- Komento SCSI-laitteiden listaukseen: *lsscsi*

Laitteiston valinta: emolevy

- Emolle integroidut laitteet tarvitsevat jokainen oman ajurinsa:
  - Äänikortin, verkkokortti, SATA-ohjain
  - Näytönohjain
- Virransäästöominaisuudet eivät aina ole yhteensopivia
- Eri valmistajien tuki Linuxille vaihtelee
  - Intel
    - Kaikki laitteet tuettuja, myös näytönohjain
    - Ajurit avoimia, valmiiksi mukana distribuutioissa
    - Intel testaa ja kehittää ajureita itse
    - Kalliimpi, eikä ole vaihtoehto jos haluaa AMD:n prosessorin
  - ATI/AMD
    - ATI/AMD:n kehittämiä ja testaamia
    - Laitteet hieman halvempia
  - 3. osapuolen emot (Asus, jne)
    - Tuki vaihtelee

Laitteiston valinta: palvelin

- Palvelinraudan Linux tuki on erinomaista
  - Palvelinvalmistajat (esim Dell, HP, IBM) pyrkivät pitämään palvelinraudan ajurit distribuutioiden oletusasennuksissa mukana
  - Tai ainakin toimittavat asennukseen tarvittavat erilliset ajurit
- Seuraavien laitteiden ajurituki kannattaa selvittää:
  - SCSI-ohjaimet
  - RAID-ohjaimet
  - Fibre Channel-ohjaimet
  - Verkkokortit
- Erillisiä softa-asennuksia tarvitaan
  - Laitteiston monitorointiin (RAID-levyjen tila)
  - Firmware-päivityksiin



Laitteiston valinta: näytönohjain

- Intel
  - Intel-ajuri on vakaa, avoin, 3D-kiihdetty
  - Hidas (mutta niin on rautakin)
- AMD/ATI
  - Suljettu, pikkuongelmia
  - 3D jokseenkin yhtä nopeaa kuin Windows-puolella
  - Helposti paketoitavissa
  - Speksit kehittäjien saatavilla: on mahdollista että kelvollinen avoin ajuri ilmestyy joskus
    - Tilanne on sama kuin vuonna 2010: se on edelleen mahdollista
- Nvidia
  - Suljettu, vakaa. Tukenut Linuxia jo yli 10 vuotta
  - 3D yhtä nopeaa kuin Windows-puolella
  - Tukee laitteistokiihdytettyä videopurkua
  - Ei speksejä, ei toivoa avoimesta ajurista
- Ubuntussa on hyvät työkalut suljettujen näytönohjainajurien asennukseen

Laitteistolistaukset

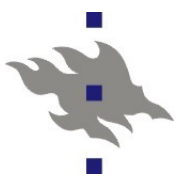
- */usr/bin/lshw* – toimiva yleistyökalu lähes kaiken laitteiston listaamiseen
  - BIOS, CPU, emo (muistit), PCI, USB, IDE, SATA, SCSI
- Mistä tuo kaikki tieto löytyy?
  - */usr/sbin/dmidecode*
    - BIOS:in mielipide koneen kokoonpanosta
  - */sbin/lspci*
    - Lista PCI-laitteista
  - */sbin/lsusb*
    - Lista USB-laitteista
  - Sysfs
    - */sys* hakemisto
    - Kernelin näkymä kaikkeen kernelin tuntemaan laitteistoon
  - Udev: */sbin/udevadm info --exportdb*
    - Udev daemonin laitteistotietokanta
    - Myös tarjoaa myös rajapinnan käyttäjätason laitteistohallintaan

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

PC:n käynnistyminen

PC:n käynnistyminen

- Aluksi kontrollin saa emolevyn firmware, BIOS
  - BIOS alustaa CPU:n, muistit ja emolevyn tuntemat levyt
  - Jokin oheislaitteet voivat sisältää myös BIOS-koodia, joka suoritetaan emolevyn varsinaisen BIOS:in toimesta
    - Mahdollistaa laitteen levy- ja RAID-ohjaimilta käynnistämisen
    - Verkkokorttien PXE-ympäristöt, verkosta käynnistämistä varten
- BIOS lataa valitulta käynnistyslaitteelta käynnistyslataajan (bootloader)
  - UEFI BIOS lukee käynnistyslataajan erilliseltä partitiolta
  - Perinteinen BIOS ei tue varsinaisia tiedostojärjestelmiä
    - Käynnistyslataajan on siis sijaittava jossa tunnetussa kohtaa levyä
    - Levyn alku (MBR) tai partition ensimmäinen blokki
  - Käynnistyslataaja on käyttöjärjestelmän omainen tuote: se osaa käyttää joitain laitteita (tavallisesti BIOS:in kautta) ja kenties lukea tiedostojärjestelmiä
  - PXE-verkkokäynnistysympäristö tukee suoritettavien varsinaisten ohjelmien latausta verkosta



Perinteinen PC:n käynnistys kovalevyltä

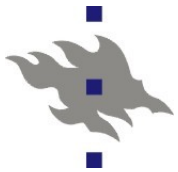
- MBR Master Boot Record sisältää partitiotaulun ja käynnistyslataajan
  - 512 oktettia levyn alussa
  - MBR:llä on listattu tasan 4 partiota, joista jotkut voivat olla käyttämättömiä
  - MBR:n käynnistyslataajan on ladattava 2. vaiheen käynnistyslataaja
    - Jonkin levypartition alusta
    - Tai jostain ennalta tunnetulta sijainnista levyltä
- Geometriaongelmat
  - Alunperin kovalevyjä osoitettiin kiinteällä joukolla bittejä, jotka valitsivat raidan, sektorin ja levyn
  - Bitit loppuivat kesken...
  - Nykyään levyjen blokkeja osoitetaan blokin järjestysnumerolla
  - Geometriaongelmat ovat historiaa

BIOS käynnistyslataajan rajoituksia

- PC:n käynnistämisessä on paljon historiallista painolastia
  - Kovalevypartitiolle asennetun käynnistyslataajan max koko on 1K (vähemmän, jos käynnistyslataaja on asennettu partitiotaulun MBR:lle)
  - Käynnistyslataaja suoritetaan 16-bit tilassa
  - Alle kilon kokoinen lataajaohjelma ei voi käytännössä sisältää laite- tai tiedostojärjestelmäajureita
    - Käynnistyslataaja käyttää BIOS:in palveluita levynlukuun
  - Käynnistyslataajaan onkin ladattava kerneli- ja initrd tai **2. vaiheen** käynnistyslataaja jostain tunnetusta kohtaa levyä
  - **Tämän vuoksi grub-käynnistyslataajan tiedostoja /boot/grub/ hakemiston alla ei voi siirtää ilman grubin uudelleenasennusta**
- Windows-asennus (ja usein myös päivitys) jyrää muut käynnistyslataajat MS:n omalla
- UEFI BIOS korjaa nämä ongelmat
  - … mutta todennäköisesti tuo mukanaan koko joukon uusia

Käynnistyslataaja ja Linux

- Käynnistyslataajan on vähintään ladattava muistiin Linuxin kerneli
  - Mikäli kerneliin on käännetty kaikki linuxin juuritiedostojärjestelmään liittämiseen tarvitut laiteajurit, käynnistyslataaja voi tämän jälkeen antaa suoraan kontrollin kernelille
  - Käynnistyslataaja antaa kernelille kernelin komentoriviparametrit (*/proc/cmdline*)
  - Parametreilla voidaan säätää myös käyttäjätason toimintaa
- Käynnistyslataaja lataa aina muistiin myös initrd:n (initial ramdisk)
- Käynnistyslataaja toteuttaa käynnistysmenun, jolla valitaan mikä mahdollisesti useasta käyttöjärjestelmästä ja kernelistä käynnistetään



GRUB 2

- Löytää käynnistyslevyt levyn metatietojen perusteella
  - Jos BIOS vain näkee levyn ja tiedostojärjestelmän grub löytää sen
- Distribuutioiden oletuskäynnistyslataaja
- Ominaisuuksia
  - Sisäänrakennettu aito skriptikieli (kontrollirakenteet)
  - Graafinen käyttöliittymä
  - Dynaaminen grub modulien lataus ajoaikana
  - Tuki eri merkistöille ja kielille
  - Kattava tuki eri tiedostojärjestelmille
  - Myös LVM ja softaRAID lohkolaitteille
- Distribuution pakettien asennusskriptit generoivat Grubin konfiguraatiotiedostot
  - Ubuntussa *update-grub*
- Tuki UEFI käynnistykselle

Käynnistäminen verkosta

- Verkkokortin firmwarelta ladataan PXE-ympäristö
  - Preboot Execution Environment
  - PXE hakee verkosta osoitteen DHCP:lla
  - DHCP-serveri kertoo IP-osoitteen lisäksi, mistä TFTP:llä haetaan käynnistyslataaja
- PXElinux
  - Käynnistyslataaja, joka toimii pxe-ympäristön alla
  - Hakee edelleen TFTP:llä konfiguraatiotiedoston
  - Konfiguraatiotiedoston ja konsolikäyttäjän tekemien valintojen pohjalta lataa (edelleen TFTP:llä) kernelin ja initrd:n ja käynnistää kernelin
  - Pxelinuxin kautta käynnistetään tavallisesti Linuxin verkkoasennus (ei tarvita käynnistysmediaa)
  - Mahdollista virittää myös levyttömiä työasemia
    - Eihän niitä kukaan enää kaipaa

Initrd

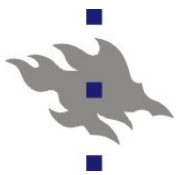
- Initrd on käynnistyslataajan levyltä muistiin lukema RAM-tiedostojärjestelmä
- Käynnistyksen alussa kernel näkee aluksi vain tämän tiedostojärjestelmän
- Initrd:ltä käynnistetään ensimmäiset järjestelmän alustavat käynnistysskriptit
  - Initrd pitää sisällään tarvittavat ajurimodulit ja ohjelmistot, jotta juuritiedostojärjestelmä saadaan liitettyä
  - Initrd:n ansiosta levyjärjestelmäajureiden ei tarvitse olla käännettynä kiinteästi kerneliin
  - Mahdollistaa kryptattujen ja softaRAID-levyjen käytön Linuxin juuritiedostojärjestelminä ja swap partitioina
  - Initrd:ltä voidaan myös palauttaa hibernoitu Linux
- Initrd:n sisältö riippuu käytetystä kernelistä ja järjestelmän konfiguraatiosta
  - Initrd generoidaan automaattisesti
  - Ubuntussa *update-initramfs* skriptillä

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Linuxin käynnistyminen

Käynnistyminen: Kernel ja Initrd

- Alussa kernel käynnistyy...
  - Alustetaan prosessorit ja muisti
  - Potkaistaan prosessit käyntiin
  - Potkaistaan kernelin sisään käännetyt ajurit käyntiin
- Käynnistyslataaja on ladannut muistiin myös initrd:n
  - Puretaan initrd:n tiedostojärjestelmä muistiin
  - Käynnistetään initrd
  - Initrd lataa käynnistämisessä tarvittavat ajurit
  - Liittää juuritiedostojärjestelmän
  - Ja lopettaa suorituksensa
- Tässä vaiheessa juuritiedostojärjestelmä on olemassa ja käytettävissä
  - Suoritetaan */sbin/init*
  - Initrd:tä ei enää tarvita ja sen käyttämä muisti voidaan vapauttaa

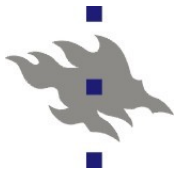


Käynnistyminen: perinteinen SysVinit

- */sbin/init* on Unixien perintökalu suoraan 70-luvulta
- Init on kaikkien käynnissä olevien prosessien esi-isä
  - Init adoptoi orvoiksi jääneet lapsiprossit
  - Init:in PID on 1
- Käynnistiessään init lukee tiedoston */etc/inittab*, missä valitaan systeemin runlevel. Perinteisestti
  - Runlevel 0: HALT
  - Runlevel 1: Single User / Repair Mode
  - Runlevel 6: Restart
- Komennolla *telinit* vaihdetaan runlevel
- Lisäksi Linuxeissa
  - Runlevel 3: Ei käynnistetä graafista ympäristöä (X)
  - Runlevel 5: Käynnistetään graafinen ympäristö (*DM)

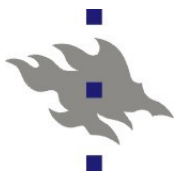
Perinteinen SysVinit

- Aluksi init käynnistää ensimmäisen järjestelmän alustusskriptin
  - Fedorassa */etc/rc.d/rc.sysinit*
- Sitten runlevelin mukaan */etc/rc.d/rc* -skriptin käynnistämään palveluita runlevelin mukaan
- Konsolin sisäänkirjautumis prosessien (*getty*) käynnistäminen on init:in vastuulla
- ctrl+alt+del tervehdyksen käsittely
- Käytössä edelleen monissa Linux distribuutioissa, mutta ei Fedorassa eikä Ubuntussa



Daemonit

- Käynnissä olevia erilaisia palveluja tarjoavia systeemiprosesseja kutsutaan Unixeissa ja Linuxeissa daemoneiksi
- *Init*-prosessin konfiguraatiolla valitaan, mitä daemoneja käynnistetään
- SysVinit:in */etc/init.d* -hakemistossa sijaitsee eri daemonien käynnistys- ja pysäytys skriptit
- Näistä tehdään symlinkkejä runlevel-kohtaisiin hakemistoihin, joilla konfiguroidaan mitkä palvelut käynnistetään ja missä järjestyksessä
- RH/Fedora pohjaisissa komentorivityökalut:
  - */sbin/chkconfig:* valitaan käynnistettävät daemonit
  - */sbin/service:* käynnistetään ja pysäytetään daemoneja



Ubuntu: Upstart

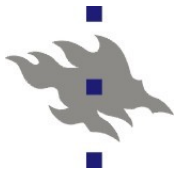
- Korvike vanhalle SysV init-ohjelmalle
  - Ja myös *init.d* -skripteille
- Käytössä Ubuntussa
- Palveluiden konfiguraatiohakemisto */etc/init*
  - Ohjelmistopaketista asennetut daemonit ja palvelut käynnistävät **.conf*-tiedostot pudotetaan tähän hakemistoon
- Reagoi abstrakteihin tapahtumiin, ei muutamaan ennaltavalittuun runleveliin
  - Esimerkiksi tapahtumat
    - *Filesystems* – tiedostojärjestelmät liitetty
    - *Network* – verkko alustettu ja toimii
- Toteuttaa *dbus*-rajapinnan palveluiden hallintaan
- Nopeuttaa järjestelmän käynnistystä
  - Teoriassa mahdollistaa shelliskriptien eliminoinnin
  - Käynnistää riippumattomat palvelut rinnakkain
- Palveluiden hallintaan kommenot */sbin/start, /sbin/stop, /sbin/reload, /sbin/stop ja /sbin/status*

Fedora: systemd

- Fedora perinteinen init on korvattu *systemd* järjestelmällä
- Samat tavoitteet kuin *upstart* daemonilla
  - Ymmärtää palveluiden väliset riippuvuudet
- Käyttää Linuxin omia rajapintoja prosessien hallintaan
  - Käyttää Linuxin cgroup-prosessiryhmiä
- Konfiguraatiohakemisto */etc/systemd*
- Komentorivikäytössä */sbin/systemctl* komento
- Toteuttaa myös palveluiden konfiguraation hallinnan
  - */sbin/systemctl* – listaa asennetut palvelut, niiden tilan ja kuvauksen
  - */sbin/systemctl start palvelu* - käynnistä palvelu
  - */sbin/systemctl stop palvelu* - pysäytä palvelu
  - */sbin/systemctl enable* - salli palvelun käynnistyminen
  - */sbin/systemctl disable* – estä palvelun käynnistyminen
- Fedorassa järjestelmän lokin tallentaa *systemd*!
  - Komento */usr/bin/journalctl* lokitiedon kyselyyn

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Järjestelmädaemonit



Udev

- Udev-daemon kehitettiin luomaan ja nimeämään laitetiedostot dynaamisesti
  - Laitteilla voi olla useita vaihtoehtoisia nimiä
    - Perinteiset nimet (*/dev/sda1, /dev/cdrom)*
    - Pyrkii pitämään perinteiset nimet samoina eri käynnistyskerroilla
    - Luo laitteella monia vaihtoehtoisia nimiä (symlinkeillä)
      - Laitteen sarjanumero tai laitteen väyläpaikka
      - Tiedostojärjestelmän nimi tai yksikäsitteinen tunniste (UUID)
  - Laitetiedosto luodaan laitteen tullessa järjestelmään ja poistetaan laitteen poistuessa
  - */dev* -hakemisto siirrettiin tmpfs-tiedostojärjestelmälle
    - Käsin tehdyt muutokset häviävät konetta käynnistettäessä
    - Käsin ei enää pitäisi tehdä muutoksia
  - Udev pitää käynnistää hyvin aikaisessa vaiheessa
    - Ilman */dev* -hakemistoa mikään ei toimi
    - Käynnistetään jo initrd:llä

Udev: toiminta

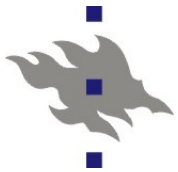
- Udev saa kerneliltä tiedon uudesta laitteesta
  - Udev lataa tarvittavan ajurin yhdistämällä laitteen tunnistetiedot kernelin modulien metadataan
    - Kernel ei osaa ladata moduleita itse
    - Joillekin laitteille on vaihtoehtoisia ajureita: ajurivalinta tehdään käyttäjätasolla (*modprobe.conf* musta lista)
  - Ajurimodulin lataus voi edelleen löytää lisää laitteita
  - Udev luo laitetiedostot ja suorittaa laitteeseen liittyvät skriptit
- Sääntöpohjainen konfigurointi
  - Tapahtuman metatiedoilla valitaan udev säännöt, jotka kertovat miten tapahtumaan tulee reagoida
    - Sääntöjä on mm. PCI-laitteille, USB-laitteille, verkkoliittymille ja tiedostojärjestelmille
    - Esim. Järjestelmään liitetyistä levyistä selvitetään levyn partitioiden tiedostojärjestelmien tyypit ja metadata (Label, UUID)
- Monitorointi: /sbin/*udevadm –monitor*
- Koko tietokannan listaus: */sbin/udevadm --export-db*

Laitteiston firmwaren lataus

- Monet laitteet tarvitsevat erillisen toimiakseen oman firmwaren
  - Perinteisesti laitteen firmware on talletettu laitteen sisäiselle flash-muistille
  - Monet laitteet nykyään olettavat, että on ajurin tehtävä ladata laitteelle firmware
- */lib/firmware* -hakemistoon on talletettu kernelin modulien tarvitsemat kernelin ulkopuoliset firmware-tiedostot
- Osa firmware tiedostoista tulee kernelin mukana
- Kernelin ajurimoduli pyytää käyttäjätasoa lataamaan firmware tiedoston kernelin muistiin
  - Ajurimoduli tietää haluamansa firmware-tiedoston nimen
- Käyttäjätasolla firmwaren lataus on udev:in tehtävä
  - Udev yksinkertaisella skriptillä kopioi laiteajurin haluaman tiedoston kernelin muistiin sysfs:n kautta.

Udev: esimerkki

1. PCI-väylältä löytyy USB-ohjain
   - Udev lataa ajurin ohjaimelle
2. USB-ohjaimen takaa löytyy USB-hubin
   - Udev lataa ajurin USB-hubille
3. USB-hubin takaa löytyy USB-muistitikku
   1. Udev lataa ajurin USB-massamuisteille
      1. Moduliriippuvuudet lataavat SCSI-levyajurin ja SCSI-alijärjestelmän
4. Kernel löytää tikulta partitiotaulun
   - Udev tarkistaa jokaisen partition tiedostojärjestelmän tyypin ja metadatan ja luo jokaiselle partitiolle laitetiedostot ja aliakset
5. Hubin takaa löytyy USB WLAN-tikku
   - Udev lataa WLAN-tikulle ajurin
   - Ajuri tarvitsee toimiakseen firmwaren tiedoston X
   - Udev lataa firmware-tiedoston kernelin muistiin

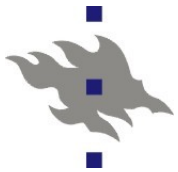


Syslog

- Syslog-daemoni ja sen C-kirjastorajapinta on unixien ja linuxien keskitetty tapa kerätä järjestelmälokia
  - Kernelin loki välitetään syslog:ille *klogd* -daemonin kautta
    - Kun syslog käynnistyy, niin ensimmäiseksi kaikki Linuxin käynnistyksessä kertynyt kernelin loki kerätään talteen
- Lokitapahtumalla on ”prioriteetti” ja aikaleima
  - Emergency, Alert, Critical, Error, Warning, Notice, Info, Debug
    - Eri ohjelmistot käyttävät näitä vaihtelevasti
  - Lokitapahtumaan liittyy myös ”alijärjestelmä”
    - Alijärjestelmien joukko lyötiin lukkoon joskus 70-luvulla...
    - Valitettavasti tämä on prioriteetin rinnalla ainoa tapa lajitella lokitapahtumia eri lokeroihin
    - Osa näistä on silti edelleen hyödyllisiä: LOG_AUTHPRIV, LOG_MAIL, LOG_KERNEL
  - Käytännössä softat lisäävät oman nimensä lokirivin alkuun

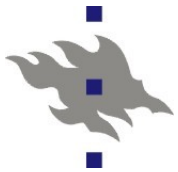
Syslog: toiminta

- Konfiguraatiotiedosto */etc/syslog.conf*
- Tyypillisesti lokia kirjoitetaan hakemistoon */var/log*
- Vanhat lokitiedostot pitää siivota pois!
  - cron:ista käynnistetään ajastetusti *logrotate*
- Syslog osaa lähettää lokia udp viesteillä toiselle verkossa olevalle syslog-daemonille
  - Lokit voi ja kannattaa kerätä keskitetysti
  - Usein kaatumistilanteissa tiedostojärjestelmä katoaa alta, eivätkä viimeisimmät (ja usein oleellisimmat) lokirivit jää talteen
  - Tietoturvan kannalta oleellista: estää lokien editoinnin jälkikäteen
    - Tai ainakin tekee siitä vaikeampaa
  - Käyttää yksinkertaisia UDP-viestejä: ei luotettavaa, ei autentikointia, ei kryptausta
    - Rsyslogd ja syslog-ng korjaavat tämän



Rsyslogd

- Käytössä Ubuntussa
- Modernimpi toteutus syslogd:sta ja syslog-protokollasta
- Toteuttaa syslog-viestien luotettavan välityksen
  - Käyttää TCP-pistokkeita
  - Osaa TLS-kryptauksen ja sertifikaattien tarkastuksen
  - Puskuroi viestit joita ei voitu toimittaa perille syslog-serverin ollessa poissa käytöstä
- Syslog-viestien suodatus
  - Säännöllisillä lausekkeilla
- Ymmärtää vanhan *syslog.conf* -tiedoston syntaksin
- Laajennettavissa moduleilla
  - Lokin voi tallettaa muuallekin kuin tekstitiedostoon



D-Bus

- Rajapinta ja daemoni prosessien väliseen viestinvälitykseen
  - Yksi järjestelmän daemoni (system bus)
  - Yksi daemoni jokaisella käyttäjäistunnolla (session bus)
  - C- ja python-kirjastot
  - Mekanismi rajapintojen tyyppiturvalliseen määrittelyyn ja listaukseen
  - Palvelua pyytävän prosessin identiteetin ja oikeuksien varmistus
  - Käytetään työpöytäohjelmistojen väliseen viestinvälitykseen
    - Esim. Työpöydän lukitseminen
  - Käyttäjärajat ylittävään viestinvälitykseen
    - Desktop-ohjelmistot voivat pyytää ylläpitäjän oikeuksin toimivilta daemoneilta palveluita
    - Laitteiston ja virransäästön hallintaan (udev), verkonhallinta (network manager) , Bluetooth-laitteet (bluetoothd) …
  - Komentoriviltä *dbus-monitor* ja *dbus-send*
  - GUI navigointiohjelmisto *d-feet*