

## Osa III: Failid, võrk ja süsteemitöö — täisspikker

Tase: **Praktiline kasutus**

### Failide kopeerimine ja sünkroonimine

**Eesmärk:** vali tööriist allika järgi: cp samas masinas, scp üle SSH, rsync korduvaks sünkroonimiseks ja curl/wget veebist toomiseks

#### Põhikujud

- cp fail.txt koopia.txt — tee koopia
- cp -R kaust kaust-koopia — kopeeri kataloog
- scp fail.txt kasutaja@server:/tmp/ — saada serverisse
- rsync -avn kaust/ kasutaja@server:/tmp/kaust/ — kontrolli enne
- rsync -av kaust/ kasutaja@server:/tmp/kaust/ — sünkrooni sisu
- wget https://artifical.eu/linux/data/generated-words.txt — laadi fail
- curl -fLO https://artifical.eu/linux/data/generated-words.txt — järgi suunamisi, HTTP-vea korral ebaõnnestu

#### Olulisemad lipud, märgid ja kiirnupud

- cp -R — kopeeri kataloog
- scp -r — saada kataloog
- rsync -a — rekursiivselt, säilitades tavalised Unixi metaandmed
- rsync -n — proovi enne
- curl -L — järgi suunamist
- curl -O — salvesta URL-i lõpuosa nimega

**Pane tähele:** Vali tööriist allika järgi: cp samas masinas, scp üle SSH, rsync korduvaks sünkroonimiseks, curl/wget veebist toomiseks.

### Kauglogimine ja SSH

**Eesmärk:** tee SSH korduvkasutatavaks: kontrolli olemasolevaid võtmeid, lisa avalik võti serverisse, kasuta failis ~/.ssh/config määratud aliaid ja macOS-is Apple'i võtmehoidjat

#### Põhirada

- ssh kasutaja@server.example.org — proovi tavalist sisselogimist
- ls -la ~/.ssh — vaata olemasolevaid võtmefaile
- ssh-add -l — vaata agendis olevaid võtmeid
- ssh-keygen -t ed25519 -C 'kasutaja@arvuti' — loo võti ainult vajadusel
- /usr/bin/ssh-add --apple-use-keychain ~/.ssh/id\_ed25519 — macOS: salvesta võtmefraas võtmehoidjasse

- `ssh-copy-id -i ~/.ssh/id_ed25519.pub kasutaja@server.example.org`  
— lisa avalik võti serverisse
- `ssh opik-server` — kasuta seadistusfaili aliast
- `ssh -v opik-server` — diagnostika

### Olulised failid ja mõisted

- `id_ed25519` — privaatvõti; ära jaga
- `id_ed25519.pub` — avalik võti serverisse
- `~/.ssh/config` — ühenduse alias
- `~/.ssh/authorized_keys` — lubatud võtmed
- `known_hosts` — serverite sõrmejäljed
- `AddKeysToAgent` — lisa võtmed agentile
- `UseKeychain` — võtmefraas macOS-i võtmehoidjas
- `IdentitiesOnly yes` — kasuta nimetatud võtit

**Pane tähele:** Alusta sisselogimisest ja võtmetest; port forwarding nagu `ssh -L` ... on järgmine, mitte esimene samm.

### Veebist sisu toomine ja tekstivaade: `curl`, `wget`, `lynx`

**Eesmärk:** vali `curl` ühe vastuse või päiste jaoks, `wget` allalaadimise jaoks ja `lynx` HTML-i rahulikuks tekstivaateks

### Põhikujud

- `curl -I https://example.com/` — kontrolli URL-i
- `curl -L -o leht.html https://example.com/` — salvesta HTML
- `wget -O leht.html https://example.com/` — laadi faili
- `curl -fsSL https://example.com/ | lynx -stdin -dump` — loe tekstina
- `lynx -dump -listonly https://example.com/` — kogu lingid

### Olulisemad lipud, märgid ja kiirnupud

- `curl -I` — vaata päiseid
- `curl -L` — järgi suunamist
- `curl -O` — salvesta URL-i lõpuosa nimega
- `wget -c` — jätkka tõmmet
- `lynx -dump` — tekstivaade
- `lynx -listonly` — ainult lingid

**Pane tähele:** Alusta alati ühest URL-ist; ulatuslik veebis roomamine ja automaatne veebikraapimine olgu järgmine, mitte esimene samm.

## Arhiivid ja pakkimine

**Eesmärk:** arhiiv koondab failid; tihendamine võib mahtu vähendada, kuid juba pakitud sisu puhul mitte tingimata

### Põhikujud

- `tar -cf proov.tar kaust/` — loo arhiiv
- `tar -tf proov.tar` — vaata sisu
- `tar -czf proov.tar.gz kaust/` — loo gzip-arhiiv
- `tar -xzf proov.tar.gz -C siht/` — paki lahti sihtkohta
- `zip -r proov.zip kaust/` — loo zip
- `unzip -l proov.zip` — vaata zip sisu

### Olulisemad lipud, märgid ja kiirnupud

- `tar -cf` — loo arhiiv
- `tar -tf` — vaata sisu
- `tar -czf` — gzip-iga kokku
- `tar -xzf` — paki gzip lahti
- `zip -r` — paki kataloog ZIP-arhiivi
- `unzip -l` — vaata zip sisu

**Pane tähele:** Enne lahtipakkimist vaata võimalusel arhiivi sisu; see aitab näha, kas sees on oodatud juurkataloog ja failipuud.

## Tervete kataloogipuude haldus ja jagamine

**Eesmärk:** terve projekti või kataloogipuu puhul vali tööriist selle järgi, kas tahad koopiat, korduvat sünkroonimist, arhiivi või versiooniajalugu

### Põhikujud

- `cp -R projekt projekt-koopia` — tee kohalik koopia
- `rsync -avn projekt/ projekt-varu/` — kontrolli enne
- `rsync -av projekt/ kasutaja@server:/srv/projekt/` — sünkrooni serverisse
- `scp -r projekt kasutaja@server:/tmp/` — saada kiirelt
- `tar -czf projekt.tgz projekt/` — tee hetkearhiiv
- `git` — jälgi ajalugu

### Olulisemad lipud, märgid ja kiirnupud

- `projekt/` — rsync-allikas: kataloogi sisu
- `projekt` — rsync-allikas: kataloog ise
- `rsync -n` — proovi enne
- `cp -a` — säilita metaandmed

**Pane tähele:** Kui vajad korduvat sünkroonimist, eelista tavaliselt rsync-i; scp -r sobib pigem üheks kiireks ülekandeks.

## Protsessid, tööd ja signaalid

**Eesmärk:** erista protsessi, shelli tööd ja signaali: vaata, millised protsessid töötavad, peata vajadusel ning jätka pikka tööd taustal

## Põhikujud

- `ps aux | grep python` — otsi protsess
- `ps aux | sort -nrk 3 | head` — suur CPU kasutus selles hetkepildis
- `ps aux | sort -nrk 4 | head` — suur mälu kasutus
- `sleep 300 &` — käivita taustal
- `jobs` — vaata töid
- `fg %1` — too töö ette
- `top` — jälgi reaalajas
- `kill` — saada signaal
- `bg` — jätkata taustal

## Olulisemad lipud, märgid ja kiirnopud

- `Ctrl-c` — katkesta programmi töö
- `Ctrl-z` — peata ajutiselt; `fg` ette, `bg` taustale
- `&` — käivita taustal
- `%1` — töö number
- `kill -15` — viisakas lõpp
- `kill -9` — jõuga lõpp

**Pane tähele:** Kasuta `kill -9` alles siis, kui tavalisem `kill` või `kill -15` ei lõpeta protsessi viisakalt.

## Logid ja teenused

**Eesmärk:** Kontrolli teenuse olekut ja seejärel sama teenuse logi. `systemctl/journalctl` vajavad `systemd`-ga Linuxit.

## Käsud ja näited

- `systemctl status ssh` — teenuse seis ja hiljutised teated; nimi võib olla `sshd`
- `journalctl -u ssh -n 50` — selle teenuse viimased 50 kirjet
- `journalctl -u ssh --since today` — tänased teenusekirjed
- `journalctl -u ssh -f` — jälgi juurde tulevaid kirjeid
- `tail -n 20 app.log` — tavalise logifaili lõpp, ka ilma `systemd`-ta
- `tail -f app.log` — jälgi faili; `Ctrl-c` lõpetab jälgimise

**Pane tähele:** Logi puudumine ei tõesta vigade puudumist: kontrolli teenuse nime, logimise seadistust ja lugemisõigust.

### Püsivad terminalisessioonid: **tmux** ja **screen**

**Eesmärk:** Säilita interaktiivne serverisessioon ka SSH-ühenduse katkemisel.

#### Käsud ja näited

- `tmux new -s katse` — loo nimega sessioon
- `tmux ls` — loetle sessioonid välisest shellist
- `tmux attach -t katse` — ühendu olemasoleva sessiooniga
- `screen -S katse` — alternatiiv: loo screeni sessioon
- `screen -r katse` — taasühendu eraldatud screeniga

#### Märgid ja valikud

- `Ctrl-b`, siis `d` — `tmux`: eraldu, töö jätkub
- `Ctrl-a`, siis `d` — `screen`: eraldu, töö jätkub
- `exit` — lõpeta praeguse akna shell

**Pane tähele:** Sessioon ei ela üle serveri taaskäivitust ega volukatkestust.

### Graafilised rakendused kaugmasinast

**Eesmärk:** kaugelt graafika kasutamisel eelista lihtsaimat toimivat teed: veebiliides enne X11-edastust, port forwarding enne toorest kaugtöölauda.

#### Põhikujud

- `ssh -N -L 127.0.0.1:8888:localhost:8888 kasutaja@server` — tunnel juba töötava serveriteenuseni
- `http://localhost:8888` — ava edasi suunatud teenus
- `ssh -X kasutaja@server` — proovi lihtsat X11
- `code --remote ssh-remote+server /tee/projektini` — IDE üle SSH

#### Valiku rusikareeglid

- veebiliides — tavaliselt mugavam
- `ssh -L` — port edasi
- `ssh -X` — X11 erijuht
- Remote SSH — arenduseks parem

**Pane tähele:** Kui sul on valida, eelista brauserit või Remote SSH-d; X11 forwarding olgu pigem varuplaan, mitte esimene mõte.