

# Kompileerimine

Tarkvara kirjutatakse tarkvaraarendajate poolt inimloetavas programmeerimiskeeles, nimetame seda lähtekoodiks (*source code*). See teeb tarkvara loomise inimestele lihtsamaks. Näiteks Linuxi tuum on kirjutatud peamiselt C-keeles. Kuid siiski arvutid ei mõista seda. Selleks on vaja lähtekood muuta masinkoodiks ehk siis kompileerida, mida teeb spetsiaalne tarkvara: kompilaator. Erinevate programmeerimiskeelte jaoks on erinevad kompilaatorid. Näiteks *GNU C Compiler* (GCC) tõlgib C-keele lähtekoodi masinkoodiks. Lisaks saavad kompilaatorid ka optimeerida kompileeritud koodi vastavalt riistvarale. Lisaks on võimalik eemaldada ka mittevajalikke komponente.

Kui kasutame igapäevaselt mõnda rakendust siis on see juba kompileeritud ja rakenduse käivitamisel käivitatakse varasemalt kompileeritud binaarfail. Nii mõnigi kord nõuab kompileerimine aega ja võib osutada üsna keeruliseks. Seetõttu üldjuhul jagatakse rakendusi kompileeritud kujul.

Ubuntu Linuxis arendusvahendite paigaldamiseks:

```
sudo apt-get update && sudo apt-get install build-essential && sudo apt-get clean
```

Lihtne näide GCC abil „hello world” kompileerimiseks - [http://www.linfo.org/create\\_c1.html](http://www.linfo.org/create_c1.html)

Näidiskood:

```
#include<stdio.h>
/* demo.c: My first C program on a Linux */
int main(void)
{
    printf("Hello world!\n");
    return 0;
}
```

... salvestada see näiteks *helloworld.c* failina lemmiktekstiredaktoriga, nt nano abil.

Kompileerimiseks:

```
gcc helloworld.c -o helloworld #-o output
```

```
make helloworld #eeldusel, et helloworld.c on olemas
```

```
chmod +x helloworld #muudame käivitatavaks
```

```
./helloworld #käivitame
```

Veel mõned näited <https://www.linux.com/learn/howdy-ubuntu-windows-writing-compiled-languages>

## Juhtprogrammide haldus

Riistvara hakkab tööle siis kui operatsioonisüsteemil on selle jaoks olemas vastav komponent, mida nimetame juhtprogrammiks (*driver*):

- kohe OSiga kaasas (tuuma kompileeritud)
- hiljem lisatav

Mõnikord võib vajalik olla eraldi püsivara failide lisamine, vt [näidet puuduva püsivara veateate korral](#).

Juhtprogrammide tugi on tuumas, uuema tuuma lisamine võib ka uuema riistvara tööle panna. Linuxi osas vt <http://enos.itcollege.ee/~edmund/materials/kernel/kernel-install-and-management.txt>

MS Windowsi osas ei ole mitme tuuma kasutamist võimaldatud.

## Linuxi tuuma moodulid

Tuum on OSi kõige tähtsam programm, mis korraldab kogu OSi tööd. Iga OSi tuum on erinev. Tuuma ülesanne on suhelda riistvaraga ja tarkvaraga ning hallata parimal viisil süsteemi ressursse. Riistvaraga

suheldakse juhtprogrammide kaudu, mis on sisuliselt tuuma moodulid, asuvad Linuxis `/lib/modules/$(uname -r)/` kaustas (täpsemalt: `/lib/modules/$(uname -r)/kernel/drivers/`) ja on `.ko` tüüpi failid 3.x ja uuema tuuma versiooni korral (2.6 ja vanemate tuumade korrali olid `.o` tüüpi failid). Tuuma mooduleid ehk siis sisuliselt juhtprogramme hallatakse käsuga `modprobe`:

`lsmod` #laaditud tuuma moodulite vaatamine, kuvab `/proc/modules` faili sisu

`less /lib/modules/$(uname -r)/modules.dep` #moodulite sõltuvuste faili vaatamine, lisainfo `man modules.dep`

`modinfo <mooduli nimi>` #laaditud tuuma mooduli info vaatamine

`modprobe <mooduli nimi>` # tuuma mooduli laadimine

`insmod <mooduli nimi>` # tuuma mooduli laadimine

`modprobe -r <mooduli nimi>` # tuuma mooduli eemaldamine

`rmmod <mooduli nimi>` # tuuma mooduli eemaldamine

`depmod -a` #korrastab kõikide moodulite sõltuvused (loob failid `/lib/modules/$(uname -r)/modules.dep`, `/lib/modules/$(uname -r)/modules.dep.bin` faili), lisainfo `man depmod`

`man modprobe` #abiinfo

`man insmod` #abiinfo

`man lsmod` #abiinfo

Näiteks virtuaalmasinas on olemas moodul `e1000`, mis on Intel PRO/1000 võrgukaardi juhtprogramm:

`lsmod | grep e1000`

`modinfo e1000`

Selle mahalaadimisel ka võrk lakkab töötamast: `modprobe -r e1000`

Uuestilaadimisel hakkab ka võrk taas tööle: `modprobe e1000`

**Moodulid, mida soovitakse algaadimisel laadida**, kirjutatakse `/etc/modules` faili (iga mooduli nimi uuel realt), lisainfo `man modules`. Koos `systemd` (süsteemi ja teenuste haldur) tulekuga on eraldi kaustad `/etc/modules-load.d/`, `/usr/lib/modules-load.d/` kus on `*.conf` failid (failinimi sisaldab tavaliselt mooduli nime), mis sisaldavad samuti mooduli nime, iga nimi uuel realt. Lisainfo `man modules-load.d`

Tuuma mooduli täpsemad parameetrid pannakse kirja `/etc/modprobe.d/<mooduli nimi>.conf` failidena.

**Musta nimekirja** (*blacklist*) pannakse juhtprogramme, mida ei soovita laadida süsteemi käivitumisel. Selle põhjuseks võib olla soov kasutada mõne teise valmistaja juhtprogrammi vms. Musta nimekirja ei saa panna tuuma mooduleid, mis on tuuma otse sisse kompileeritud ega ole eraldi `.ko` tüüpi failidena. Need pannakse kirja `/etc/modprobe.d/blacklist-<mooduli nimi>.conf` failidena või ka eraldi failis `/etc/modprobe.d/blacklist.conf` ja peale selle muutmist käivitada:

`sudo update-initramfs -u` (lisainfo: `man update-initramfs`)

Ka süsteemi käivitumisel saab üksikuid tuuma mooduleid keelata laadimast, muutes faili `/etc/default/grub`:

`GRUB_CMDLINE_LINUX_DEFAULT="quiet splash`

`modprobe.blacklist=<moodul1>,<moodul2>,...,<moodulx>"`

`sudo update-grub` #peale GRUBi sätete muutmist tuleb ka uuendada `/boot/grub/grub.cfg` faili.

## MS Windows'i juhtprogrammid

MS Windowsi puhul pakutakse juhtprogramme kas paigaldusnõustajaga (*install wizard*) koos või .inf failidena, millel hiire paremklõpsu tehes saab valida "Install". Neid tegevusi tuleb teha superkasutaja õigustes. Juhtprogrammide haldamiseks on olemas Device Manager:

Super ->Run ->devmgmt . msc

Seadmehaldur Device Manager [https://en.wikipedia.org/wiki/Device\\_Manager](https://en.wikipedia.org/wiki/Device_Manager)

Win7 juhtprogrammide haldus <https://technet.microsoft.com/en-us/library/dn249968.aspx> ,  
<https://www.lifewire.com/device-manager-2625860>

Win10 eripärad <https://www.makeuseof.com/tag/take-back-control-driver-updates-windows-10/>

MS Windowsi juhtprogrammide tüübid <https://docs.microsoft.com/en-us/windows-hardware/drivers/kernel/types-of-windows-drivers>

Raamistik Windowsi juhtprogrammide jaoks, Windows Driver Model

[https://en.wikipedia.org/wiki/Windows\\_Driver\\_Model](https://en.wikipedia.org/wiki/Windows_Driver_Model) , <https://docs.microsoft.com/en-us/windows-hardware/drivers/kernel/windows-driver-model>

## Linux'i tuuma kompileerimine

<https://www.kernel.org/doc/> - tuuma dokumentatsioon

See võib vajalikuks osutuda kui soovitakse:

- lubada eksperimentaalset funktsionaalsust, mis vaikimisi ei ole lubatud
- keelata vaikimisi lubatud funktsionaalsust
- luua oma tarkvara, mis vajab tööks muudetud tuuma

Paigaldada kompileerimiseks vajalikud paketid (kui eelnevalt ei ole paigaldatud):

```
sudo apt-get update && sudo apt-get install build-essential && sudo apt-get clean
```

Vajadusel lisada: *git fakeroot ncurses-dev xz-utils libssl-dev bc*

Kontrolliks:

```
whereis gcc
```

```
which gcc
```

```
gcc --version
```

Vaja on ka ~15...20 GB vaba kettaruumi ja internetiühendust tuuma lähtekoodi allalaadimiseks. Tuuma lähtekoodi saab alla laadida aadressilt <https://www.kernel.org/> -> „tarball” ehk siis täielik lähtekood (\*.tar.xz). See tuleb lahti pakkida:

```
tar xf linux-x.xx.x.tar.xz #asendada x tegeliku versiooninumbriga
```

- *arch* näitab Linux'i tuuma arhitektuuri
- *crypto* krüptoalgoritmide tugi, nt AES, DES
- *drivers* juhtprogrammide moodulid, nt IDE, SCSI, juhtmega ja juhtmevabad võrguliidesed jne
- *fs* failisüsteemide tugi, nt ext4, FAT, NTFS
- *net* võrguprotokollide tugi, nt IPv4, IPv6, tcp, 802.11

```
cd linux-x.xx.x #siseneda lahtipakitud kausta
```

Enne uue tuuma kompileerimist tuleb seadistada, millised moodulid pannakse tuuma kaasa ja millised mitte. Lihtsam viis on kopeerida olemasoleva tuuma sätetefail ja seejärel seadistada tuum:

```
cp /boot/config-$(uname -r) .config
```

Kompileerimiseks on käsk *make*, lisainfo *man make*

*make menuconfig* #käsureal toimiv ncurses teeki kasutatav liides, <https://en.wikipedia.org/wiki/Ncurses>

*make nconfig* #uuem ncurses teegil toimiv kasutajaliides

*make xconfig* #kasutajasõbralik graafiline kasutajaliides Qt teegi baasil, mis nõuab paketi *packagekit-qt5* paigaldamist sõltuvusena. See on soovituslik valik – eriti vähemkogenud kasutajatele kuna siin on lihtsam liikuda ja info iga sammu kohta on samuti kuvatud.

*make gconfig* # graafiline kasutajaliides sarnaselt xconfig'ile kuid kasutab Gtk teeki

Kompileerimiseks on käsk *make -j x* #kus x on protsessorituumade arv, mida soovime kasutada, vaatamiseks käsk *nproc* – kui seda teha virtuaalmasinas (VM) siis on vaja eelnevalt lubada VirtualBox'i sätetest rohkem tuumi, selleks tuleb VM vahepeal seisma panna

Tuuma moodulite paigaldamiseks: *sudo make modules\_install -j x*

Tuuma paigaldamiseks: *sudo make install -j x*

Ubuntu puhul tasub vaadata <https://wiki.ubuntu.com/KernelTeam/GitKernelBuild>

.deb failide paigaldamiseks: *sudo dpkg -i \*.deb*

Enne GRUB'i uuendamist uuendada ka algne muutmäluketas (initial ramdisk):  
*update-initramfs -c -k x.x.x* #asendada x tegeliku tuumaversiooniga

Viimaks uuendada alglaadur GRUB:

```
sudo update-grub
```

Seejärel on aeg arvuti taaskäivitada ja proovida uut tuuma.

## Viited

<https://wiki.ubuntu.com/KernelTeam/GitKernelBuild>

[https://wiki.archlinux.org/index.php/Kernels/Traditional\\_compilation](https://wiki.archlinux.org/index.php/Kernels/Traditional_compilation)

<https://www.linux.com/learn/how-compile-linux-kernel>

<https://medium.freecodecamp.org/building-and-installing-the-latest-linux-kernel-from-source-6d8df5345980>

<https://www.makeuseof.com/tag/compile-linux-kernel/>

<https://www.wikihow.com/Compile-the-Linux-Kernel>

<https://stackoverflow.com/questions/2310261/what-is-the-difference-between-compile-code-and-executable-code>

Initial ramdisk

[https://en.wikipedia.org/wiki/Initial\\_ramdisk](https://en.wikipedia.org/wiki/Initial_ramdisk)

<https://www.linux.com/learn/kernel-newbie-corner-initrd-and-initramfs-whats>

<https://wiki.ubuntu.com/Initramfs>

<https://wiki.debian.org/initramfs>

<http://www.linuxfromscratch.org/blfs/view/svn/postlfs/initramfs.html>

Systemd

<https://en.wikipedia.org/wiki/Systemd>

<https://www.freedesktop.org/wiki/Software/systemd/>

<https://wiki.ubuntu.com/systemd>

<https://wiki.ubuntu.com/SystemdForUpstartUsers>

<https://wiki.debian.org/systemd>

<https://wiki.archlinux.org/index.php/systemd>

Linixi juhtprogrammid

<https://www.cyberciti.biz/faq/inux-kernel-where-to-find-modules/>

<https://www.cyberciti.biz/faq/add-remove-list-linux-kernel-modules/>

[https://help.ubuntu.com/community/Loadable\\_Modules](https://help.ubuntu.com/community/Loadable_Modules)

<https://askubuntu.com/questions/110341/how-to-blacklist-kernel-modules>

<https://wiki.debian.org/KernelModuleBlacklisting>

<https://unix.stackexchange.com/questions/10254/how-to-change-the-order-of-the-network-cards-eth1-eth0-on-linux>

<https://www.kernel.org/doc/Documentation/kbuild/modules.txt> – tuumaväliste moodulite ehitamine

Kõrgkeel

<https://et.wikipedia.org/wiki/K%C3%B5rgkeel>

[https://en.wikipedia.org/wiki/High-level\\_programming\\_language](https://en.wikipedia.org/wiki/High-level_programming_language)

C-keel

[https://et.wikipedia.org/wiki/C\\_\(programmeerimiskeel\)](https://et.wikipedia.org/wiki/C_(programmeerimiskeel))

[https://en.wikipedia.org/wiki/C\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/C_(programming_language))

Masinkood

<https://et.wikipedia.org/wiki/Masinakeel>

[https://en.wikipedia.org/wiki/Machine\\_code](https://en.wikipedia.org/wiki/Machine_code)