



IT KOLLEDŽ
TALLINNA TEHNIKAÜLIKOOL

Kasutajakeskkond ja protsessid

Operatsioonisüsteemid ja nende haldamine ICA0001

Edmund Laugasson

edmund.laugasson@itcollege.ee

https://wiki.itcollege.ee/index.php/User:Edmund#eesti_keeles

Käesoleva dokumendi paljundamine, edasiandmine ja/või muutmine on sätestatud ühega järgnevatest litsentsidest kasutaja valikul:

* GNU Vaba Dokumentatsiooni Litsentsi versioon 1.2 või uuem

* Creative Commons Autorile viitamine + Jagamine samadel tingimustel 4.0 litsents (CC BY-SA)



Kasutajakeskkonna seaded

- Kasutaja süsteemi sisenemisel käivitatakse failid (skriptid), mis initsialiseerivad kasutajakeskkonna (luuakse kasutaja seanss)
 - Seadistatakse keskkonnamuutujad
 - Seadistatakse käskude aliased
 - näiteks `| = 'ls -l'`
 - Funktsioonid
- Initsialiseeritakse kõigile kasutajatele mõeldud seanss */etc/profile*
- Initsialiseeritakse kasutaja personaalsed seaded kasutaja kodukataloogis
 - *~/.profile*
 - *~/.bash_profile*
 - *~/.bashrc*
- Kasutaja saab personaalseid seadeid muuta

Bash shelli konfiguratsioonifailid

- **.bash_profile** kasutajate keskkonna individuaalne seadistamine. Võimaldab muuta vaikimisi sätteid ja lisada uusi. Käivitatakse kasutaja sisselogimisel.
- **.bash_login** käivitatakse ainult kasutaja sisselogimisel. Kui *.bash_profile* faili ei eksisteeri, loetakse *.bash_login* sisse.
- **.bashrc** käivitatakse näiteks terminali akna avamisel. (interaktiivne kestprogramm)
.bash_aliases – lühikäskude ehk **aliaste fail**
- **.bash_history** selles failis on kasutaja sisestatud käskude ajalugu. Vaata haldusprogrammi *history*. (Uuri mida teevad ! ja ^r)
- **.bash_logout** sisaldab käske, mida käivitatakse välja logimisel.
- **/etc/profile** sarnane *.bash_profile* failile, ainult et laieneb kõikidele kasutajatele.
- **/etc/profile.d** selles kataloogis olevad failid loetakse sisse sarnaselt **/etc/profile** failiga. Kui soovitakse teha erinevate funktsioonide jaoks eraldi failid, siis see on sobiv võimalus.
- Miks kasutatakse tihti *.d katalooge?
 - <http://unix.stackexchange.com/questions/4029/what-does-the-d-stand-for-in-directory-names>
 - https://wiki.itcollege.ee/index.php/BASH_shell



Seansiskriptide käivitamine

- *..skript*
- *source skript*
- *~/.bashrc* loetakse sisse iga kord, kui käivitatakse uus *shell*, näiteks avatakse uus terminaliaken
- */etc/profile* ja *~/.profile* käivitatakse kasutaja süsteemi sisenemisel
- Kui muudate näiteks *~/.profile* sisu, siis tuleb muudatuste rakendamiseks kas välja/sisse logida või käivitada *source .profile* (*..profile*)
- ka võrgusätted kaustas */etc/network/interfaces.d/* käsitletakse sessiooniskriptidena, sellekohane sissekanne on ka failis */etc/network/interfaces*:
 - *source /etc/network/interfaces.d/**

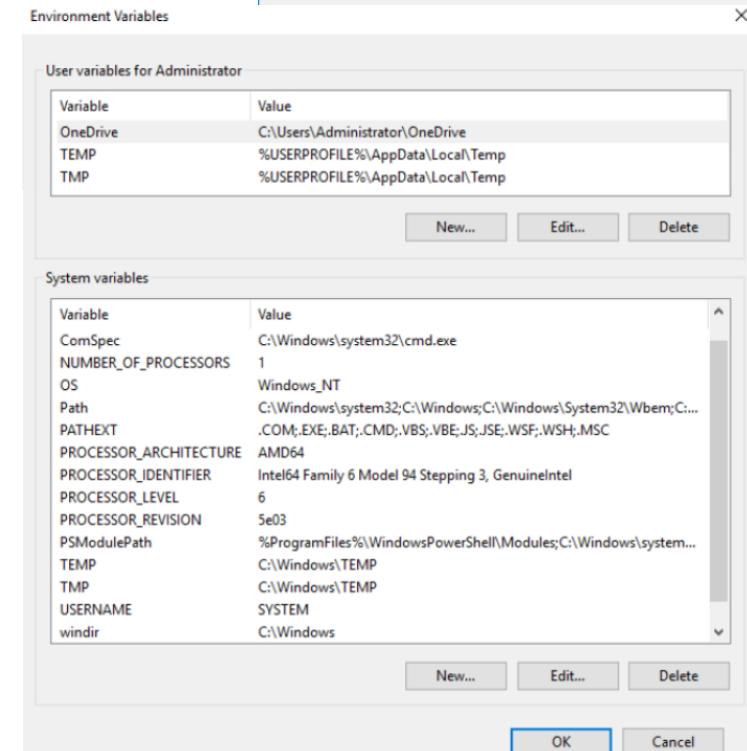
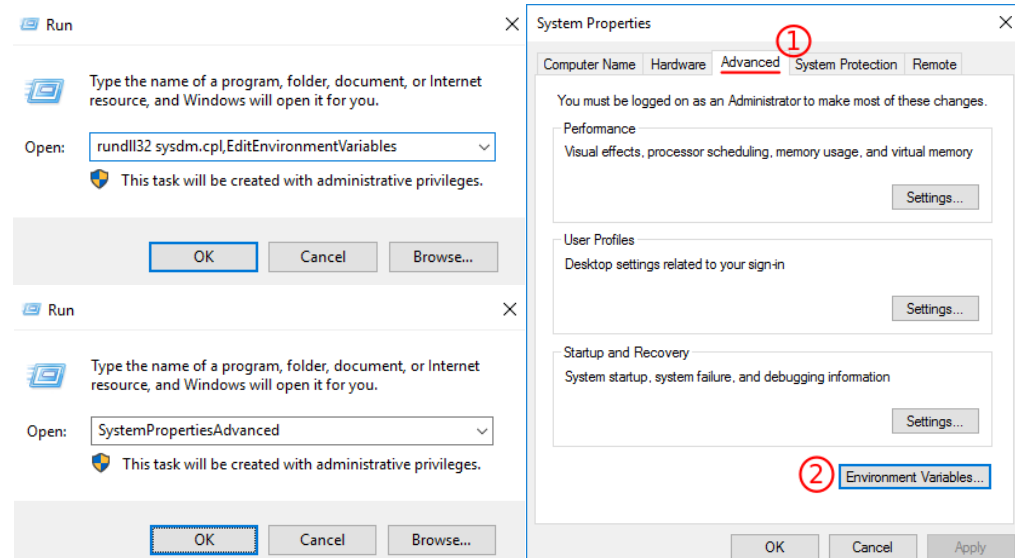


Lühikäsk (*alias*)

- Iga kasutaja saab enda jaoks defineerida pikkade käskudele lühivorme ehk *alias*'i
- *Bash*'i kestprogrammi lühikäsud *ls* käsule koos lisaparameetritega
 - `alias ll='ls -l'`
 - `alias la='ls -A'`
 - `alias lf='ls -CF'`
- https://wiki.itcollege.ee/index.php/Alias_bash_shellis
- https://wiki.itcollege.ee/index.php/.bash_aliases
- <https://www.digitalocean.com/community/tutorials/how-to-read-and-set-environmental-and-shell-variables-on-a-linux-vps>

Keskkonnamuutujad

- Linuxis pannakse käskudes keskkonnamuutuja ette \$
 - muutuja väärtuse vaatamine: `echo $HOME`
 - muutmine (üks neist)
 - globaalselt: `/etc/profile`, `/etc/environment`, `/etc/bash.bashrc`
 - kasutajapõhiselt: `~/.profile`, `~/.bashrc`, `~/.bash_profile` (kui `~/.bash_profile` ei eksisteeri siis loetakse `~/.bash_login` sisse); köikidele uutele kasutajatele panna vastav fail `/etc/skel/` kausta (nt: `/etc/skel/.profile`)
- MS Windowsis pannakse keskkonnamuutuja kahe protsendimärgi vahele
 - muutujate väärtuste vaatamine/muutmine:
 - Super+R
 - `rundll32 sysdm.cpl,EditEnvironmentVariables` #otsepäring
 - `SystemPropertiesAdvanced` → `Advanced` → `Environment Variables`
 - käsurreal muutuja vaatamine:
 - cmd
 - `set userprofile` #kodikataloogi asukoht
 - `echo %USERPROFILE%`
 - PowerShell (ketas ja kodukataloog)
 - `$env:homedrive;$env:homepath`
- Keskkonnamuutujate kuvamine käsurreal
 - Linuxis saab kasutada korraldust `env` või ka `printenv`
 - `set | less` #kestprogrammi - ja keskkonnamuutujad, kohalikud muutujad, kestprogrammi funktsioonid
 - `set -o posix; set | less` #kestprogrammi - ja keskkonnamuutujad,
 - MS Windows'is saab kasutada korraldust
 - cmd: `set` (ka `set | more`)
 - PowerShell: `Get-ChildItem Env:`
- Laiendatud keskkonnamuutujate nimekiri bash'is on nähtav `declare` käsuga





Keskkonnamuutujad 2

- Bash'i kestprogrammis seatakse
 - keskkonnamuutujad **export** korralduse abil (on kasutatavad ka teiste rakenduste poolt)
 - **export** *muutuja1=väärtus*
 - **export** *muutuja2=väärtus1:väärtus2*
 - **export** *muutuja3="Selle muutuja väärtus"*
 - kestprogrammi bash muutujad omistatakse võrduse teel (ei ole kasutatavad teiste rakenduste poolt):
 - *muutuja=väärtus*
 - *LC_ALL=C* käsk #vaid konkreetse käsu jaoks omistatakse USA inglise keel
- C kestprogrammis
 - **setenv** *muutuja väärtus*
- MS Windowsis
 - cmd: **set** *muutuja=väärtus*
 - PowerShell: **\$env:muutuja=väärtus**

Keskkonnamuutujad 3

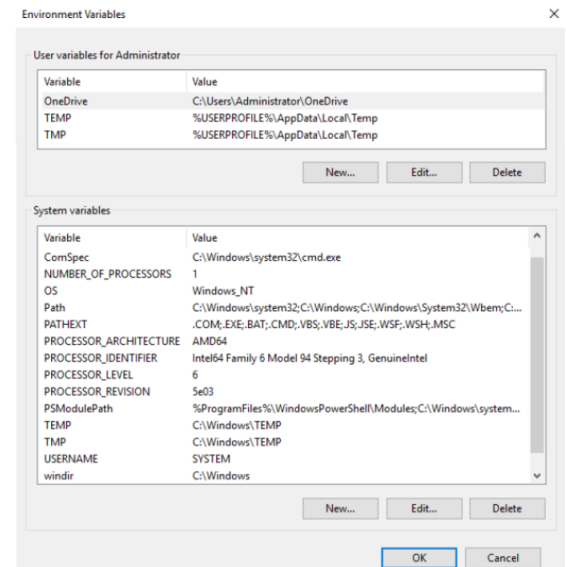
- Mõned levinud keskkonnamuutujad Linuxis:
 - USER – kasutajanimi
 - PATH – otsiteekond (kataloogide nimekiri, millest süsteem otsib programmifaile, mida kasutaja käivitab ilma kataloogile viitamata)
 - HOME – kasutaja kodukataloog
 - SHELL – kasutaja kestprogramm
 - EDITOR – kasutaja poolt eelistatud tekstiredaktor
 - HOSTNAME – masina nimi
- Mõned levinud keskkonnamuutujad MS Windows'is:
 - TEMP, TMP – ajutised failid
 - PATH – otsiteekond
 - USERNAME – kasutajanimi
 - WINDIR – süsteemi paigalduskataloog
 - APPDATA – rakenduste sätete kaust
 - COMPUTERNAME – masina nimi
 - SYSTEMDRIVE – süsteemiketas
 - SYSTEMROOT – süsteemi juurkataloog
 - HOMEDRIVE – ketas kus asub kodukataloog
 - HOMEPATH – kodukataloog
 - PROGRAMDATA – kõikide kasutajate rakenduste andmed
 - PROGRAMFILES – rakenduste paigaldused

lisainfo MS Windowsi eriotstarbelistest kaustadest:

<https://msdn.microsoft.com/en-us/library/windows/desktop/bb762494.aspx>

<https://msdn.microsoft.com/en-us/library/windows/desktop/dd378457.aspx>

```
student@server:~$ env
SHELL=/bin/bash
TERM=linux
USER=student
LS_COLORS=rs=0:di=01:34:ln=01:36:mh=
1:mi=00:su=37:41:sg=30:43:ca=30:41:
=01:31:*.arj=01:31:*.taz=01:31:*.lha
;31:*.tzo=01:31:*.t7z=01:31:*.zip=01
1:31:*.lzo=01:31:*.xz=01:31:*.bz2=01
.rpm=01:31:*.jar=01:31:*.war=01:31:
=01:31:*.cpio=01:31:*.7z=01:31:*.rz=
35:*.pbm=01:35:*.pgm=01:35:*.ppm=01:
*.png=01:35:*.svg=01:35:*.svgz=01:35
m2u=01:35:*.mkv=01:35:*.webm=01:35:
=01:35:*.nuv=01:35:*.wmv=01:35:*.asf
35:*.flv=01:35:*.gl=01:35:*.dl=01:35
gv=01:35:*.ogx=01:35:*.aac=00:36:*.a
00:36:*.mp3=00:36:*.mpc=00:36:*.ogg=
6:*.xspf=00:36:
MAIL=/var/mail/student
PATH=/home/student/bin:/home/student
ames:/snap/bin
PWD=/home/student
LANG=en_US.UTF-8
SHLVL=1
HOME=/home/student
LOGNAME=student
LESSOPEN=| /usr/bin/lesspipe %s
LESSCLOSE=/usr/bin/lesspipe %s %s
_/usr/bin/env
```





Protsessid

- Protsessi loomine
- Sisend/väljund ja vead
- Suunamine
- Protsessidevaheline kommunikatsioon
- Signaalid
- Tööd



Protsessid

- Protsess on käivitatud programm, millele on eraldatud protsessori ja mälu (aadressiruum) ressursid
- Protsessil on tunnus PID (*process ID*)
- Protsess võib käivitada teisi protsesse
 - Protsess, mis käivitas teise protsessi nimetatakse vanemaprotsessiks *parent process*
- Protsessid moodustavad protsessipuu, mille tipuks on esimesena käivitatud protsess (Linuxilaadsetel **init**)
 - `sudo stat /proc/1/exe`
 - `systemd --version`

Protsessitabel

- Operatsioonisüsteem peab arvet protsesside ja ressursside kohta
- Andmeid protsesside kohta hoitakse protsessitabelis
- Protsessipuu saab kuvada korraldusega (Linux/Unix)

ps tree

- Protsessitabeli saab kuvada korraldusega (Linux/Unix)

ps -ef

- <https://wiki.itcollege.ee/index.php/Ps>
- man ps
- veel valikut (vaja paigaldada): htop, atop

```
CPU: 0.0% Tasks: 29, 16 thr: 1 running
Mem: 70.7%/406M Load average: 0.00 0.00 0.00
Swap: 0k/512k Uptime: 16:24:05

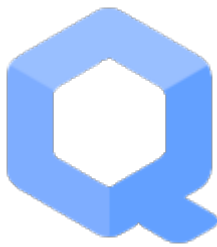
PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command
3523 root 20 0 27284 3816 3108 R 0.0 0.0 0:00.03 htop
1 root 20 0 37688 5788 4036 S 0.0 1.2 0:01.60 /sbin/init
431 root 20 0 27768 2692 2360 S 0.0 0.5 0:00.16 /lib/systemd/systemd-journald
456 root 20 0 1808 1528 1292 S 0.0 0.3 0:00.00 /sbin/umountd -f
460 root 20 0 44992 4264 3004 S 0.0 0.9 0:01.34 /lib/systemd/systemd-udevd
733 systemd-t 20 0 97M 2548 2336 S 0.0 0.5 0:00.00 /lib/systemd/systemd-timesyncd
724 systemd-t 20 0 97M 2548 2336 S 0.0 0.5 0:00.09 /lib/systemd/systemd-timesyncd
1004 root 20 0 270M 6200 5504 S 0.0 1.2 0:00.91 /usr/lib/accounts-service/accounts-daemon
1012 root 20 0 270M 6200 5504 S 0.0 1.2 0:00.00 /usr/lib/accounts-service/accounts-daemon
990 root 20 0 270M 6200 5504 S 0.0 1.2 0:00.93 /usr/lib/accounts-service/accounts-daemon
991 daemon 20 0 20044 2116 1920 S 0.0 0.4 0:00.00 /usr/sbin/atd -f
1007 syslog 20 0 250M 3340 2672 S 0.0 0.7 0:00.00 /usr/sbin/rsyslogd -n
1008 syslog 20 0 250M 3340 2672 S 0.0 0.7 0:00.00 /usr/sbin/rsyslogd -n
1009 syslog 20 0 250M 3340 2672 S 0.0 0.7 0:00.01 /usr/sbin/rsyslogd -n
992 syslog 20 0 250M 3340 2672 S 0.0 0.7 0:00.03 /usr/sbin/rsyslogd -n
996 root 20 0 28548 2976 2636 S 0.0 0.6 0:00.06 /lib/systemd/systemd-logind
997 messagebu 20 0 42984 3768 3392 S 0.0 0.8 0:00.04 /usr/bin/ibus-daemon --system --adre
1019 root 20 0 95368 1444 1312 S 0.0 0.3 0:00.00 /usr/bin/xcfs /var/lib/xcfs/
1020 root 20 0 95368 1444 1312 S 0.0 0.3 0:00.00 /usr/bin/xcfs /var/lib/xcfs/
1013 root 20 0 95368 1444 1312 S 0.0 0.3 0:00.00 /usr/bin/xcfs /var/lib/xcfs/
1014 root 20 0 30228 2880 2600 S 0.0 0.6 0:00.08 /usr/sbin/cron -f
1016 root 20 0 4400 1228 1132 S 0.0 0.2 0:00.24 /usr/sbin/acpid
1035 root 20 0 270M 21912 13608 S 0.0 4.4 0:00.06 /usr/lib/snapd/snapd
1036 root 20 0 270M 21912 13608 S 0.0 4.4 0:00.00 /usr/lib/snapd/snapd
1041 root 20 0 270M 21912 13608 S 0.0 4.4 0:00.03 /usr/lib/snapd/snapd
1044 root 20 0 270M 21912 13608 S 0.0 4.4 0:00.00 /usr/lib/snapd/snapd
1048 root 20 0 270M 21912 13608 S 0.0 4.4 0:00.00 /usr/lib/snapd/snapd
1049 root 20 0 270M 21912 13608 S 0.0 4.4 0:00.02 /usr/lib/snapd/snapd
1017 root 20 0 270M 21912 13608 S 0.0 4.4 0:00.15 /usr/lib/snapd/snapd
1029 root 20 0 270M 6012 5336 S 0.0 1.2 0:00.00 /usr/lib/policykit-1/polkitd --no-deb
```



Protsessid

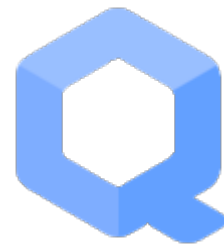
- Protsesside vahel ressursside jagamisega tegeleb operatsioonisüsteemi tuum (*kernel*)
- Protsess võib olla järgnevas olekus
 - Loodud (*created*)
 - Töötav (*running*)
 - Ootav (*waiting*)
 - ka välja saalitud ja ootel
 - Blokeeritud (*blocked*)
 - ka välja saalitud ja blokeeritud
 - Lõpetatud (*terminated*)
 - Vanemprotsessita protsess (*zombie*)

Qubes OS



- <https://www.qubes-os.org/> - eraldatud virtuaalarvutid, sh eraldi protsessid, lõikepuhver -> riskide minimeerimine
 - tutvustus <https://www.qubes-os.org/intro/>
 - alustamine <https://www.qubes-os.org/getting-started/>
 - dokumentatsioon <https://www.qubes-os.org/doc/>
 - testitud riistvara <https://www.qubes-os.org/hcl/>
 - tarkvara ja versioonid <https://www.qubes-os.org/doc/supported-versions/>
 - arhitektuur <https://www.qubes-os.org/doc/architecture/>
 - põhisüsteemi turvaline uuendamine <https://www.qubes-os.org/doc/dom0-secure-updates/>
 - rakendused töötavad eraldi virtuaalmasinates, Qubes OS ise võrgus ei ole (dom0)
 - ruuter jagab internetti ülejäänud virtuaalmasinatele
 - töö (*work*)
 - isiklik (*personal*)
 - usaldamatu (*untrusted*)
 - jne
 - <https://www.qubes-os.org/doc/templates/>
 - <https://www.qubes-os.org/doc/qubes-builder/>
 - võimalik teha ajutisi virtuaalmasinaid (Disposable VM, DispVM) <https://www.qubes-os.org/doc/dispvm/> - kustub kui suletakse
 - ühekasutaja süsteem koos mitme virtuaalmasinaga

Qubes OS



Use Fedora, Debian, or
even Windows



with  networking

SECURE COMPARTMENTALIZATION

Qubes brings to your personal computer the security of the Xen hypervisor, the same software relied on by many major hosting providers to isolate websites and services from each other. [Learn more](#)

OPERATING SYSTEM FREEDOM

Can't decide which Linux distribution you prefer? Still need that one Windows program for work? With Qubes, you're not limited to just one OS. [Learn more](#)

SERIOUS PRIVACY

With Whonix integrated into Qubes, using the Internet anonymously over the Tor network is safe and easy. [Learn more](#)



Protsessidevaheline side

- Protsessid saavad omavahel andmeid vahetada
 - Kasutades jagatud faile
 - Kasutades jagatud mälu
 - Kasutades pidemeid *socket*
 - Saates signaale
 - Kasutades semafore (lippe)
 - Kasutades toru *pipe*



Protsessid

- Protsessidel on (Linux laadsed)
 - Sisendvoog **STDIN** (0) <https://linuxjourney.com/lesson/stdin-standard-in-redirect>
 - Väljundvoog **STDOUT** (1) <https://linuxjourney.com/lesson/stdout-standard-out-redirect>
 - Veavoog **STDERR** (2) <https://linuxjourney.com/lesson/stderr-standard-error-redirect>
- Protsessi väljundit saab suunata teise protsessi sisendisse toru ehk | abil
 - **ps -ef | less**
 - Protsessi *ps* väljundvoog suunatakse protsessi *less* sisendisse
 - **ps -ef | grep bash | wc**
 - *wc* näitab vastavalt: ridade arv; sõnade arv; baitide arv
- väljund tasub suunata veebiteenusesse enne abi otsimist
 - <https://help.ubuntu.com/community/Pastebinit>



Faili suunamine

- Protsessi sisendi saab võtta failist suunajaga <
 - *cat < /dev/urandom*
 - Programmi cat sisendvoog võetakse juhuarvude genereerimise failist
- Protsessi väljundi saab kirjutada faili suunajaga > või >>
 - *cat < /dev/urandom > juhuarvud.dat*
 - Programm cat kirjutab oma väljundi faili juhuarvud.dat
 - Kusjuures juhuarvud.dat kirjutatakse üle
 - *cat < /dev/urandom >> juhuarvud.dat*
 - >> kirjutab faili lõppu



Veavoo suunamine

- Kui soovime, et programme ei kirjutaks väljundisse, siis suuname väljundi seadmesse `/dev/null`
 - `cat </dev/zero > /dev/null`
- Veavoog jääb suunamata ja selle suunamiseks kasutage `2>&1` lõppu
- `./ei_taha_tead_a > /dev/null 2>&1`
 - Veavoog suunatakse sinna, kuhu suunati väljundvoog (`/dev/null` e „must auk”)
- `./ei_taha_tead_a 2>/dev/null`
 - veavoog suunatakse otse “musta auku” (`/dev/null`)

Signaalid

- Protsessile saab saata signaale
- Protsess töötleb saadud signaalid
 - Protsess saab seadistada kindlad tegevused mitmetele signaalidele
 - Protsess võib mõningaid signaale ignoreerida
- Signaalidel on numbrilised märgendid ja ka lühinimed
- Signaali saatmine toimub **kill** korraldusega (vaikimisi SIGTERM)
- loetelu signaalidest: `kill -l`
- <https://wiki.itcollege.ee/index.php/Kill>

```
student@server:~$ kill -l
1) SIGHUP      2) SIGINT      3) SIGQUIT     4) SIGILL      5) SIGTRAP
6) SIGABRT    7) SIGBUS     8) SIGFPE     9) SIGKILL    10) SIGUSR1
11) SIGSEGV   12) SIGUSR2   13) SIGPIPE   14) SIGALRM    15) SIGTERM
16) SIGSTKFLT 17) SIGCHLD  18) SIGCONT   19) SIGSTOP   20) SIGTSTP
21) SIGTTIN   22) SIGTTOU  23) SIGURG    24) SIGXCPU   25) SIGXFSZ
26) SIGTALRM  27) SIGPROF  28) SIGWINCH  29) SIGIO     30) SIGPWR
31) SIGSYS    34) SIGRTMIN 35) SIGRTMIN+1 36) SIGRTMIN+2 37) SIGRTMIN+3
38) SIGRTMIN+4 39) SIGRTMIN+5 40) SIGRTMIN+6 41) SIGRTMIN+7 42) SIGRTMIN+8
43) SIGRTMIN+9 44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9 56) SIGRTMAX-8 57) SIGRTMAX-7
58) SIGRTMAX-6 59) SIGRTMAX-5 60) SIGRTMAX-4 61) SIGRTMAX-3 62) SIGRTMAX-2
63) SIGRTMAX-1 64) SIGRTMAX
```

Signaalid 2

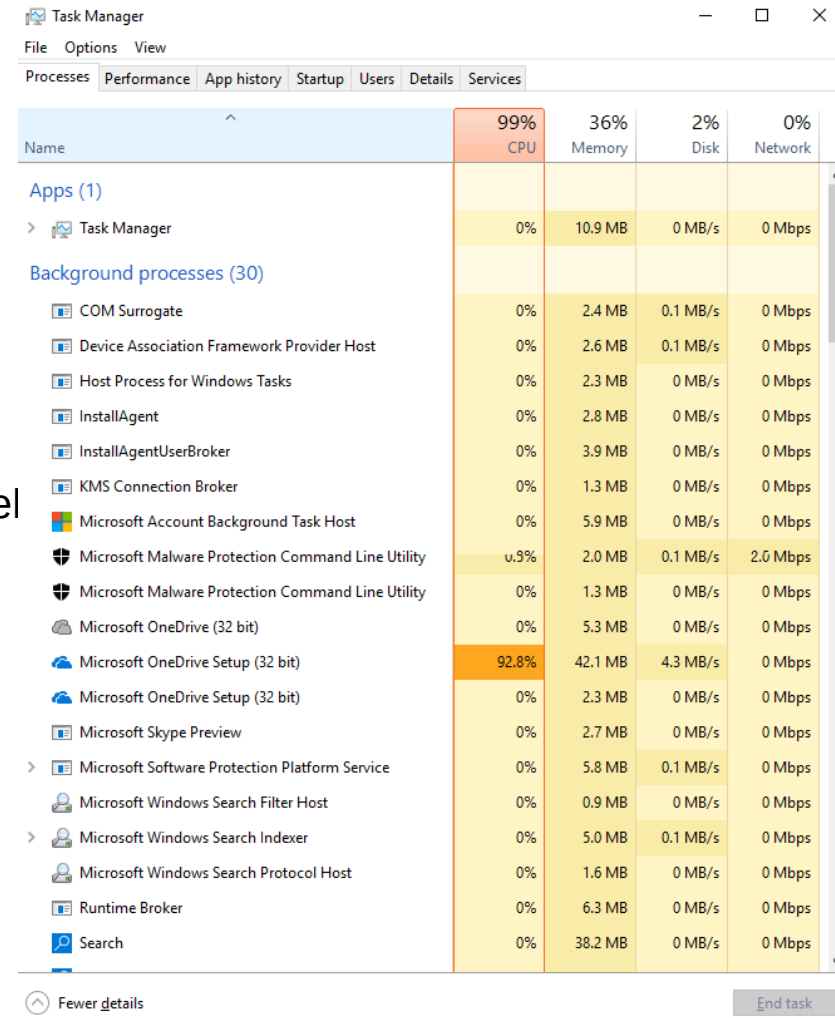
- Mõned signaalid (*kill*, *killall*)
 - SIGHUP 1 hangumine või protsessi suremine, kasutada konfiguratsiooni uuestilaadimiseks või logifailide sulgemiseks/avamiseks
 - SIGABRT6 Abort, tekitab core faili protsessi andmetest mälus
 - SIGKILL 9 Protsessilt jõuga ressursside eemaldamine, kasutada viimase sammuna
 - SIGPIPE 13 Toru maas (pole mõtet edasi kirjutada, sest keegi ei loe)
 - SIGTERM 15 Protsessi viisakas sulgemine, vaikimisi ja kõige ohutum viis protsess sulgeda
 - SIGUSR1 30,10,16 Kasutaja (programmeerija) poolt defineeritud signaal1
 - SIGUSR2 31,12,17 Kasutaja poolt defineeritud signaal2
- PID vaatamine rakenduse nime järgi: `pidof <rakendus>`
 - `ps -ef | grep <rakendus>`
- programmi sulgemine (vaikimisi signaaliga 15 ehk SIGTERM):
 - `kill $(pidof <rakendus>)` või ka `kill `pidof <rakendus>`` (mitme käsu kombineerimine)
 - `killall <rakendus>`

Signaalid 3

- Signaali saatmine protsessile toimub käsuga **kill**
 - **kill <pid1> <pid2>**
 - **kill -9 3242**
 - termineerimissignaali -9 (kill) saatmine protsessile 3242
 - **kill -TERM 9588**
 - termineerimissignaali -15 (term) saatmine protsessile 9588
- Signaale SIGKILL ja SIGSTOP ei saa ignoreerida ega töödelda programmi enda poolt
- sulgemine jõuga (täpse) protsessinime abil
 - **killall firefox** (vaikimisi SIGTERM 15)
 - **killall -15 firefox** (viisakas sulgemine)
 - **killall -9 firefox** (jõuga sulgemine)
 - <https://wiki.itcollege.ee/index.php/Killall>
 - [https://en.wikipedia.org/wiki/Signal_\(IPC\)#POSIX_signals](https://en.wikipedia.org/wiki/Signal_(IPC)#POSIX_signals)
- *zombie process* – vanemprotsessiga sideme kaotanud lõpetanud protsess, nende sulgemine ei pruugi kõige lihtsam olla kuid üldiselt vanemprotsessi sulgemine kaotab ka *zombie* protsessi
 - `ps -xal #vanemaprotsessi PID 4.veerus (PPID – Parent PID)`
 - PPID vaatamine
 - `ps j [PID]`
 - `pstree -sg <PID>`
 - `top #f` avab sätted, PPID kohal d lubab/keelab näitamise, nooleklahv paremale märgib PPID ja nooleklahviga üles, alla saab muuta järjekorda, nooleklahv vasakule või Enter kinnitab uue asukoha, q väljub sätetest
 - https://en.wikipedia.org/wiki/Zombie_process
 - <https://stackoverflow.com/questions/16944886/how-to-kill-zombie-process>

Signaalid 4

- MS Windows 10
 - protsesside vaatamine
 - cmd: *tasklist*
 - Powershell: *Get-Process* (ka: *ps*)
 - *taskkill /?* (protsessi sulgemine PID'i alusel)
 - *taskkill /f* (protsessi sulgemine nimega, jõuga)
 - avada Notepad ja kirjutada midagi sinna, seejärel käsuraal:
 - *taskkill /im notepad.exe #viisakas sulgemine, küsitakse faili salvestamist*
 - *taskkill /f /im notepad.exe #jõuga sulgemine, ei küsita faili salvestamist*
 - *taskkill notepad #jõuga sulgemine*
 - graafiline haldamine – Task Manager
 - CTRL+SHIFT+ESC
 - kaart *Details* näitab ka PID



The screenshot shows the Windows Task Manager Performance tab. At the top, it displays overall system usage: 99% CPU, 36% Memory, 2% Disk, and 0% Network. Below this, there are sections for 'Apps (1)' and 'Background processes (30)'. The 'Background processes' section is expanded to show a list of running processes with their respective resource usage.

Name	CPU	Memory	Disk	Network
Apps (1)				
Task Manager	0%	10.9 MB	0 MB/s	0 Mbps
Background processes (30)				
COM Surrogate	0%	2.4 MB	0.1 MB/s	0 Mbps
Device Association Framework Provider Host	0%	2.6 MB	0.1 MB/s	0 Mbps
Host Process for Windows Tasks	0%	2.3 MB	0 MB/s	0 Mbps
InstallAgent	0%	2.8 MB	0 MB/s	0 Mbps
InstallAgentUserBroker	0%	3.9 MB	0 MB/s	0 Mbps
KMS Connection Broker	0%	1.3 MB	0 MB/s	0 Mbps
Microsoft Account Background Task Host	0%	5.9 MB	0 MB/s	0 Mbps
Microsoft Malware Protection Command Line Utility	0.3%	2.0 MB	0.1 MB/s	2.0 Mbps
Microsoft Malware Protection Command Line Utility	0%	1.3 MB	0 MB/s	0 Mbps
Microsoft OneDrive (32 bit)	0%	5.3 MB	0 MB/s	0 Mbps
Microsoft OneDrive Setup (32 bit)	92.8%	42.1 MB	4.3 MB/s	0 Mbps
Microsoft OneDrive Setup (32 bit)	0%	2.3 MB	0 MB/s	0 Mbps
Microsoft Skype Preview	0%	2.7 MB	0 MB/s	0 Mbps
Microsoft Software Protection Platform Service	0%	5.8 MB	0.1 MB/s	0 Mbps
Microsoft Windows Search Filter Host	0%	0.9 MB	0 MB/s	0 Mbps
Microsoft Windows Search Indexer	0%	5.0 MB	0.1 MB/s	0 Mbps
Microsoft Windows Search Protocol Host	0%	1.6 MB	0 MB/s	0 Mbps
Runtime Broker	0%	6.3 MB	0 MB/s	0 Mbps
Search	0%	38.2 MB	0 MB/s	0 Mbps

taskkill'i kasutamine MS Windowsis <https://technet.microsoft.com/en-us/library/bb491009.aspx>
Process Explorer <https://docs.microsoft.com/en-us/sysinternals/downloads/process-explorer>
lisavõimalustega programm Process Hacker <http://processhacker.sourceforge.net/>



Tööd

- Vahel me ei soovi panna protsessi tööle esiplaanis
 - `./programm &`
 - Kui soovid luua faili, mis lõpeb märgiga `&`, siis kasuta apostroofe (ülakomasid) või paomärki `\`
 - Näiteks ***touch 'kalaätt&'*** või ***touch kala\&***
- Ülevaate taustal töötavatest programmidest
 - `jobs`
- Terminalis töötava programmi saab ajutiselt seisata klahvikombinatsiooniga `CTRL+Z` (SIGSTOP) ja lõpetada `CTRL+C` (SIGINT), näide (eriti mugav üle SSH, ei pea uut sessiooni looma)
 - `sudo nano /etc/network/interfaces #avame võrgusätteid`
 - `CTRL+Z #paneme faili muutmise taustale ootele`
 - `ifconfig -a #vaatame võrguliideste tähised`
 - `fg #naaseme faili ja kirjutame lõpuni`
 - `CTRL+O #salvestame`
- <http://superuser.com/questions/262942/whats-different-between-ctrlz-and-ctrlc-in-unix-command-line>
 - `CTRL+Z` paneb protsessi taustale (SIGSTOP signaaliga)
 - `CTRL+C` sulgeb protsessi (SIGINT signaaliga, INT - *interrupt*)



Tööd 2

- Esiplaanile toomine
 - ***fg <töö nr>***
- Tahaplaanile viimine
 - ***bg <töö nr>***
- Töö jõuga sulgemine (tapmine)
 - ***kill %<töö nr>***
 - ***kill %% (viimase töö tapmine)***
- Vaatame, mis protsessid on konkreetse rakendusega seotud:
 - ***pgrep ssh***
 - ***pgrep -u root ssh***



Proovimiseks

- Paigaldage programm cowsay
- Käivitage järgmised read ja vaadake tulemust (ja loodud faile)
 - ***sudo apt update && sudo apt install cowsay***
 - ***cowsay möööö***
 - ***cowsay -f sheep määä > lammas.txt***
 - ***cowsay Mis lammas >> lammas.txt***
- <https://wiki.itcollege.ee/index.php/Cowsay>

Küsimused? Tänan tähelepanu eest!



IT KOLLEDŽ
TALLINNA TEHNIKAÜLIKOOL



TTÜ IT KOLLEDŽ

Raja 4C, 12616 Tallinn

tel +372 628 5800

info@itcollege.ee

<http://www.itcollege.ee/>