

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Marko Esna 176045IDAR

**PRAKTIKUMIMATERJAL: UBUNTU
LINUXI TÖÖJAAMA PAIGALDAMISE
AUTOMATISEERIMINE**

Diplomitöö

Juhendaja: Edmund Laugasson
MSc

Tallinn 2020

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Marko Esna

18.05.2020

Annotatsioon

Käesoleva diplomitöö eesmärgiks on praktikumimaterjali koostamine õppeainele „Operatsioonisüsteemid ja nende haldamine”. Praktikumimaterjal koosneb kahest osast: Ubuntu Linuxi kohandatud ISO-faili loomine automaatpaigalduseks ning Ubuntu Serveri kohandamine võrgust alglaadimisserveriks. Materjali läbinud tudengid saavad kogemuse ning teadmised automatiseerimisprotsesside läbiviimiseks.

Töö käigus luuakse eelnevalt seadistatud vastusfail (*preseed*), mis võimaldab Ubuntu MATE Desktopi automaatset paigaldamist kahes etapis: ilma ja koos paigaldusserveriga. Serveri kasutamine annab võimaluse teostada võrgupõhist paigaldamist paralleelselt mitmele arvutile kasutades võrguliideselt alglaadimist (*PXE*) ja oluliselt lihtsustada vastusfaili muutmist.

Töö rõhk on praktilisel teostusel ja võimalusel tehtut ka kontrollida. Kasutatakse vabavaral põhinevaid tehnoloogiaid, mistõttu loodud praktikumimaterjal on kasutatav kõigile sõltumata eelarvest. Töö tulemusena valmis praktikumimaterjal, mis on läbinud ka esmase testimise sama õppeaine tudengite hulgas.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 22 leheküljel, 5 peatükki, 10 joonist, 1 tabel.

Abstract

Lab Package: Automating Ubuntu Linux Workstation Deployment

The purpose of current thesis is to create a lab package for the course „Operating systems and its management”. Lab package consists of two parts: Creating a customized Ubuntu Linux ISO for automatic installation and creating a Ubuntu Linux PXE boot server. After completing the materials the students will get experience in automation processes.

During the lab there will be created Ubuntu MATE Desktop automated installation in two stages: with and without installation server. Using the installation server gives us opportunity to deploy multiple installations simultaneously via PXE boot and significantly simplify editing preseed file.

The emphasis of the work is on practical implementation and, if possible, on checking what has been done. During the lab free software based technologies are used, so the created material is available to everyone, regardless of budget. As a result of the work, the practicum material was completed, which has also passed the initial testing among the students of the same subject.

The thesis is in Estonian and contains 22 pages of text, 5 chapters, 10 figures, 1 table.

Lühendite ja mõistete sõnastik

<i>DHCP</i>	<i>Dynamic Host Configuration Protocol</i> , Protokoll, mis väljastab automaatselt IP aadresseid
<i>Free software</i>	Vabatarkvara (tarkvara, mida kasutaja võib muuta ja levitada)
<i>Initrd</i>	<i>Initial ramdisk</i> , Linuxi käivitusprotsessi osa, mille käigus laaditakse juurfailisüsteem ajutiselt muutmällu esmaseks kasutamiseks
<i>ISO-file</i>	ISO9660 ¹ standardile vastav alglaaditav tõmmisfail
<i>NFS</i>	<i>Network File System</i> , Võrgu-failisüsteem
<i>Preseed</i>	Eelseadistus või vastusfail
<i>PXE</i>	<i>Preboot Execution Environment</i> , Keskkond alglaadimiseelseteks toiminguteks
<i>TFTP</i>	<i>Trivial File Transfer Protocol</i> , Ülilihntne failiedastus protokoll

1 <https://www.iso.org/obp/ui/#iso:std:iso:9660:ed-1:v1:en>

Sisukord

1	Sissejuhatus.....	9
2	Metoodika.....	10
2.1	Lähtetingimused.....	10
2.2	Vabavaralised automatiseerimistarkvarad.....	11
2.2.1	FAI – <i>Fully Automated Installation</i>	11
2.2.2	Cobbler.....	12
2.3	Kasutatud tehnoloogiad.....	12
2.3.1	Vastusfail (<i>preseed</i>).....	12
2.3.2	PXE alglaadmine.....	13
2.3.3	TFTP.....	13
2.3.4	DHCP.....	14
2.3.5	NFS.....	14
3	Praktikumimaterjal.....	15
3.1	Ülesehitus.....	15
3.2	Ubuntu Linux automaatpaigaldus ilma serverita.....	15
3.3	Ubuntu Linux automaatpaigaldus koos PXE-serveriga.....	19
3.4	Tagasiside küsimustik.....	23
4	Analüüs.....	24
4.1	Tagasiside küsitlus.....	24
4.2	Materjali vastavus lähtetingimustele.....	27
4.3	Tulevikuarendused.....	28
5	Kokkuvõte.....	30
	Kasutatud kirjandus.....	31
	Lisa 1 – Praktikumimaterjal.....	33
	Lisa 2 – Tagasiside küsitlus.....	73

Jooniste loetelu

Joonis 1: Alglaadimisparameetrite muutmine failis.....	17
Joonis 2: Eelseadistus fail.....	18
Joonis 3: Virtuaalmasina loomise skript.....	19
Joonis 4: Topoloogia.....	20
Joonis 5: dnsmasq konfiguratsioon.....	20
Joonis 6: Sätetefaili otsingu ülesehitus.....	22
Joonis 7: Materjali sobilikkus.....	24
Joonis 8: Materjali arusaadavus.....	25
Joonis 9: Materjali mahukuse hinnang.....	26
Joonis 10: Ajakulu materjali läbimiseks.....	27

Tabelite loetelu

Tabel 1: Vastavus lähtetingimustele.....	27
--	----

1 Sissejuhatus

Tänapäeval on kasvanud koolides ning ettevõtetes kasutatavate tööjaamade ning sülearvutite arv märkimisväärselt. Tihtipeale on asutustes ka palju sarnase profiili või konfiguratsiooniga seadmeid, mille paigaldus on ühesugune. Sellest tingituna on seadmetele ükshaaval operatsioonisüsteemide paigaldamine ebaotstarbekas. Sellepärast on loodud erinevaid lahendusi, kuidas seda teha massiliselt ning automaatselt. Eelkõige aitab see kokku hoida aega, kuid samuti on lihtsam sisse viia ka hilisemaid muudatusi.

Kui vaadata hetkel IT süsteemide administreerimise õppekava aineid [1], siis leiab sealt ainult ühe automatiseerimisega seotud aine Automaattestimine (ICD0004) . Tegemist valikaineaga, mis keskendub rohkem tarkvaraarendusele. Autor soovib seda olukorda parandada, ning lisada õppekavasse automatiseerimist käsitlevat õppematerjali.

Käesoleva diplomitöö eesmärk on koostada eestikeelne praktikumimaterjal aines „ICA0001 Operatsioonisüsteemid ja nende haldamine” kasutades vaid vabavaralisi vahendeid. Praktikumimaterjali sisuks on automatiseeritud tööjaama paigaldus Ubuntu Linuxil näitel ilma ja koos paigaldusserveriga. Loodav materjal võimaldab tehtut kontrollida ja seeläbi pakub kasutusvõimalust eri tasemel tudengitele. Kuna tegemist on 1. kursuse ainega, siis on tudengitel hea võimalus varakult tutvuda automatiseeritud lahenduse loomisega.

Töö sisuline osa koosneb kolmest peatükist. Esimeses peatükis kirjeldatakse meetodika, kuidas valiti lahendused praktikumimaterjali loomiseks. Teises peatükis antakse ülevaade loodud praktikumimaterjalist. Kolmandas peatükis analüüsitakse tudengitelt saadud tagasisidet loodud materjali kohta.

2 Metoodika

Käesolevas peatükis antakse ülevaade õppeainest Operatsioonisüsteemid ja nende haldamine, ning kirjeldatakse, kuidas tehti valikud praktikumimaterjali loomiseks kasutatavate lahenduste osas.

2.1 Lähtetingimused

Operatsioonisüsteemid ja nende haldamine on 1. kursuse õppeaine, mis kuulub IT süsteemide administreerimise õppekava kohustuslike ainete koosseisu. Kursuse eesmärgiks on anda olulisemad põhiteadmised ning praktilised oskused operatsioonisüsteemide haldamiseks. Loengutes pannake rõhku üldisematele teadmistele, aga praktikumides omandatakse oskused ühe süsteemi abil, milleks on valitud Ubuntu Linux. Põhiliseks tööriistaks, mida kursusel kasutatakse, on riistvara virtualiseerimistarkvara VirtualBox [2]. Lisaks sellele kasutatakse enamus toimingute läbiviimiseks pigem käsurida kui graafilist liidest. Sellest tulenevalt sõnastati lähtetingimused praktikumimaterjali loomiseks:

- Materjali loomisel arvestatakse erinevate IT-teadmiste tasemega tudengitega ehk seletatakse samm-sammult tehtavad toimingud
- Tudeng saab kontrollida sooritatud sammu õigsust
- Tegevused saab sooritada VirtualBox virtuaalmasinas
- Luuakse automaatselt paigalduv Ubuntu Linux tööjaam koos seletavate juhistega
- Luuakse Ubuntu Linxi tööjaama automaatse paigalduse server koos seletavate juhistega
- Ülesanded on eraldiseisvad, see tähendab praktikumi saab läbida ka ilma automaatse paigalduse serverita

- Kasutatav tarkvara on vabavara, et ei oleks takistusi tarkvara paigaldamisel ja kasutamisel

2.2 Vabavaralised automatiseerimistarkvarad

Tulenevalt eesmärgist kasutada vaid vabavaralisi vahendeid sai uuritud levinud Linuxi automaatpaigalduse rakendusi kasutades internetiotsingut ja lisaks alternatiivide otsingumootorit¹, mis võimaldab piirata otsingut litsentsitüübi ja operatsioonisüsteemi alusel. Kasutatud märksõnad:

- top free linux desktop automated deployment software
- top free linux desktop provisioning software
- top free linux desktop automated provisioning software
- unattended ubuntu install
- ubuntu pxe server

Siin kirjeldatud rakendused annavad ülevaate ning seeläbi ka suunised praktikumimaterjali koostamisel kasutatud tehnoloogiate osas.

2.2.1 FAI – *Fully Automated Installation*

FAI [3] on operatsioonisüsteemide massilise paigalduse tarkvara, mis on keskendunud ainult Linuxile. Seda kasutatakse Linuxi süsteemide ning nende tarkvarapakettide paigaldamiseks ja seadistamiseks arvutitesse kui ka virtuaalmasinatesse. Sobib nii väikeste klassiruumide kui ka suuremahuliste süsteemide, näiteks klastrite ja virtuaalsete keskkondade paigaldamiseks.

FAI paigaldus käivitatakse võrgukaardilt, CD-lt või USB-mälupulgalt. Ta saab IP-aadressi ja käivitab Linuxi tuuma, mis paigaldab juurfailisüsteemi paigaldusfailist NFSi kaudu. Pärast tuuma käivitamist teostab FAI käivitamiskript automaatse paigalduse, mis ei vaja sekkumist. Esmalt luuakse kettajaod, seejärel luuakse failisüsteemid ning

1 <https://alternativeto.net>

viimaks paigaldatakse tarkvarapaketid. Pärast seda seadistatakse värske operatsioonisüsteem vastavalt vajadustele mõne skripti abil. Lõpuks käivitatakse uus operatsioonisüsteem kohalikult kettalt.

2.2.2 Cobbler

Cobbler on Linuxi paigaldusserveri tarkvara, mis võimaldab erinevat tüüpi toiminguid Linuxi tööjaamade ja serverite paigaldamiseks ning seadistamiseks [4]. Cobbleri kasutamiseks eeldatakse kasutajalt häid teadmisi PXE alglaadmise ning valitud operatsioonisüsteemi paigaldamise automatiseerimise meetodite kohta. Cobbler kasutab oma funktsionaalsuse saavutamiseks enamjaolt erinevaid *Pythoni* teeke. Tuntud ettevõtted nagu Dell¹ ja TomTom² kasutavad just Cobblert.

2.3 Kasutatud tehnoloogiad

2.3.1 Vastusfail (*preseed*)

Vastusfaili kasutamine annab võimaluse eelnevalt määrata vastused küsimustele, mida tavaliselt paigaldamise ajal küsitakse. Samuti võimaldab see muuta paigaldamise automaatseks ning pakub ka mõningaid võimalusi, mida tavapaigalduse juures määrata ei saa. Vastusfail ei ole paigalduse jaoks kohustuslik, seega tühja või ka osaliselt vastamata vastusfaili puhul käitub paigaldusprotsess analoogselt tavatingimustele.

Tuginedes ametlikule dokumentatsioonile on suurimaks piiranguks vastusfaili kasutamise puhul puuduv võimekus kasutada varasemaid kõvakettajagusid. Seega tuleb alati luua uued kettajaod või kasutada kogu vaba ruumi [5].

Vastusfaili kasutamiseks on kolm võimalust: *initrd*, fail ja võrk. *Initrd* puhul on kogu vastusfaili sisu eelnevalt lisatud *initrd* faili ning käivitatakse kohe paigalduse alguses. Kasutades vastusfaili faili kujul *faili_nimi.seed*, on tarvis see eelnevalt valitud ISO-sse lisada */preseed* kataloogi. Ning vastusfaili loetakse pärast seda, kui CD/DVD on eelnevalt laetud. Võrgu puhul laetakse vastusfail alla siis kui võrguseadistused on

1 www.dell.com

2 www.tomtom.com

tehtud. Vastusfailis defineeritavate küsimuste vastused tuleb siduda vastavate muutujatüüpidega:

- *string* - kasutaja võib sisestada ükskõik millise väärtuse
- *password* - analoogne *string* tüübile, kuid ei kuvata väärtuseid
- *boolean* - kasutatakse jah/ei (*yes/no*) või tõene/väär (*true/false*) tüüpi küsimuste vastamiseks
- *select* - kasutaja saab valida ühe valiku nimekirjast
- *multiselect* - kasutaja saab valida üks, mitu või mitte ühtegi valikut nimekirjast
- *note* - kasutatakse sõnumi kuvamiseks kasutajale

2.3.2 PXE algladmine

PXE (*Preboot eXecution Environment*) on standardiseeritud klient-server keskkond, mis kasutab algladimiseks läbi võrgu saadud tarkvara. Selle toimimise eelduseks on klientarvutil PXE toega võrgukaart [6].

Põhilised protokollid, mida PXE kasutab, on DHCP ja TFTP. DHCP on vajalik, et seade saaks võrguseadistuse, et oleks võimalik suhtlus teiste seadmetega võrgus. TFTP on vajalik, et laadida algladimiseks vajalikud failid seadmesse.

Et kasutada klientarvutil PXE keskkonda, tuleb see eelnevalt võimaldada. Kui riistvaraline tugi on olemas, siis saab seda teha arvuti BIOS/UEFI keskkonnas. Lisaks sellele on võimalik ka *wake-on-lan* funktsioon. See tähendab, et arvuti saab sisselülitada läbi võrgukaardi, kasutades selleks spetsiaalset signaali. PXE ei toeta wireless võrgukaarte.

2.3.3 TFTP

TFTP (*Trivial File Transfer Protocol*) on FTP (*File Transfer Protocol*) failivahetusprotokolli lihtsustatud versioon [7]. Suurim erinevus FTP suhtes seisneb selles, et FTP kasutab failide edastamiseks TCP protokoll, kuid TFTP kasutab UDP-d.

Üldjuhul puudub TFTP puhul ka võimalus loendada, ümber nimetada või kustutada faile või kaustu, mis asuvad serveris. Samuti puuduvad juurdepääsu kontrollmehhanismid nagu isikutuvastus, paroolid jms. Tänapäeval kasutatakse TFTP-d kõige enam just teostamiseks võrgualglaadimist.

2.3.4 DHCP

DHCP (*Dynamic Host Configuration Protocol*) on protokoll, mis võimaldab automaatselt jagada dünaamilisi IP-aadresse võrku ühendatud seadmetele ning vajadusel ka staatilisi IP-aadresse [8]. Viimaseid eelkõige serveritele. Lisaks sellele võib DHCP server jagada ka ajaserverit, arvuti nime, domeeni nime. Võrgualglaadimise puhul on DHCP oluline, sest peale IP-aadressi saamist, saab selle abil klientarvuti ka info TFTP serveri aadressi.

2.3.5 NFS

NFS (*Network File System*) ehk võrgu-failisüsteem on protokoll, mis laseb arvutil kasutada näiteks serveris asuvat failisüsteemi, nagu paikneks see kohalikus masinas. See osa failisüsteemist või terve süsteem on ligipääsetav klientidele vastavalt sellele, millised õigused on igale failile antud. Suur pluss võrgu-failisüsteemi puhul on selle keskne haldus, mis vähendab vajadust kõvaketta ruumi järele individuaalsetes masinates. Miinuseks on ebaturvalisus [9].

3 Praktikumimaterjal

Käesolevas peatükis antakse ülevaade valminud praktikumimaterjalist ning selle kohta loodud tagasiside küsimustikust.

3.1 Ülesehitus

Olemasolevate tarkvarade uurimisel ning testimisel selgus, et nende kasutamine ei sobi õppeaine spetsiifikaga, kuna ei ole võimalik piisavalt täpselt jälgida, millised protsessid toimuvad. Samuti ei õnnestunud virtualiseeritud taristu peal tarkvarasid tööle saada piisava võimekusega, mis olid lähtetingimustes kirjas. Sellest tulenevalt loodi vajalik lahendus käsitsi ning tugineti selle loomisel eelnevas peatükis kirjeldatud tehnoloogiatele.

Valminud materjal on kirja pandud Google Docs formaadis Google Drive keskkonnas ning mahult 34lk. Google Docsi kasutati, et hilisem materjali täiendamine ning muutmine oleks lihtne. Töölehe sisu on jagatud kaheks suuremaks peatükiks: automaatpaigaldus ilma serverita ning automaatpaigaldus koos PXE-serveriga. Materjali läbinud tudengid saavad esmase kogemuse, kuidas luua vastusfaili, mille abil teostada Ubuntu Linuxi tööjaama automatiseeritud paigaldus. Lisaks sellele ka teadmised, kuidas ehitada algelist PXE-serverit, mille abil teostada võrgualglaadimist kasutades eelnevalt loodud vastusfaili.

3.2 Ubuntu Linux automaatpaigaldus ilma serverita

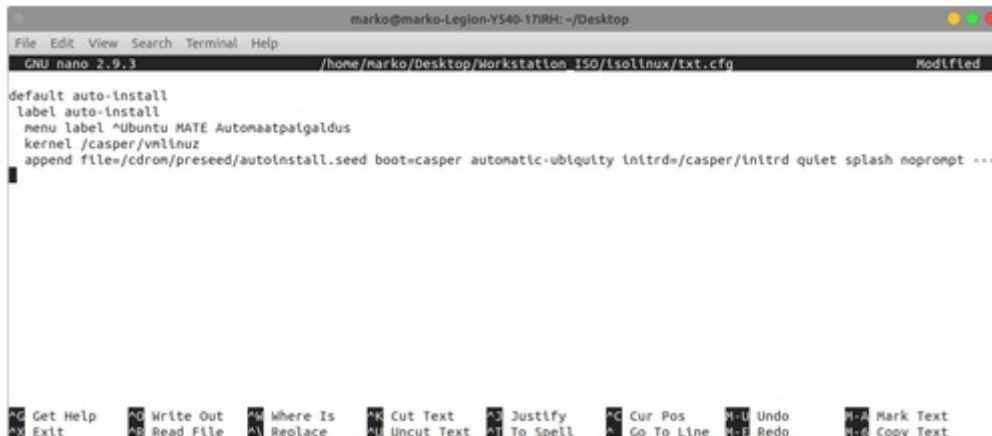
Praktikumimaterjali esimene osa annab ülevaate, kuidas on võimalik luua ISO faili, mida kasutades paigaldub Ubuntu Linux tööjaam automaatselt. Automaatselt tähendab, et kasutajalt ei küsita ühtegi paigaldusega seotud küsimust.

Esmalt kirjeldatakse, kuidas kasutada kasutaja kataloogide muutujaid käskudes, millega edasisi tegevusi sooritatakse. See andis võimaluse kirjutada käsud formaadis, mis ei sõltu kasutajanimest. Seejärel laetakse alla ametlik Ubuntu Linux MATE tööjaama ISO fail ning pakitakse selle sisu lahti, et teha vajalikud muudatused automatiseeritud paigalduse tarbeks. Ubuntu Linux MATE versiooni kasutatakse sellepärast, et see nõuab vähem ressursi kui Ubuntu Linux Gnome töölaua versioon. Seetõttu on seda ka tõhusam virtuaalmasinas kasutada.

ISO-faili lahtipakkimine teostatakse kasutades töölauapõhist arhiivihaldurit. Arhiivihaldurit kasutatakse sellepärast, et see ei nõua superkasutaja õiguseid. Alternatiivsed ISO-faili töötlemiseks mõeldud tarkvarad või moodused nõuavad superkasutaja õiguseid. Selliseid õiguseid tudengitele kooliarvutites ei anta. Alternatiivse lahendusena saab kasutada võimalust, et luuakse eraldi virtuaalmasin Ubuntu Linuxiga ning sooritatakse ISO-faili muutmise ja loomise toimingud seal. Sellisel juhul saab kasutada ka teisi tarkvarasid selle tarbeks nagu näiteks Cubic¹.

Järgmiseks näidatakse, millises failis teha muudatused ning seletatakse ülevahtlikult muudatuste sisu. Joonisel 1 on näha, kuidas muuta alglaadimise parameetreid *txt.cfg* failis:

1 <https://launchpad.net/cubic>



```
marko@marko-Legion-Y540-17RH: ~/Desktop
GNU nano 2.9.3 /home/marko/Desktop/Workstation_ISO/iso/linux.txt.cfg Modified
default auto-install
label auto-install
menu label ^Ubuntu MATE Automaatpaigaldus
kernel /casper/vmlinuz
append file=/cdrom/preseed/autoinstall.seed boot=casper automatic-ubiquity initrd=/casper/initrd quiet splash noprompt ---
Get Help Write Out Where Is Cut Text Justify Cur Pos Undo Mark Text
Exit Read File Replace Uncut Text To Spell Go To Line Redo Copy Text
```

default auto-install - määrab, millist valikut vaikimisi kasutatakse

label auto-install - valiku nimetus, ei pea olema auto-install, kasutaja saab ise nimetuse anda

menu label - alglaadimise menüüs kuvatav valiku nimetus

kernel - näitab tuuma asukohta meediumil

append - alglaadimise parameetrite lisamiseks

file - näitab preseed faili asukohta (see fail luuakse järgmises juhendi osas)

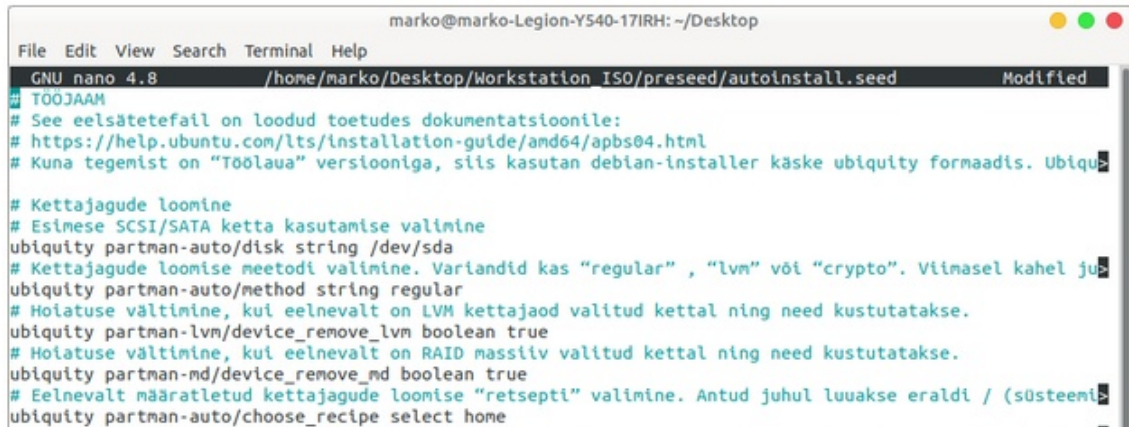
boot=casper - alglaadimine live süsteemi

automatic-ubiquity - paigalduskeskkonna automaatrežiimis käivitamiseks

Joonis 1: Alglaadimisparameetrite muutmine failis

Alglaadimiseparametrid, mida kasutatakse, on valdavalt analoogsed vaikimisi seadistusele. Lisatud on automatiseeritud paigalduse jaoks lisaparametrid *automatic-ubiquity* ning *noprompt*. Viimane tähendab, et paigalduse lõppedes ei küsita kasutaja käest meediumi eemaldamise küsimust. Näidatakse ära ka järgmises etapis loodava vastusfaili asukoht ja nimi. Et vältida probleeme võrgu või internetiga, kasutatakse faili asukohaks pigem ISO-faili ennast. Alternatiivina oleks saanud vastusfaili luua ka näiteks enose *public_html* kausta, mis oleks võimaldanud hiljem lihtsamal muuta seda ning eemaldada vajaduse iga kord ISO-fail uuesti luua.

Pärast seda luuakse vastusfail, mis moodustab materjali esimese osa tuumiku. Käsud, mida failis kasutatakse, on seletatud kommentaaridena nagu programmeerimises hea tava ette näeb. Joonisel 2 on näha vastusfaili ülesehitus:



```
marko@marko-Legion-Y540-171RH: ~/Desktop
File Edit View Search Terminal Help
GNU nano 4.8 /home/marko/Desktop/Workstation_ISO/preseed/autoinstall.seed Modified
# TOOJAAM
# See eelsätetefail on loodud toetudes dokumentatsioonile:
# https://help.ubuntu.com/lts/installation-guide/amd64/apbs04.html
# Kuna tegemist on "Töölaua" versiooniga, siis kasutan debian-installer käske ubiquity formaadis. Ubiquity
# Kettajagude loomine
# Esimese SCSI/SATA ketta kasutamise valimine
ubiquity partman-auto/disk string /dev/sda
# Kettajagude loomise meetodi valimine. Variandid kas "regular" , "lvm" või "crypto". Viimasel kahel ju
ubiquity partman-auto/method string regular
# Hoiatuse vältimine, kui eelnevalt on LVM kettajaod valitud kettal ning need kustutatakse.
ubiquity partman-lvm/device_remove_lvm boolean true
# Hoiatuse vältimine, kui eelnevalt on RAID massiiv valitud kettal ning need kustutatakse.
ubiquity partman-md/device_remove_md boolean true
# Eelnevalt määratletud kettajagude loomise "retsepti" valimine. Antud juhul luuakse eraldi / (süsteemi)
ubiquity partman-auto/choose_recipe select home
```

Joonis 2: Eelseadistus fail

Vastusfaili loomisel tugineti väga palju ametlikule Ubuntu paigalduse automatiseerimise dokumentatsioonile [10]. Kuna töölaua versioon kasutab *ubiquity* paigalduskeskkonda, siis automatiseeritud paigalduse jaoks lisandusid mõned piirangud ning alternatiivsed käsud [11]. Sellest tulenevalt esitati kõik käsud *ubiquity* formaadis, et käsud oleks ühtlasemalt jälgitavad.

Kuna vastusfaili loomise aluseks olev dokumentatsioon ei olnud ammendav, tuli uurida ka teiste kasutajate poolt loodud vastusfaile ning nende ülesehitust [12], [13]. Lõplik vastusfail loodi mitmete katsetuste tulemusel. Vastusfailis on kaetud paigalduse ajal küsitavad küsimused piisavalt, et tulemuseks oleks täielikult automaatne paigaldus. Lisati ka mõned lisatarkvara paigaldamise käsud.

Seejärel luuakse uus alglaaditav ISO-fail, mis sisaldab vastusfaili ning muudetud *txt.cfg* faili. Kuna eraldi tarkvara kasutada ei saanud ISO-faili muutmiseks, kasutati selle loomiseks käsku *mkisofs* [14], sest selle jaoks pole vajalikud superkasutaja õigused.

Alglaaditava ISO-faili kasutamiseks luuakse tühi virtuaalmasin. Virtuaalmasina loomise parameetrite aluseks võeti Ubuntu kodulehel soovitatud riistvaralised näitajad [15] v.a muutmälu, mille puhul testimiseks piisas mahust 2GB.

Boonusena loodi *bashi* skript, millega sai ka virtuaalmasina loomise muuta automaatsemaks. Skripti loomise ja kasutamise eesmärk oli kujutada reaalselt süsteemiadministraatori tööd. Selle jaoks kasutati VirtualBoxi käsurealiidese käsked [16].

```

# Kõik erinevad --ostype variandid saab kuvada käsuga: "VBoxManage list ostypes"
# --basefolder näitab, kus kaustas hakkab loodav virtuaalmasin paiknema.
VBoxManage createvm --name $MACHINE_NAME --ostype "Ubuntu_64" --register --basefolder $(xdg-user-dir DESKTOP)

# I/O APIC lubamine
VBoxManage modifyvm $MACHINE_NAME --ioapic on

# Muutmälu ja videomälu määratlemine
VBoxManage modifyvm $MACHINE_NAME --memory 2048 --vram 128

# Graafikakontrolleri valik. Valikuvariandid:
# -graphicscontroller none|vboxvga|vmsvga|vboxsvga
VBoxManage modifyvm $MACHINE_NAME --graphicscontroller vmsvga

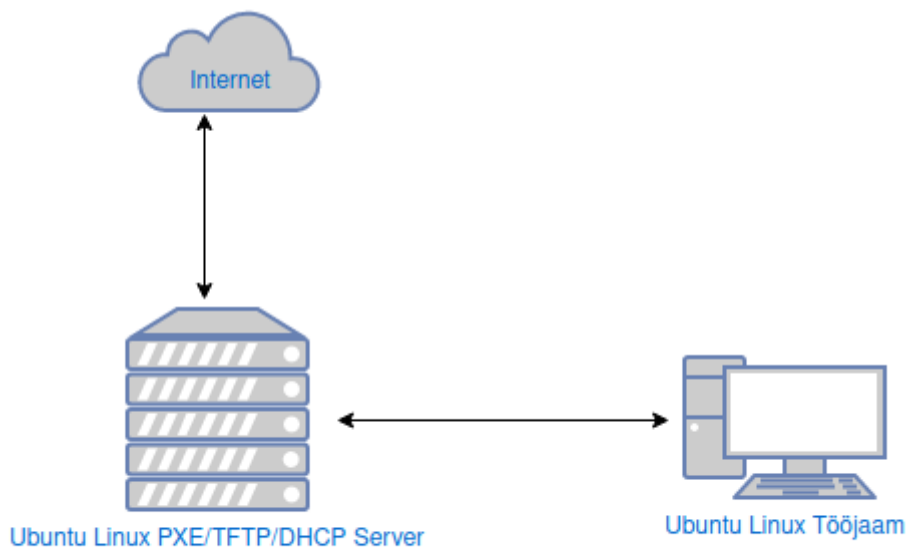
```

Joonis 3: Virtuaalmasina loomise skript

Skripti käivitamise tulemusel luuakse tühi virtuaalmasin, mis kasutab alglaadimiseks eelnevalt loodud ISO-faili ning teostub automaatpaigaldus. Kui enne skripti käivitamist polnud VirtualBox käivitatud, esines mõnikord anomaalia, kus virtuaalmasin ei läinud korrektselt tööle. Selle põhjust välja selgitada ei suudetud.

3.3 Ubuntu Linux automaatpaigaldus koos PXE-serveriga

Praktikumimaterjali teine osa kirjeldab, kuidas kohandada Ubuntu Linux server algeliseks PXE-serveriks. Tehnilise lahenduse loomisel võeti aluseks õpetus [17]. Esmalt luuakse uus virtuaalmasin, kuhu paigaldatakse Ubuntu Server. Serveri paigalduseks kasutatakse oskusi, mis on eelnevalt õppeaines omandatud. Kasutatakse kahte võrguliidest, üks internetiühenduse jaoks ning teine virtuaalmasinate vahelise võrgu (*internal network*) tarbeks. Internetiühendus on vajalik, et laadida serverisse lisapaketid, mida kasutatakse lõpliku funktsionaalsuse saavutamiseks.



Joonis 4: Topoloogia

Seejärel paigaldatakse serverisse *dnsmasq* ning *nfs-kernel-service* paketid, et seadistada server pakkuma DHCP, TFTP ja NFS teenust. *Dnsmasq* valiti, sest see sobib hästi väiksemahuliste võrkude ja projektide jaoks. Lisaks sellele võimaldab see pakkuda DHCP ja TFTP teenust samaaegselt. Alternatiivid oleks olnud eraldi paketid *tftpd-hpa* ning *isc-dhcp-server*. Joonisel 5 on näidatud *dnsmasq* seadistusfail.

```
# Liidese valik, mille kaudu DHCP ja TFTP teenust pakutakse
# Käsk, millega näha käsurealt kõiki võrguliideseid:
# netstat -i | cut -d' ' -f1 | sed 1,2d -
interface=ens0s8

# DHCP parameetrid vastavalt järjekorrale:
# Kasutatav liides, DHCP vahemiku algus ja lõpp, võrgumask, IP liisingu kestvus tundides
dhcp-range=ens0s8, 192.168.0.25, 192.168.0.50, 255.255.255.0, 24h

# TFTP teenuse aktiveerimine
enable-tftp

# TFTP juurkataloogi määramine
tftp-root=/srv/tftp

# Failinimi, mida klientarvuti saab alglaadimiseks kasutada ning IP mille kaudu seda küsida
dhcp-boot=pxelinux.0,192.168.0.100
```

Joonis 5: dnsmasq konfiguratsioon

NFS on vajalik, et klientarvuti saaks ligipääsu paigaldusfailidele, kui võrgust alglaadimine on teostatud.

Lisaks paigaldatakse ka paketid *pxelinux* ja *syslinux*. Nendest saab kätte vajalikud algladuri failid ning algladimismenüü kuvamiseks tarvilikud failid.

Edasi luuakse kataloogihierarhia, kuhu majutatakse eraldi vajalikud failid võrgust algladimise jaoks ning Ubuntu Linux MATE paigalduse jaoks. Lähtudes FHSi [18] standardist, kasutatakse juurkaustana */srv* kausta. Lisaks luuakse sinna alamkaustad */srv/tftp* ning */srv/nfs*. TFTP kasuta paigutatakse failid, mis saadi *pxelinux* ja *syslinux* pakettidest (vt Lisa 1). Lisaks luuakse samasse kausta *pxelinux.cfg* kaust, kuhu tekitatakse algladimiseks vaikimisi fail *default*.

Järgmisena laetakse serverisse Ubuntu Linux MATE töölaue versiooni ISO-fail. Pärast ISO-faili külge haakimist kopeeritakse kogu sisu */srv/nfs* loodud alamkataloogi *ubuntu_mate*. Selle kataloogi juurde pööratakse pärast algladimise teostamist, et saada paigalduseks vajalikud failid. Lisaks kopeeritakse veel ISO-failist algladav tuum (*vmlinuz*) ning ajutine juurfailisüsteem (*initrd*) */srv/tftp* kausta, et võimaldada algladimine.

Sarnaselt praktikumi esimesele osale muudetakse algladimisparameetreid (failis *default*), et võimaldada automaatpaigaldus. Laetakse alla esimeses osas loodud vastusfail, ning kohandatakse seda virtuaalmasinate sisevõrgu tarbeks (vt Lisa 1).

Lõpuks luuakse uus tühi virtuaalmasin käsitsi või kasutades muudetud virtuaalmasina loomise skripti (vt Lisa 1) ning teostatakse algladimine võrgust, millele järgneb operatsioonisüsteemi automaatpaigaldus. Virtuaalmasina loomiseks kasutatakse samasuguseid jõudlus parameetreid nagu eelmises praktikumi osas. Erinevus on võrguliidese seadistuse osas, mis antud juhul peab olema valik *internal network*.

PXE algladimise protsess [19] Linuxil näitel:

- Klientmasin käivitatakse (eelnevalt BIOS-is määratud algladimise teostamine kasutades võrguliidest)
- Võrgukaart käivitab DHCP päringu

- DHCP server vastab päringule standardse infoga nagu IP-aadress, võrgumask, vaikelüüsi aadress jne. Lisaks sellele saadetakse ka TFTP-serveri aadress ning alglaadimistõmmise (*boot image*) *pxelinux.0* info
- Kui klientmasin on selle info vastu võtnud, luuakse ühendus TFTP-serveriga, et alglaadimistõmmis allalaadida
- TFTP-server saadab alglaadimistõmmise ning klient käivitab selle
- Vaikimisi otsib tõmmis TFTP-serveris kausta *pxelinux.cfg*, kus paiknevad alglaaduri sätetefailid
- Esmalt otsitakse MAC-aadressile vastava nimega sätetefaili. Kui sellist ei leita, siis kliendile omistatud IP-aadressi kuueteiskümnendarvu formaadis. Kui ka sellist ei leita, siis ükshaaval eemaldatakse IP-aadressi lõpust üks number ja korratatakse kuni lõpuni. Kui ühtegi sobivat varianti ei leita, otsitakse faili *default*. Vt. Joonis 6
- Kui leiti sobiv, siis klientmasin laeb alla kõik vajalikud failid (tuum ja juurfailisüsteem) ning laeb need muutmällu
- Paigalduse lõppedes klientmasin taaskäivitub

```

/mybootdir/pxelinux.cfg/b8945908-d6a6-41a9-611d-74a6ab80b83d
/mybootdir/pxelinux.cfg/01-88-99-aa-bb-cc-dd
/mybootdir/pxelinux.cfg/C0A8025B
/mybootdir/pxelinux.cfg/C0A8025
/mybootdir/pxelinux.cfg/C0A802
/mybootdir/pxelinux.cfg/C0A80
/mybootdir/pxelinux.cfg/C0A8
/mybootdir/pxelinux.cfg/C0A
/mybootdir/pxelinux.cfg/C0
/mybootdir/pxelinux.cfg/C
/mybootdir/pxelinux.cfg/default

```

Joonis 6: Sätetefaili otsingu ülesehitus

3.4 Tagasiside küsimustik

Loodud praktikumimaterjali sobivuse ning kvaliteedi hindamise tarbeks tehti tagasiside küsimustik (vt Lisa 2). Küsimustiku koostamisel kasutati nii valikvastustega küsimusi kui ka avatud küsimusi. Avatud küsimuste kasutamine oli oluline, et saada tudengitelt vabas vormis põhjendusi. Kokku koosnes küsimustik 10-st küsimusest.

Küstiti näiteks tudengitelt varasema kogemuse kohta operatsioonisüsteemide haldamise kohta, et saada teavet teiste hinnangute kohta – kas materjal oli jõukohane ja arusaadav. Lisaks sellele, kas leidub seoseid materjali läbimiseks kulunud aja ning varasema kogemuse suhtes.

Sooviti näha kas loodud materjal on tudengitele huvitav ning kas sellest sõltub ka läbimise aeg.

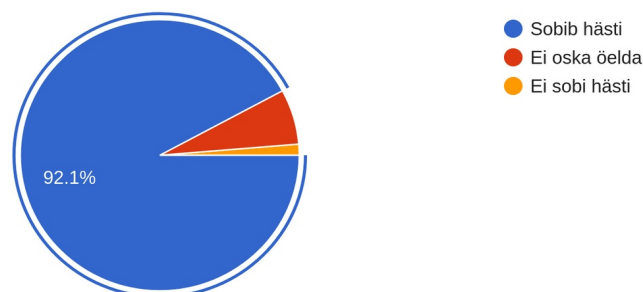
4 Analüüs

Selles peatükis antakse ülevaade tagasiside küsitluse tulemustest ning tehtud järeldustest.

4.1 Tagasiside küsitlus

Valminud praktikumimaterjali anti testida TalTech IT Kolledži tudengitele, kellel oli käsil õppeaine Operatsioonisüsteemid ja nende haldamine. Tagasisideküsimustikule vastas 76 õpilast, kuid reaalne läbinute arv võis olla suurem. Tagasiside eesmärk oli eelkõige välja selgitada, kas loodud materjal on õppeaine jaoks sobilik, arusaadav ning jõukohane.

1. Kuidas loodud materjal sobib õppeainega "Operatsioonisüsteemid ja nende haldamine" ?
76 responses



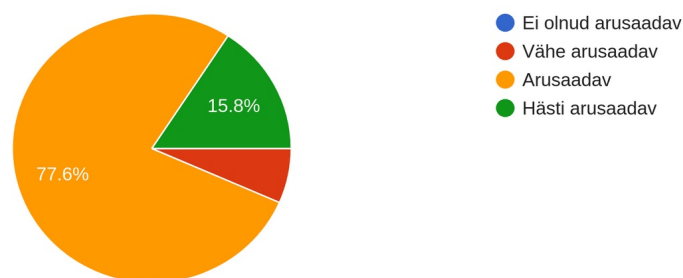
Joonis 7: Materjali sobilikkus

Valdav enamus tudengitest leidis, et materjal on sobilik antud õppeainesse. Ainult 1 tudeng arvas, et ei ole sobilik, põhjendades, et sobiks rohkem Linuxi administreerimise ainesse. Sellest võib järeldada, et praktikumimaterjalis käsitletud teema valik oli õnnestunud.

71 tudengit leidis, et materjal on kas hästi arusaadav või arusaadav. Vähe arusaadavaks pidas materjali ainult 5 vastanut.

6. Kuidas hindate materjali arusaadavust ?

76 responses



Joonis 8: Materjali arusaadavus

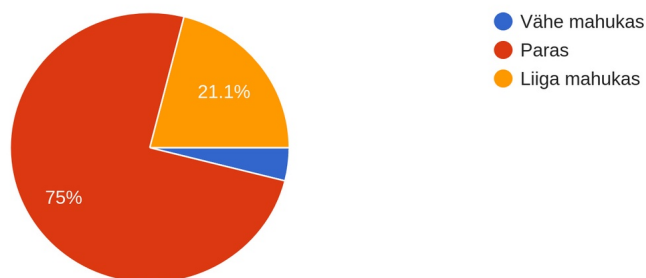
67 vastanut arvas, et tegemist on jõukohase materjaliga ning 9 vastanut leidis, et materjal on liiga keeruline. Liiga lihtsaks ei pidanud materjali mitte keegi. Võib järeldada, et loodud materjali sisu pigem ei olnud tudengite jaoks liiga keeruline ning oli üldjoontes esitatud piisavalt detailselt. Seda toetab ka fakt, et valdaval enamusel vastanutest oli esimene kokkupuude Linuxiga just Operatsioonisüsteemid ja nende haldamine aine raames. Kui materjal testimiseks anti, olid tudengid juba enamuse kursusest läbitöötanud. Selle põhjal võib väita, et pigem sobib see praktikumimaterjal läbimiseks kursuse lõpupoole. Sest siis on tudengitel juba piisavalt kogemusi käsureaga ning ka virtualiseerimistarkvaraga.

Lisaks uuriti veel materjali mahukust ning paluti läbijatel ka mõõta aega, mis kulus selle läbimiseks.

57 vastanut leidis, et tegemist on paraja materjali mahuga ning 16 vastanut arvas, et see on liiga mahukas. 3 vastanut pidas materjali liiga väheseks.

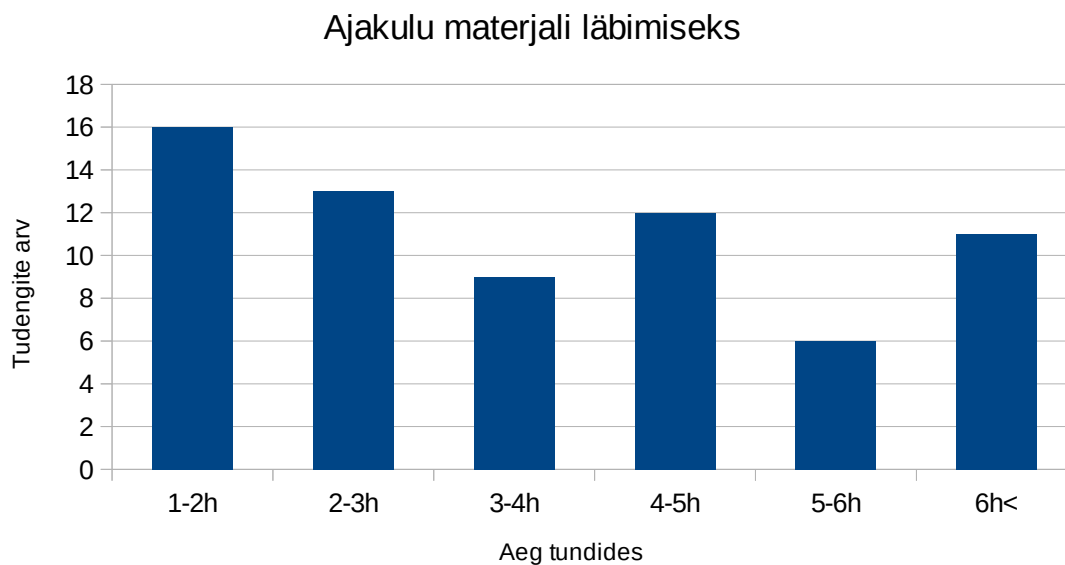
3. Kas praktikumimaterjal oli:

76 responses



Joonis 9: Materjali mahukuse hinnang

Materjali läbimiseks mõõtis aega 67 vastanut. Joonisel 10 on näha, et kõige rohkem tudengeid läbis materjali 1-2h tunniga. Sellest võib järeldada, et kõik konfiguratsiooni failide sisud ning käsud tõenäoliselt kopeeriti ning ei süvenetud piisavalt. Samas on näha, et enamusel kulus siiski kauem kui 3 tundi. Kuna tavaliselt on praktikumi pikkuseks 90 minutit, siis saab väita, et ühest praktikumist ei piisa selle materjali läbimiseks. Kuigi enamus tudengeid ei pidanud materjali liiga mahukaks, võib kulutatud aega vaadates väita, et materjal oleks mõistlik jagada vähemalt kaheks eraldi praktikumiks.



Joonis 10: Ajakulu materjali läbimiseks

Tagasisides paluti vastanutel jagada mõtteid ja ideid, kuidas loodud materjali saaks paremaks muuta. Kõige rohkem sooviti, et tegevused oleks rohkem ekraanitõmmistega toetatud. Siit võib järeldada kas materjali tekstiline osa oli liiga kehvasti esitatud või jäi tudengite funktsionaalne lugemisoskus pisut nõrgaks. Lisaks sellele viidati asjaolule, et materjalis esineb mitmeid kohti, kus valesti tegemisel hakkavad esinema veateated. Sooviti, et oleks ära toodud ka põhilised vead, mis võivad tekkida.

4.2 Materjali vastavus lähtetingimustele

Allolevas tabelis on ära toodud, millised lähtetingimused said täidetud ning millistes esines puudujääke.

Tabel 1: Vastavus lähtetingimustele

Lähtetingimus	Põhjendus
Materjali loomisel arvestatakse erinevate IT-teadmiste tasemega tudengitega ehk seletatakse samm-sammult tehtavad toimingud	Üldiselt täidetud - Toimingud on materjalis kirjeldatud käsu haaval ning käsu tegevus ning oodatav tulemus on üldiselt seletatud. Praktikumimaterjali teises osas on seda nõuet osaliselt eiratud.

Tudeng saab kontrollida sooritatud sammu õigsust	Üldiselt täidetud - Enamus käskude puhul on ära toodud, milline peab olema selle käsu väljund. Väljundid on kuvatud kas teksti või pildi formaadis.
Tegevused saab sooritada VirtualBox virtuaalmasinas	Täidetud - Terve materjal on ülesehitatud virtuaalmasinate eripära silmas pidades.
Luuakse automaatselt paigalduv Ubuntu Linux tööjaam koos seletavate juhistega	Täidetud - Praktikumimaterjali esimese osa käsitleb, kuidas ning milliste vahenditega saab luua Ubuntu MATE tööjaama ISO-faili, millelt saab teostada automaatpaigaldust.
Luuakse Ubuntu Linux'i tööjaama automaatse paigalduse server koos seletavate juhistega	Täidetud - Praktikumimaterjali teine osa käsitleb, kuidas ja milliste vahenditega saab luua PXE-serverit ning selle abil teostada automaatpaigaldust kasutades võrguliideselt alglaadimist.
Ülesanded on eraldiseisvad, see tähendab praktikumi saab läbida ka ilma automaatse paigalduse serverita	Täidetud - Kui PXE-serverit ei looda, valmib praktikumi tulemusena siiski võimalus luua automaatselt paigalduvat Ubuntu Linux tööjaama.
Kasutatav tarkvara on vabavara, et ei oleks takistusi tarkvara paigaldamisel ja kasutamisel	Täidetud - Materjali loomiseks kasutatavad tööriistad ning tehnoloogiad on kõik vabavaralised ning seega ei sea piiranguid paigaldamiseks ega kasutamiseks.

4.3 Tulevikuarendused

Esmalt tuleb viia sisse muudatused, et terve materjal vastaks täielikult lähtetingimustele. Lisaks sellele tuleks ka materjali muuta detailsemaks, mis omakorda loob võimalused ka muuta hetkel loodud õpetav praktikum kontrollitavaks praktikumiks. Lisaks saaks tekitada võimaluse luua mitme erineva Linux'i distributsiooni profiilid PXE-serverisse või lasta tudengitel ise luua vastusefail, mis vastab ülesandes seatud nõuetele.

Kuna selgus, et osadel tudengitel oli põhisüsteemina kasutusel MS Windows, siis ei saanud praktikumimaterjali läbida selliselt nagu kirjas. Selle tarbeks tuleks materjalis käsitleda ka võimalused, kuidas MS Windowsi põhisüsteemi kasutades saaks luua Ubuntu Linux'i automaatpaigalduse ISO-faili.

Materjali koostamise lõpuks tuli välja ka Ubuntu versioon 20.04. Mõnel tudengil õnnestus proovida ka sellega praktikumi läbida, kuid selgus, et PXE-serverit tööle seal

ei saanud. Seega tuleks uurida ja testida, miks ei toiminud ning mida peab materjalis muutma.

Hetkel on materjal ülesehitatud kasutama BIOS tüüpi püsivara (*firmware*), sest õppeaines virtuaalmasinaid luues sellele suurt rõhku pole pandud. Kuid, kuna reaalses maailmas on pigem levinud UEFI tüüpi püsivara, tuleb kindlasti materjalile lisada eraldi osa, mis katab ka paigalduse seadistamise sellisel juhul [20].

Kuigi hetkel kasutab õppeaine pigem Ubuntu Linuxit, siis tulevikus võib ka rohkem Windowsiga seotud tegevusi sinna lisanduda. Sellisel juhul oleks võimalik ka luua analoogne praktikumimaterjal, mis käsitleks MS Windowsi operatsioonisüsteemiga tööjaama automaatpaigaldust [21].

5 Kokkuvõte

Valminud diplomitöö eesmärk oli koostada tööjaamale operatsioonisüsteemi paigaldamise automatiseerimist tutvustav praktikumimaterjal. Töö tulemusena valmis kaheosaline praktikumimaterjal „Ubuntu Linuxi tööjaama paigaldamise automatiseerimine” kursuse Operatsioonisüsteemid ja nende haldamine tarbeks.

Diplomitöö koosneb teoreetilisest ja praktilisest osast. Teoreetilises osas viidi läbi empiiriline uuring, milliseid tarkvarasid kasutada, anti ülevaade kasutatud tehnoloogiatest ning nõuetest, millele materjal peab vastama. Praktilises osas loodi praktikumimaterjal, mis realiseeris teoreetilises osas kirjeldatud tehnoloogiad. Materjal jagunes kaheks teemaks: Ubuntu Linuxi tööjaama automaatpaigaldus kohandatud ISO-failiga ning PXE-serveri loomine ja selle abil Ubuntu Linuxi tööjaama automaatpaigaldus.

Tulevikus saab loodud materjali parandada tuginedes tudengitelt saadud tagasisidele. Lisaks sellele saab laiendada ka materjali Windows operatsioonisüsteemi paigaldamise automatiseerimiseks. Töö käigus selgus ka, et tulevikus võiks loodud materjali jagada eraldi praktikumideks, sest mõlemad teemad osutusid suuremaks, kui esmapilgul tundus.

Kasutatud kirjandus

- [1] IAAB17/19 - IT süsteemide administreerimine. [WWW] https://ois.ttu.ee/portal/page?_pageid=37,674560&_dad=portal&_schema=PORTAL&p_action=view&p_fk_str_yksus_id=50001&p_kava_version_id=50494&p_net=internet&p_lang=ET&p_rezhiim=0&p_mode=1&p_from (28.04.2020)
- [2] VirtualBox. [WWW] <https://www.virtualbox.org> (10.05.2020)
- [3] FAI – Fully Automatic Installation. [WWW] <https://fai-project.org> (15.03.2020)
- [4] Cobbler. [WWW] <https://cobbler.github.io> (03.05.2020)
- [5] B.1.Introduction. Appendix B. Automating the installation using preseeding. [WWW] <https://help.ubuntu.com/lts/installation-guide/amd64/apbs01.html> (28.04.2020)
- [6] Preboot Execution Environment. [WWW] https://en.wikipedia.org/wiki/Preboot_Execution_Environment (28.04.2020)
- [7] THE TFTP PROTOCOL (REVISION 2). [WWW] <https://tools.ietf.org/html/rfc1350> (28.04.2020)
- [8] What is DHCP?. [WWW] <https://www.efficientip.com/what-is-dhcp-and-why-is-it-important> (28.04.2020)
- [9] Network File System. [WWW] <https://searchenterprisedesktop.techtarget.com/definition/Network-File-System> (28.04.2020)
- [10] B.2.Using preseeding. [WWW] <https://help.ubuntu.com/lts/installation-guide/amd64/apbs02.html> (22.04.2020)
- [11] UbiquityAutomation. [WWW] <https://wiki.ubuntu.com/UbiquityAutomation> (15.05.2020)
- [12] GitLab. [WWW] <https://gitlab.rayark.com/open/ubuntu-preseed/blob/303574b7b14395a934c66684f4d04ba6eee9c264/ubuntu.seed> (16.05.2020)
- [13] GitHub. [WWW] <https://github.com/netson/ubuntu-unattended/blob/master/netson.seed> (16.05.2020)
- [14] Building the ISO image. [WWW] https://help.ubuntu.com/community/InstallCDCustomization#Building_the_ISO_image (10.05.2020)
- [15] Installation/SytemRequirements. [WWW] <https://help.ubuntu.com/community/Installation/SystemRequirements> (15.05.2020)
- [16] Chapter 8. VboxManage. [WWW] <https://www.virtualbox.org/manual/UserManual.html#vboxmanage> (18.05.2020)

- [17] Configuring PXE Network Boot Server on Ubuntu 18.04 LTS. [WWW]
https://linuxhint.com/pxe_boot_ubuntu_server (18.05.2020)
- [18] Filesystem Hierarchy Standard. [WWW]
https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard (28.04.2020)
- [19] Enterprise Manager Lifecycle Management Administrator's Guide. [WWW]
https://docs.oracle.com/cd/E24628_01/em.121/e27046/appdx_pxeboot.htm#EMLCM12198 (18.05.2020)
- [20] PXE-netboot-install. [WWW] <https://wiki.ubuntu.com/UEFI/PXE-netboot-install>
(18.05.2020)
- [21] How to create an unattended installation of Windows 10. [WWW]
<https://www.windowscentral.com/how-create-unattended-media-do-automated-installation-windows-10> (18.05.2020)

Lisa 1 – Praktikumimaterjal

https://docs.google.com/document/d/1EpHpKHKJb_41ycZcTViF5CVMJaAacmzPjS-jTaYGxjg/edit?usp=sharing

[Sissejuhatus](#)

[Esimene osa - Automaatpaigaldus ilma serverita](#)

[Kasutaja kataloogide muutujad](#)

[ISO-failide allalaadimine](#)

[Tööjaama ISO kohandamine](#)

[Teine osa - Automaatpaigaldus koos PXE serveriga](#)

[PXE Serveri seadistamine](#)

[DHCP, TFTP](#)

[Kokkuvõte](#)

[Tulevikus täiendamiseks](#)

Sissejuhatus

Valminud praktikumimaterjal on praktiline osa lõputööst “Praktikumimaterjal: Tööjaama paigaldamise automatiseerimine Ubuntu Linuxi näitel”. Töö autoriks on Marko Esna ning juhendajaks Edmund Laugasson. Diplomitöö on valminud TalTech IT Kolledžis.

Praktikumi läbides palun jälgida, kui kaua kulub selle läbimiseks ning hiljem tagasisideküsimustikus see kirja panna. Küsimustiku leiab järgmiselt lingilt:

<https://forms.gle/T9DCRQ9gFSxbz59z7>

Käesolev praktikumimaterjal tutvustab võimalusi, kuidas osaliselt automatiseerida Ubuntu Linuxi tööjaama paigaldamist. Praktikumi eesmärk on eelkõige anda tudengitele esmane kogemus automatiseeritud protsessi loomiseks ning tekitada huvi automatiseerimise osas.

Praktikumi teoreetiline osa:

https://docs.google.com/document/d/1NlqqUKg9_adIJcwp7Oonv-JC02xOVM8uw3adypufcY/edit?usp=sharing

Praktikum koosneb kahest osast. Esimeses osas luuakse kohandatud ISO fail Ubuntu MATE tööjaama versiooni automatiseeritud paigaldamise jaoks. Teises osas luuakse Ubuntu Server ning kohandatakse see PXE serveriks, et läbi selle teostada Ubuntu MATE tööjaama automatiseeritud paigaldus kasutades võrguliideselt alglaadimist. Paigaldused viiakse läbi kasutades virtuaalmasinaid. Praktikum on loodud eeldusega, et hostarvuti operatsioonisüsteemiks on Ubuntu Linux.

Esimene osa - Automaatpaigaldus ilma serverita

Kasutaja kataloogide muutujad

Kasutaja kataloogide muutujad on kirjas `~/.config/user-dirs.dirs` ja nende poole saab pöörduda käsu `xdg-user-dir` abil (asub tarkvarapaketi `xdg-user-dirs`), täiendav teave `man xdg-user-dir`:

This command expects the name of an XDG user dir as argument.
The possible names are:

- DESKTOP
- DOWNLOAD
- TEMPLATES
- PUBLICSHARE
- DOCUMENTS
- MUSIC
- PICTURES
- VIDEOS

Kui me soovime sõltumata kasutaja poolt valitud keelest kirjutada alati toimivad käsud, mis sisaldavad kataloogide nimesid siis tuleb kasutada kasutaja kataloogide muutujaid:

```
xdg-user-dir DESKTOP      #töölaud
xdg-user-dir DOWNLOAD    #allalaadimised
xdg-user-dir TEMPLATES   #mallid
xdg-user-dir PUBLICSHARE #avalik
xdg-user-dir DOCUMENTS   #dokumendid
xdg-user-dir MUSIC       #muusika
xdg-user-dir PICTURES    #pildid
xdg-user-dir VIDEOS      #videod
```

Neid väärtusi saab kasutada järgmiselt (valida üks neist), näitena kuvame faili sisu:

```
cat $(xdg-user-dir DOWNLOAD)/SHA256SUM
```

või ka

```
ls -la `xdg-user-dir DOWNLOAD`/SHA256SUM
```

Tegemist on sisuliselt absoluutse aadressiga kuna sisaldab endas muutujaid alates juurkataloogist.

ISO-failide allalaadimine

Luu failide allalaadimiseks kaust:

```
mkdir -v $(xdg-user-dir DOWNLOAD)/ISO
```

Väljund:

```
mkdir: created directory
```

Laadida alla nii tööjaama kui serveri tarbeks vajalikud ISO-failid:

Ubuntu 18.04 LTS

Tööjaam:

```
wget -q --show-progress -O $(xdg-user-dir DOWNLOAD)/ISO/ubuntu-  
mate-18.04.4-desktop-amd64.iso http://cdimage.ubuntu.com/ubuntu-  
mate/releases/18.04/release/ubuntu-mate-18.04.4-desktop-  
amd64.iso
```

Väljund:

```
|u-mate-18.04.4-desk  2%[          _ ] 45,59M  3,82MB/s
```

Kontrollsumma:

```
wget -q --show-progress -O $(xdg-user-dir  
DOWNLOAD)/ISO/SHA256SUMS_ubuntu-1804_desktop  
http://cdimage.ubuntu.com/ubuntu-mate/releases/18.04/release/SHA  
256SUMS
```

Kontrollsumma kontrollimine:

```
cd $(xdg-user-dir DOWNLOAD)/ISO/ && sha256sum -c  
SHA256SUMS_ubuntu-1804_desktop --ignore-missing
```

Väljund:

```
ubuntu-mate-18.04.4-desktop-amd64.iso: OK
```

Server:

```
wget -q --show-progress -O $(xdg-user-dir DOWNLOAD)/ISO/ubuntu-  
18.04.4-server-amd64.iso  
http://cdimage.ubuntu.com/releases/18.04.4/release/ubuntu-  
18.04.4-server-amd64.iso
```

Kontrollsumma:

```
wget -O $(xdg-user-dir DOWNLOAD)/ISO/SHA256SUMS_ubuntu-  
1804_server -q --show-progress  
http://cdimage.ubuntu.com/releases/18.04.4/release/SHA256SUMS
```

kontrollsumma kontrollimine:

```
cd $(xdg-user-dir DOWNLOAD)/ISO/ && sha256sum -c  
SHA256SUMS_ubuntu-1804_server --ignore-missing
```

Serveri ISO allalaadimisel ja kontrollsumma kontrollimisel väljundid analoogsed tööjaama omadega.

Tööjaama ISO kohandamine

Kõik toimingud tehakse käsurealt. Valmiv ISO on mõeldud BIOS tüüpi seadmetel kasutamiseks.

Navigeerime kasutaja töölauale ning loome kataloogi, kuhu lahti pakkida tööjaama ISO fail

```
mkdir -v $(xdg-user-dir DESKTOP)/Workstation_ISO
```

Väljund:

```
mkdir: created directory "loodud kausta absoluutne aadress"
```

Pakime lahti ISO faili sisu eelnevalt loodud kataloogi, kasutades MATE töölauakeskkonna arhiivihaldurit. Teiste Ubuntu distributsioonide puhul võib kasutada neile omaseid arhiivihaldureid. Ubuntu MATE puhul:

```
engrampa -e $(xdg-user-dir DESKTOP)/Workstation_ISO $(xdg-user-dir DOWNLOAD)/ISO/ubuntu-mate-18.04.4-desktop-amd64.iso
```

Väljund: käsuga **ls -la Workstation_ISO** näeme, et ISO failid on lahti pakitud.

```
marko@marko-Legion-Y540-17IRH:~/Desktop$ ls -la Workstation_ISO/
total 88
drwxrwxr-x 13 marko marko 4096 apr 23 15:17 .
drwxr-xr-x 10 marko marko 4096 apr 23 15:17 ..
drwx----- 3 marko marko 4096 veebr 3 20:57 boot
drwx----- 2 marko marko 4096 apr 23 15:17 '[BOOT]'
```

```
drwx----- 2 marko marko 4096 veebr 3 20:57 casper
drwx----- 2 marko marko 4096 veebr 3 20:57 .disk
drwx----- 3 marko marko 4096 veebr 3 20:55 dists
drwx----- 3 marko marko 4096 veebr 3 20:57 EFI
drwx----- 2 marko marko 4096 veebr 3 20:57 install
drwx----- 2 marko marko 12288 veebr 3 20:57 isolinux
-rw-rw-r-- 1 marko marko 24556 veebr 3 20:57 md5sum.txt
drwx----- 2 marko marko 4096 veebr 3 20:55 pics
drwx----- 5 marko marko 4096 veebr 3 20:55 pool
drwx----- 2 marko marko 4096 veebr 3 20:55 preseed
-rw-rw-r-- 1 marko marko 238 veebr 3 20:55 README.diskdefines
marko@marko-Legion-Y540-17IRH:~/Desktop$ █
```

Järgmiseks muudame alglaadimise parameetreid, et oleks võimalik automaatselt paigaldust alustada. Kasutame selle jaoks näiteks tekstiredaktorit **nano**.

```
nano $(xdg-user-dir DESKTOP)/Workstation_ISO/isolinux/txt.cfg
```

Seejärel avaneb järgmine vaatepilt:

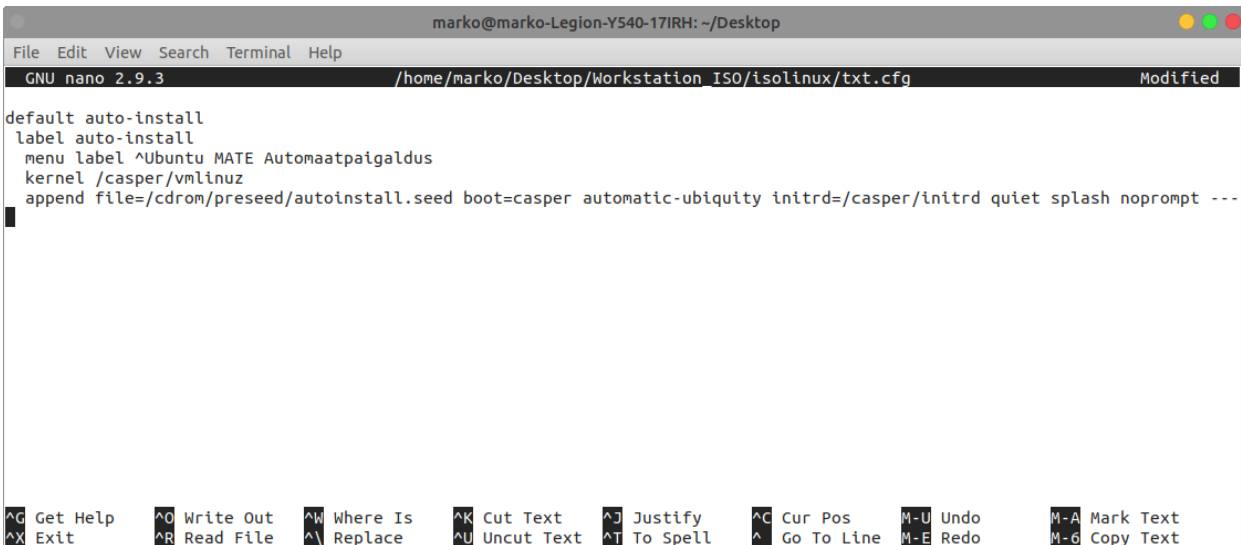
```
marko@marko-Legion-Y540-17IRH: ~/Desktop
File Edit View Search Terminal Help
GNU nano 2.9.3 /home/marko/Desktop/Workstation_ISO/isolinux/txt.cfg

default live
label live
  menu label ^Try Ubuntu MATE without installing
  kernel /casper/vmlinuz
  append file=/cdrom/preseed/ubuntu-mate.seed boot=casper initrd=/casper/initrd quiet splash $
label live-install
  menu label ^Install Ubuntu MATE
  kernel /casper/vmlinuz
  append file=/cdrom/preseed/ubuntu-mate.seed boot=casper only-ubiquity initrd=/casper/initrd$
label check
  menu label ^Check disc for defects
  kernel /casper/vmlinuz
  append boot=casper integrity-check initrd=/casper/initrd quiet splash ---
label memtest
  menu label Test ^memory
  kernel /install/mt86plus
label hd
  menu label ^Boot from first hard disk
  localboot 0x80

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell   ^_ Go To Line
```


Tegemist on standard menüüga, mida kuvatakse kasutajale tavaliselt siis, kui üritatakse näiteks USB pulgalt teostada algladimist. Kuna automaatpaigalduseks seda tarvis pole, siis võime kogu faili sisu ära kustutada ning seejärel luua uue valiku:

```
default auto-install  
  
label auto-install  
  
menu label ^Ubuntu MATE Automaatpaigaldus  
  
kernel /casper/vmlinuz  
  
append file=/cdrom/preseed/autoinstall.seed boot=casper  
automatic-ubiquity initrd=/casper/initrd quiet splash noprompt  
--
```



```
marko@marko-Legion-Y540-17IRH: ~/Desktop  
File Edit View Search Terminal Help  
GNU nano 2.9.3 /home/marko/Desktop/Workstation_ISO/isolinux/txt.cfg Modified  
default auto-install  
label auto-install  
menu label ^Ubuntu MATE Automaatpaigaldus  
kernel /casper/vmlinuz  
append file=/cdrom/preseed/autoinstall.seed boot=casper automatic-ubiquity initrd=/casper/initrd quiet splash noprompt ---  
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo M-A Mark Text  
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line M-E Redo M-6 Copy Text
```

default auto-install - määrab, millist valikut vaikimisi kasutatakse

label auto-install - valiku nimetus, ei pea olema auto-install, kasutaja saab ise nimetuse anda

menu label - algladimise menüüs kuvatav valiku nimetus

kernel - näitab tuuma asukohta meediumil

append - algladimise parameetrite lisamiseks

file - näitab preseed faili asukohta (see fail luuakse järgmises juhendi osas)

boot=casper - alglaadimine live süsteemi

automatic-ubiquity - paigalduskeskkonna automaatrežiimis käivitamiseks

initrd -näitab ajutise juurfailisüsteemi asukohta

quiet splash - ei kuvata alglaadimise protsessi teateid, vaid näidatakse ainult käivituslogo

noprompt - pärast paigaldamist ei küsi meediumi eemaldamist süsteemist

Seejärel salvestame muudatused **Ctrl + O** klahvikombinatsiooniga ning kinnituse küsides vajutada **Enter** (salvestame faili sama nime all).

Järgmise sammuna loome ise tööjaama eelsätetefaili (*preseed*), milles sisalduvad vaikimisi vastused/väärtused, mida paigalduse ajal tavaliselt küsitakse.

```
nano                                                                    $(xdg-user-dir  
DESKTOP)/Workstation_ISO/preseed/autoinstall.seed
```

Sinna faili tuleb lisada kõik järgnevad käsud:

```
# TÖÖJAAM
```

```
# See eelsätetefail on loodud toetudes dokumentatsioonile:
```

```
#
```

```
https://help.ubuntu.com/lts/installation-guide/amd64/apbs04.html
```

```
# Kuna tegemist on "Töölaua" versiooniga, siis kasutan debian-  
installer käske ubiquity formaadis. Ubiquity on live-paigaldus  
keskkond.
```

```
# Kettajagude loomine
```

```
# Esimese SCSI/SATA ketta kasutamise valimine
```

```
ubiquity partman-auto/disk string /dev/sda
```

Kettajagude loomise meetodi valimine. Variandid kas "regular" , "lvm" või "crypto". Viimasel kahel juhul kasutatakse LVM kettajagude süsteemi. Viimasel juhul kasutatakse ka krüpteeringut.

ubiquity partman-auto/method string regular

Hoiatuse vältimine, kui eelnevalt on LVM kettajaod valitud kettal ning need kustutatakse.

ubiquity partman-lvm/device_remove_lvm boolean true

Hoiatuse vältimine, kui eelnevalt on RAID massiiv valitud kettal ning need kustutatakse.

ubiquity partman-md/device_remove_md boolean true

Eelnevalt määratletud kettajagude loomise "retsepti" valimine. Antud juhul luuakse eraldi / (süsteemi) ja /home kettajaod.

ubiquity partman-auto/choose_recipe select home

Kettajagude loomine toimub ilma kinnitust küsimata. Kuna tegemist BIOS tüüpi masinaga, tulevad MBR tüüpi kettajaod vaikimisi.

ubiquity partman-partitioning/confirm_write_new_label boolean true

ubiquity partman/choose_partition select finish

ubiquity partman/confirm boolean true

ubiquity partman/confirm_nooverwrite boolean true

Süsteemikeele valik, sisestuskeele valik ning klaviatuur

Operatsioonisüsteemi keel.

ubiquity debian-installer/locale string en_US.UTF-8

Kui vaja lisada ka teiste keelte tugi

*ubiquity localechooser/supported-locales multiselect en_US.UTF-8 nL_NL.UTF-8*

Klaviatuuri paigutuse valikuks eesti. Võimalike variante saab vaadata käsuga: `localectl list-x11-keymap-layouts`. Kasutada saab paigaldusel ainult ühte.

`ubiquity keyboard-configuration/layoutcode string ee`

Lülitab automaatse klaviatuuripaigutuse tuvastamise küsimuse välja.

`ubiquity console-setup/ask_detect boolean false`

Kella ja ajatsooni seadistamine

Kontroll, kas riistvara kell on seadistatud UTC järgi.

`ubiquity clock-setup/utc-auto boolean true`

`ubiquity clock-setup/utc boolean true`

Ajatsooni valik

`ubiquity time/zone string Europe/Tallinn`

Kontroll, kas kasutada NTP-d (Võrguaja protokoll) kellaaja määramiseks paigaldusel.

`ubiquity clock-setup/ntp boolean true`

Minimaalse paigalduse valik

`ubiquity ubiquity/minimal_install boolean true`

Uuenduste laadimine paigalduse ajal

`ubiquity unattended-upgrades/enable_auto_updates boolean true`

Varamute määramine

Riigi valik, mille servereid kasutatakse.

https://docs.moodle.org/dev/Table_of_locales tabelis on nimekiri lokaalidest, millest saab tuletada riigi koodi, mille serverit kasutada

`ubiquity mirror/country string EE`

Multiverse, universe ja restricted APT varamute lubamine

`ubiquity apt-setup/multiverse boolean true`

`ubiquity apt-setup/restricted boolean true`

```
ubiquity apt-setup/universe boolean true

# Kasutaja student loomine koos salasõnaga
# Kasutaja täisnimi
ubiquity passwd/user-fullname string Student
# Kasutajanimi
ubiquity passwd/username string student
# Kasutaja salasõna
ubiquity passwd/user-password password student
ubiquity passwd/user-password-again password student
# Nõrga salasõna lubamine
ubiquity user-setup/allow-password-weak boolean true

# Lisatarkvara paigaldamine
# Süsteemi uuenduste paigaldamine
# GIMP pilditöötlus tarkvara lisamine
# APTi varamu lisamine ning Byobu tarkvara paigaldamine
# Käsud tuleb eraldada semikooloniga, reavahetuse puhul kasutada
"\". # Käskude sees "\" kasutada ei saa.
ubiquity ubiquity/success_command \
in-target add-apt-repository -y ppa:byobu/ppa; \
in-target apt-get update -y; \
in-target apt-get upgrade -y; \
in-target apt-get dist-upgrade; \
in-target apt-get autoremove; \
in-target apt-get autoclean; \
in-target apt-get clean; \
in-target apt-get install -y byobu; \
in-target apt-get install -y gimp;
```

```
# Automaatne taaskäivitus pärast paigalduse lõppu.  
Väljalülitamiseks kasutada: ubiquity ubiquity/poweroff boolean  
true
```

```
ubiquity ubiquity/reboot boolean true
```

Seejärel taaskord **Ctrl + O** salvestamiseks ning kinnitamiseks **Enter**.

```
marko@marko-Legion-Y540-171RH: ~/Desktop
File Edit View Search Terminal Help
GNU nano 4.8 /home/marko/Desktop/Workstation_ISO/preseed/autoinstall.seed Modified
# TÖÖJAAM
# See eelsätetefail on loodud toetudes dokumentatsioonile:
# https://help.ubuntu.com/lts/installation-guide/amd64/apbs04.html
# Kuna tegemist on "Töölaua" versiooniga, siis kasutan debian-installer käske ubiquity formaadis. Ubiquity
# Kettajagude loomine
# Esimese SCSI/SATA ketta kasutamise valimine
ubiquity partman-auto/disk string /dev/sda
# Kettajagude loomise meetodi valimine. Variandid kas "regular", "lvm" või "crypto". Viimasel kahel juhul
ubiquity partman-auto/method string regular
# Hoiatuse vältimine, kui eelnevalt on LVM kettajaod valitud kettal ning need kustutatakse.
ubiquity partman-lvm/device_remove_lvm boolean true
# Hoiatuse vältimine, kui eelnevalt on RAID massiiv valitud kettal ning need kustutatakse.
ubiquity partman-md/device_remove_md boolean true
# Eelnevalt määratletud kettajagude loomise "retsepti" valimine. Antud juhul luuakse eraldi / (süsteemi)
ubiquity partman-auto/choose_recipe select home
# Kettajagude loomine toimub ilma kinnitust küsimata. Kuna tegemist BIOS tüüpi masinaga, tulevad MBR tüüpi
ubiquity partman-partitioning/confirm_write_new_label boolean true
ubiquity partman/choose_partition select finish
ubiquity partman/confirm boolean true
ubiquity partman/confirm_nooverwrite boolean true
# Süsteemikeele valik, sisestuskeele valik ning klaviatuur
# Operatsioonisüsteemi keel.
ubiquity debian-installer/locale string en_US.UTF-8
# Kui vaja lisada ka teiste keelte tugi
# ubiquity localechooser/supported-locales multiselect en_US.UTF-8 nl_NL.UTF-8
# Klaviatuuri paigutuse valikuks eesti. Võimalike variante saab vaadata käsuga: localectl list-x11-keymap
ubiquity keyboard-configuration/layoutcode string ee
# Lülitab automaatse klaviatuuripaigutuse tuvastamise küsimuse välja.
ubiquity console-setup/ask_detect boolean false
# Kella ja ajatsooni seadistamine
# Kontroll, kas riistvara kell on seadistatud UTC järgi.
ubiquity clock-setup/utc-auto boolean true
ubiquity clock-setup/utc boolean true
# Ajatsooni valik
ubiquity time/zone string Europe/Tallinn
# Kontroll, kas kasutada NTP-d (Võrguaja protokoll) kellaaja määramiseks paigaldusel.
ubiquity clock-setup/ntp boolean true
# Minimaalse paigalduse valik
ubiquity ubiquity/minimal_install boolean true
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^_ Go To Line M-E Redo
```

Järgmise sammuna loome uue ISO faili **Workstation_ISO** kausta sisust. ISO loomise aluseks on võetud skript, mis asub Ubuntu wikis https://help.ubuntu.com/community/InstallCDCustomization#Building_the_ISO_image

```
mkisofs -r -V "UbuntuAutomaatpaigaldus" -cache-inodes -J -l -b  
isolinux/isolinux.bin -c isolinux/boot.cat -no-emul-boot -boot-  
load-size 4 -boot-info-table -o $(xdg-user-dir DESKTOP)/ubuntu-  
desktop-automaatpaigaldus.iso $(xdg-user-dir  
DESKTOP)/Workstation_ISO
```

Väljundi viimased read:

```
Total translation table size: 2048  
Total rockridge attributes bytes: 67041  
Total directory bytes: 210944  
Path table size(bytes): 1142  
Max brk space used b9000  
959717 extents written (1874 MB)
```

Selle tulemusel peaks tekkima hetkel sisseloginud kasutaja töölauale *ubuntu-desktop-automaatpaigaldus.iso* fail. ISO testimiseks tuleks luua VirtualBoxis tühi virtuaalmasin. Kes soovib, võib selle luua käsitsi, kasutades graafilist kasutajaliidest. Kes soovib protsessi veelgi automaatsemaks muuta, saab proovida virtuaalmasina loomist **bash**i skriptiga, mis loob ning käivitab masina juba eelnevalt kohandatud ISO failiga.

Käsitsi loomisel virtuaalmasina parameetrid:

Nimi: Ubuntu MATE Automaatpaigaldus

Tüüp: Linux, Ubuntu(64bit)

Mälu: 2048MB

Kõvaketas: 25GB, dünaamiliselt suurenev

Protsessor: 2 tuuma


Videomälu: 128MB

Videokontroller: VMSVGA

Võrguliides: NAT

Eemaldatav meedia: Eelnevalt loodud ISO

Alglaadimis järjekord: Optiline seade, kõvaketas

General Name: Ubuntu Automaat Operating System: Ubuntu (64-bit)	Preview 
System Base Memory: 2048 MB Processors: 2 Boot Order: Optical, Hard Disk Acceleration: VT-x/AMD-V, Nested Paging, KVM Paravirtualization	
Display Video Memory: 128 MB Graphics Controller: VMSVGA Remote Desktop Server: Disabled Recording: Disabled	
Storage Controller: IDE IDE Secondary Master: [Optical Drive] ubuntu-desktop-automaatpaigaldus.iso (1,83 GB) Controller: SATA SATA Port 0: Ubuntu Automaat.vdi (Normal, 25,00 GB)	
Audio Host Driver: PulseAudio Controller: ICH AC97	
Network Adapter 1: Intel PRO/1000 MT Desktop (NAT)	
USB USB Controller: OHCI Device Filters: 0 (0 active)	
Shared folders None	
Description None	

Skript, mille käivitamisel luuakse uus virtuaalmasin vastavate parameetritega ning käivitatakse koos ISO tõmmisega automaatselt. Eelnevalt on parem juba VirtualBox käivitada, mõnel korral skriptiga loomise järel alglaadimine ebaõnnestus, kui VirtualBox ei olnud eelnevalt käivitatud.

```
#!/bin/bash
```

```
# MACHINENAME on muutuja, mille väärtus saadakse skripti  
käivitamisel esimesest argumendist ehk planeeritav  
virtuaalmasina nimi.
```

```
MACHINENAME=$1
```

```
# ISO_NAME on muutuja, mille väärtus saadakse skripti  
käivitamisel teisest argumendist ehk eelnevalt loodud ISO faili  
nimi.
```

```
ISO_NAME=$2
```

```
# Skript käivitatakse kahe muutujaga: ./skripti_nimi.sh "loodava  
virtuaalmasina nimi" "ISO nimi"
```

```
# Kasutatud on materjale aadressilt:
```

```
https://www.andreafortuna.org/2019/10/24/how-to-create-a-  
virtualbox-vm-from-command-line/
```

```
# Skriptis kasutatavatest käskudest saab detailsemalt lugeda  
VirtualBoxi dokumentatsioonist.
```

```
# https://www.virtualbox.org/manual/ch08.htm
```

```
# Virtuaalmasina loomine ning registreerimine VirtualBoxiga.  
Kasutades --register parameetrit ei pea käsitsi loodavat masinat
```

eraldi registreerima VirtualBoxi tarkvara jaoks, vaid see toimub loomise ajal.

Kõik erinevad --ostype variandid saab kuvada käsuga:

"VBoxManage list ostypes"

--basefolder näitab, kus kaustas hakkab loodav virtuaalmasin paiknema.

```
VBoxManage createvm --name $MACHINE_NAME --ostype "Ubuntu_64" --register --basefolder $(xdg-user-dir DESKTOP)
```

I/O APIC lubamine

```
VBoxManage modifyvm $MACHINE_NAME --ioapic on
```

Muutmälu ja videomälu määratlemine

```
VBoxManage modifyvm $MACHINE_NAME --memory 2048 --vram 128
```

Graafikakontrolleri valik. Valikuvariandid:

-graphicscontroller none|vboxvga|vmsvga|vboxsvga

```
VBoxManage modifyvm $MACHINE_NAME --graphicscontroller vmsvga
```

Võrguliidese tüübi valik. Võimalik lisada ka mitu. Formaat on

järgmine: --nic<1-N> ning tüüp: none|null|nat|natnetwork|

bridged|intnet|hostonly|generic

```
VBoxManage modifyvm $MACHINE_NAME --nic1 nat
```

Virtuaalse kõvaketta loomine. --size parameeter on

megabaitides.

--variant Standard dünaamiliselt suurenev ketas.

```
VBoxManage createmedium disk --filename $(xdg-user-dir DESKTOP)/$MACHINE_NAME/$MACHINE_NAME_DISK.vdi --size 25000 --format VDI --variant Standard
```

Kontrolleri lisamine mäluseadmete jaoks.

```
VBoxManage storagectl $MACHINE_NAME --name "SATA Controller" --  
add sata --controller IntelAhci
```

Kõvaketta külge ühendamise

```
VBoxManage storageattach $MACHINE_NAME --storagectl "SATA  
Controller" --port 0 --device 0 --type hdd --medium $(xdg-user-  
dir DESKTOP)/$MACHINE_NAME/$MACHINE_NAME_DISK.vdi
```

CD/DVD külge haakimine

```
VBoxManage storageattach $MACHINE_NAME --storagectl "SATA  
Controller" --port 1 --device 0 --type dvddrive --medium $(xdg-  
user-dir DESKTOP)/$ISO_NAME
```

Algladimisjärjekorra seadistamine

```
VBoxManage modifyvm $MACHINE_NAME --boot1 dvd --boot2 disk --  
boot3 none --boot4 none
```

Virtuaalmasina käivitamine

```
VBoxManage startvm $MACHINE_NAME
```

Skript salvestada formaadis:

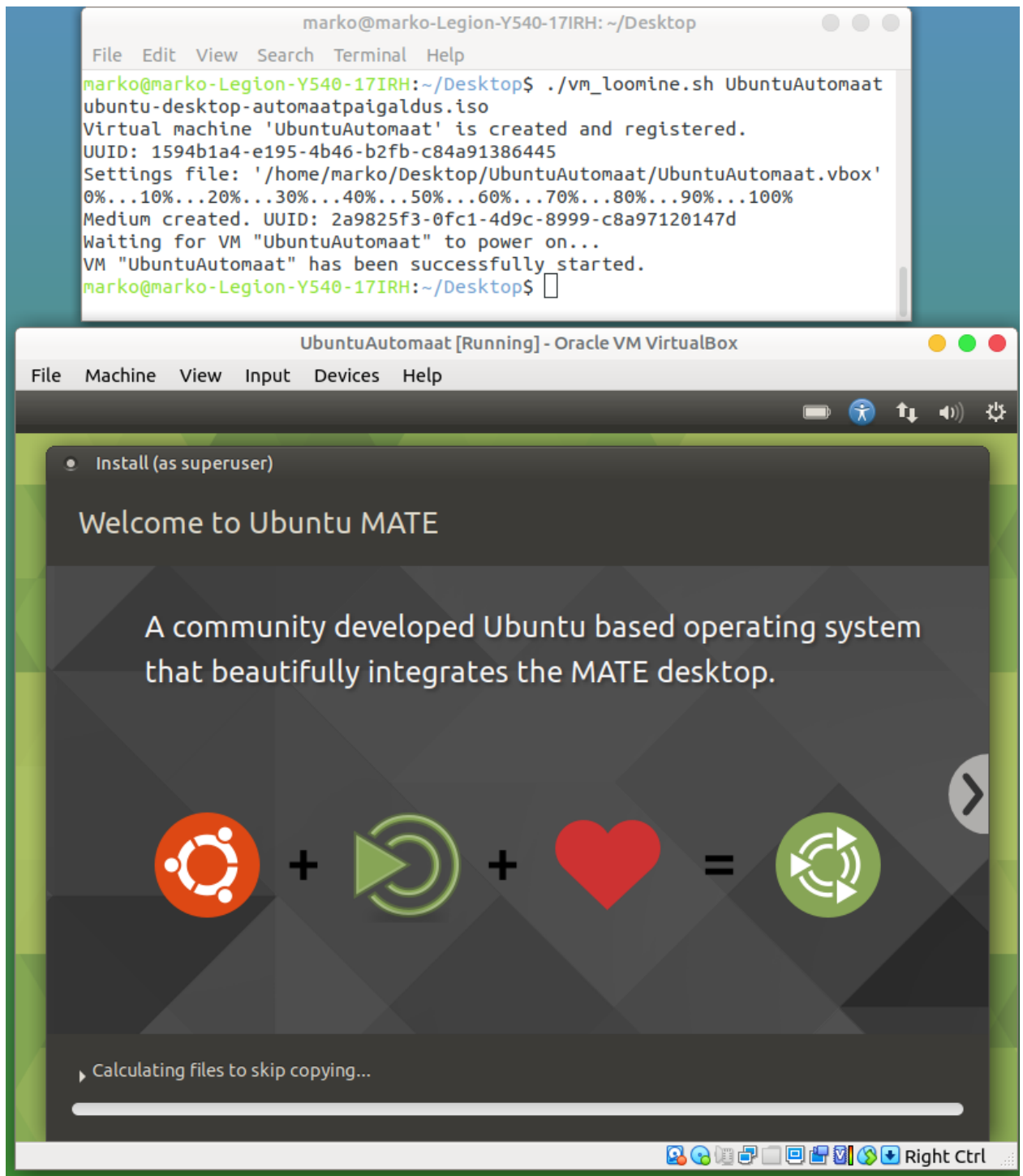
```
skripti_nimi.sh
```

Skripti muutmine käivitavaks:

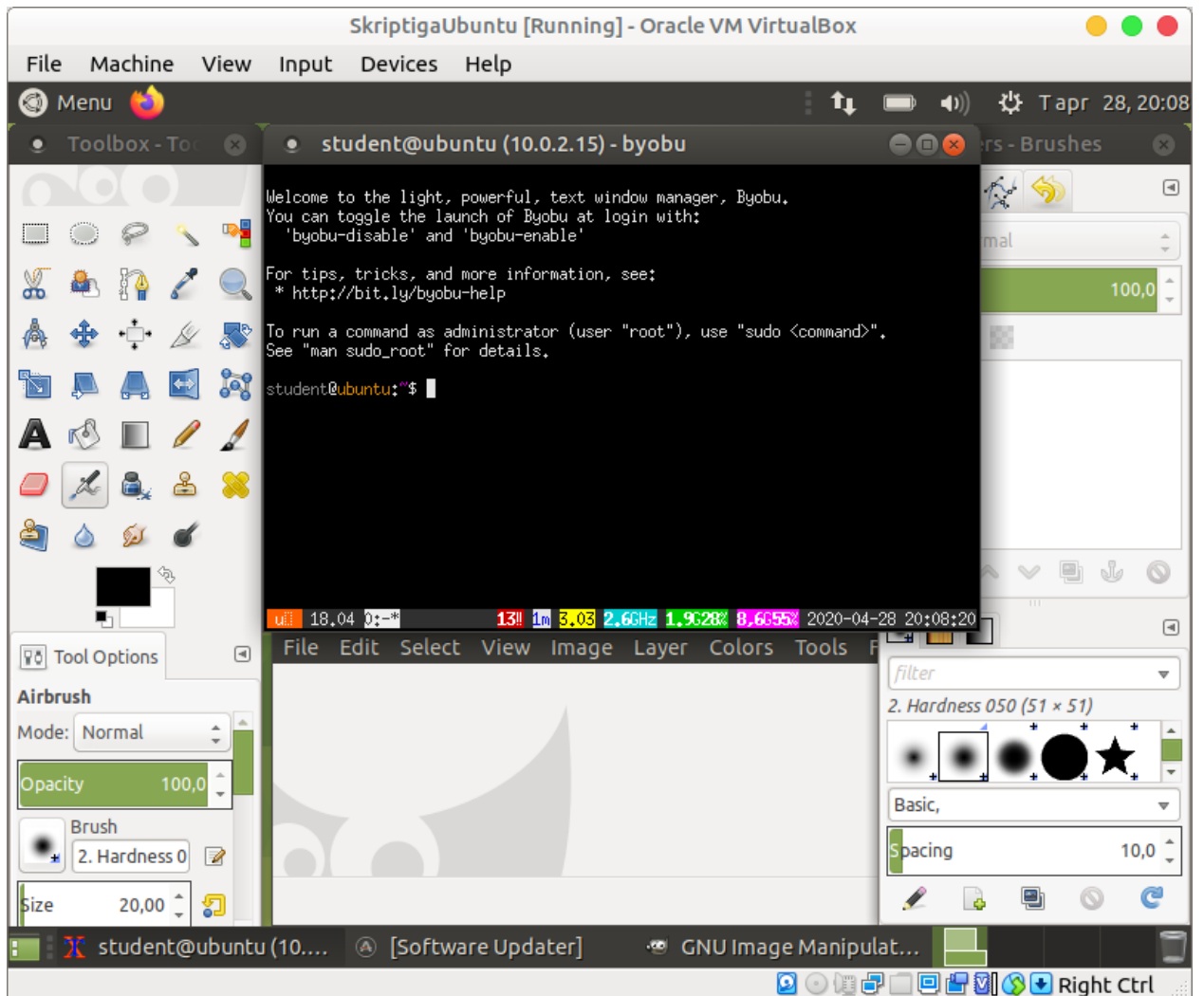
```
chmod +x skripti_nimi.sh
```

Ning seejärel käivitamine:

```
./skripti_nimi.sh VIRTUAALMASINALE_PANDAV_NIMI ISO_FAILI_NIMI
```



Pärast paigaldust saab teostada sisselogimise Student kasutaja. Ning saab veenduda ka, et eelmainitud tarkvarad on paigaldunud.



Teine osa - Automaatpaigaldus koos PXE serveriga

PXE Serveri seadistamine

Esmalt on tarvis luua virtuaalmasin Ubuntu Server 18.04.4 versiooniga. Vajadusel leiab juhendi selle kohta leiab järgnevalt lingilt:

<https://enos.itcollege.ee/~edmund/osadmin/est/praktikumid/OSadmin-ubuntu-server-install.pdf>

Masinale lisada 2 võrguliidest, millest üks on **NAT** ning teine **Internal Network**. Pärast paigaldamist võib ka uuendused paigaldada.

Pärast sisselogimist alustame sisevõrgu (*internal network*) liidesele staatilise IP andmisega. Selle tarbeks tuleks vaadata, millised liidesed süsteemis saadaval on. Käsuga:

```
ip a
```

Näeme, millised liidesed olemas on.


```
PXE_UBUNTU [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
marko@ubuntu:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:47:6d:24 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 86234sec preferred_lft 86234sec
    inet6 fe80::a00:27ff:fe47:6d24/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:b7:b5:fc brd ff:ff:ff:ff:ff:ff
marko@ubuntu:~$ sc_
```

Kuna VirtualBox kasutab NAT võrgu puhul 10.0.0.0 võrku, siis teame, et **enp0s3** on NAT liides. Seega **enp0s8** on sisevõrgu jaoks. Edasi saab sisevõrgu liidese seadistada staatilise IP-ga.

Kui kasutusel LIVE server siis :

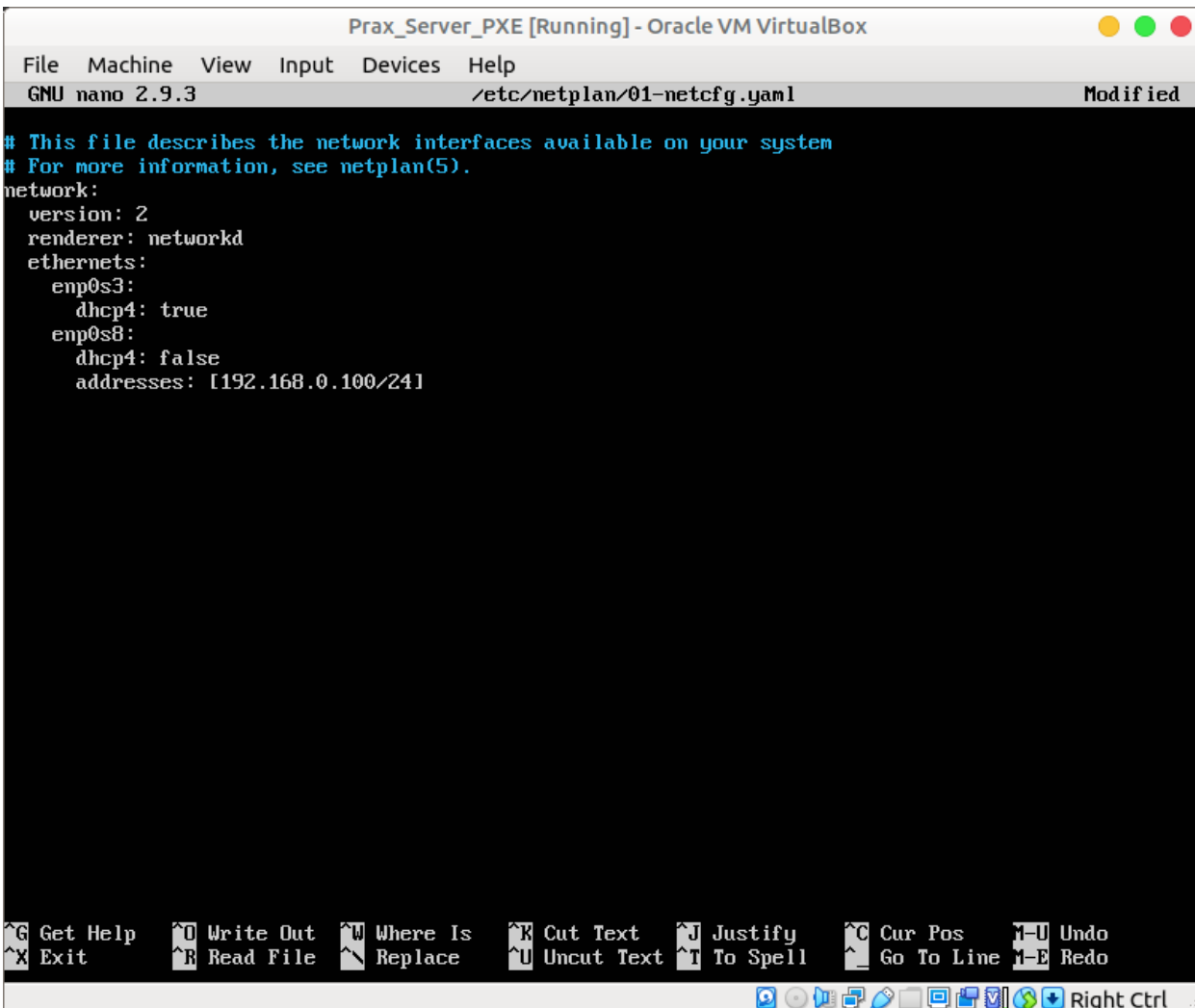
```
sudo nano /etc/netplan/50-cloud-init.yaml
```

Kui tavaline, siis:

```
sudo nano /etc/netplan/01-netcfg.yaml
```

Kui failinimi ei ole õige siis vaadata *.yaml faile kaustas /etc/netplan/

Antud praktikumi raames piisab, kui määrame ainult IP aadressi ning keelame DHCP.

A screenshot of a virtual machine window titled "Prax_Server_PXE [Running] - Oracle VM VirtualBox". The window shows a terminal running the nano text editor. The editor is editing the file "/etc/netplan/01-netcfg.yaml". The content of the file is as follows:

```
# This file describes the network interfaces available on your system
# For more information, see netplan(5).
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: false
      addresses: [192.168.0.100/24]
```

The nano editor interface includes a menu bar at the top with "File", "Machine", "View", "Input", "Devices", and "Help". Below the menu bar, it shows "GNU nano 2.9.3" and the file path. At the bottom of the terminal window, there is a status bar with various keyboard shortcuts and a "Right Ctrl" button.

Pärast salvestamist tuleb muudatused aktiveerida. Lisades lipu *--debugging* näeb vea korral kirjeldust, mis on valesti:

```
sudo netplan --debugging apply
```

Seejärel peaks liides automaatselt aktiivseks muutuma. Kui ei muutunud, siis saab võrguliidese muuta aktiivseks käsuga:

```
sudo ip link set enp0s8 up
```

DHCP, TFTP

Järgmiseks on vaja, et meie server käituks ka **DHCP** ning **TFTP** serverina. Antud praktikumi tarbeks piisab sellisest tarkvarast nagu **dnsmasq**. See on saadaval ilma täiendavaid varamuid lisamata. Käsus **-y** lipp on automaatse "yes" valimiseks, kui küsitakse paigalduse kinnitust.

```
sudo apt-get install -y dnsmasq
```

Kui paigaldus on edukalt lõppenud, tuleb muuta **dnsmasq** sätetefaili, et määrata, millist liidest pidi hakatakse **DHCP** teenust pakkuma. Samuti tuleb võimaldada ka **TFTP** serveri teenus. Konfiguratsiooni faili asukoht: **/etc/dnsmasq.conf**. **TFTP** jaoks loome ka eraldi juurkausta.

```
sudo mkdir -v /srv/tftp
```

Seejärel muudame dnsmasqi sätteid:

```
sudo nano /etc/dnsmasq.conf
```

Järgnevad sätted lisada esialgse faili lõppu ning pärast ka salvestada. Kellel sügavam huvi, siis väljakommenteeritud osades on kirjas valdav osa võimalustest, kuidas **dnsmasq-i** millegi jaoks vajadusel seadistada. Praktikumi tarbeks piisab järgnevast:

```
# Liidese valik, mille kaudu DHCP ja TFTP teenust pakutakse  
# Käsk millega näha kõiki võrguliideseid käsurealt: netstat -i |  
cut -d' ' -f1 | sed 1,2d -  
interface=enp0s8  
  
# DHCP parameetrid vastavalt järjekorrale:  
# liidese nimetus, DHCP vahemiku algus ja lõpp, võrgumask, IP  
liisingu kestvus tundides
```

```
dhcp-range=enp0s8, 192.168.0.25,192.168.0.50,255.255.255.0,24h
```

```
# TFTP teenuse aktiveerimine
```

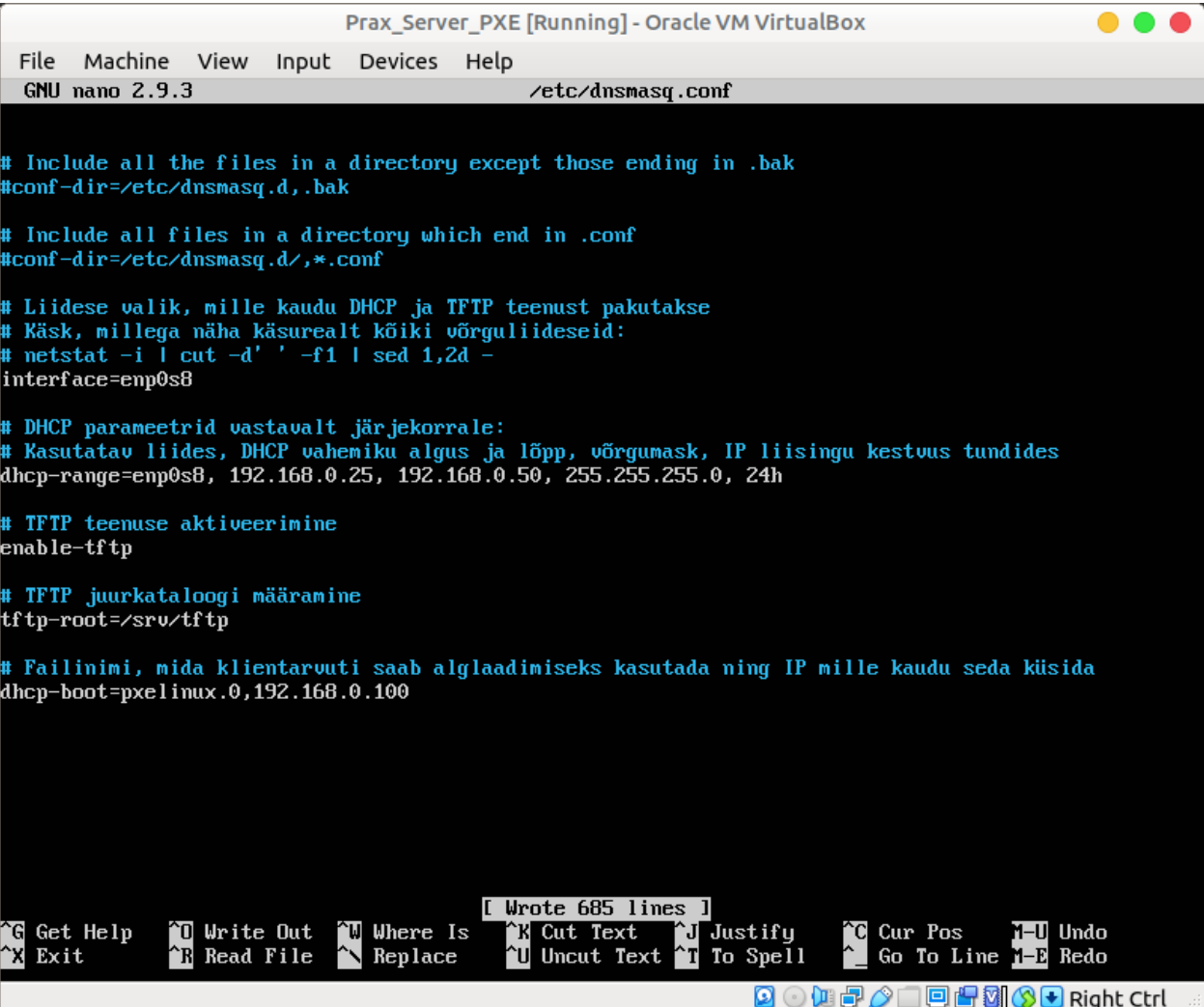
```
enable-tftp
```

```
# TFTP juurkataloogi määramine
```

```
tftp-root=/srv/tftp
```

```
# Failinimi, mida klient saab alglaadimiseks kasutada ning IP mille kaudu seda saada
```

```
dhcp-boot=pxelinux.0,192.168.0.100
```



```
Prax_Server_PXE [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
GNU nano 2.9.3 /etc/dnsmasq.conf

# Include all the files in a directory except those ending in .bak
#conf-dir=/etc/dnsmasq.d,.bak

# Include all files in a directory which end in .conf
#conf-dir=/etc/dnsmasq.d/*conf

# Liidese valik, mille kaudu DHCP ja TFTP teenust pakutakse
# Käsk, millega näha käsurealt kõiki võrguliideseid:
# netstat -i | cut -d' ' -f1 | sed 1,2d -
interface=enp0s8

# DHCP parameetrid vastavalt järjekorrale:
# Kasutatu liides, DHCP vahemiku algus ja lõpp, võrgumask, IP liisingu kestvus tundides
dhcp-range=enp0s8, 192.168.0.25, 192.168.0.50, 255.255.255.0, 24h

# TFTP teenuse aktiveerimine
enable-tftp

# TFTP juurkataloogi määramine
tftp-root=/srv/tftp

# Failinimi, mida klientarvuti saab alglaadimiseks kasutada ning IP mille kaudu seda küsida
dhcp-boot=pxelinux.0,192.168.0.100

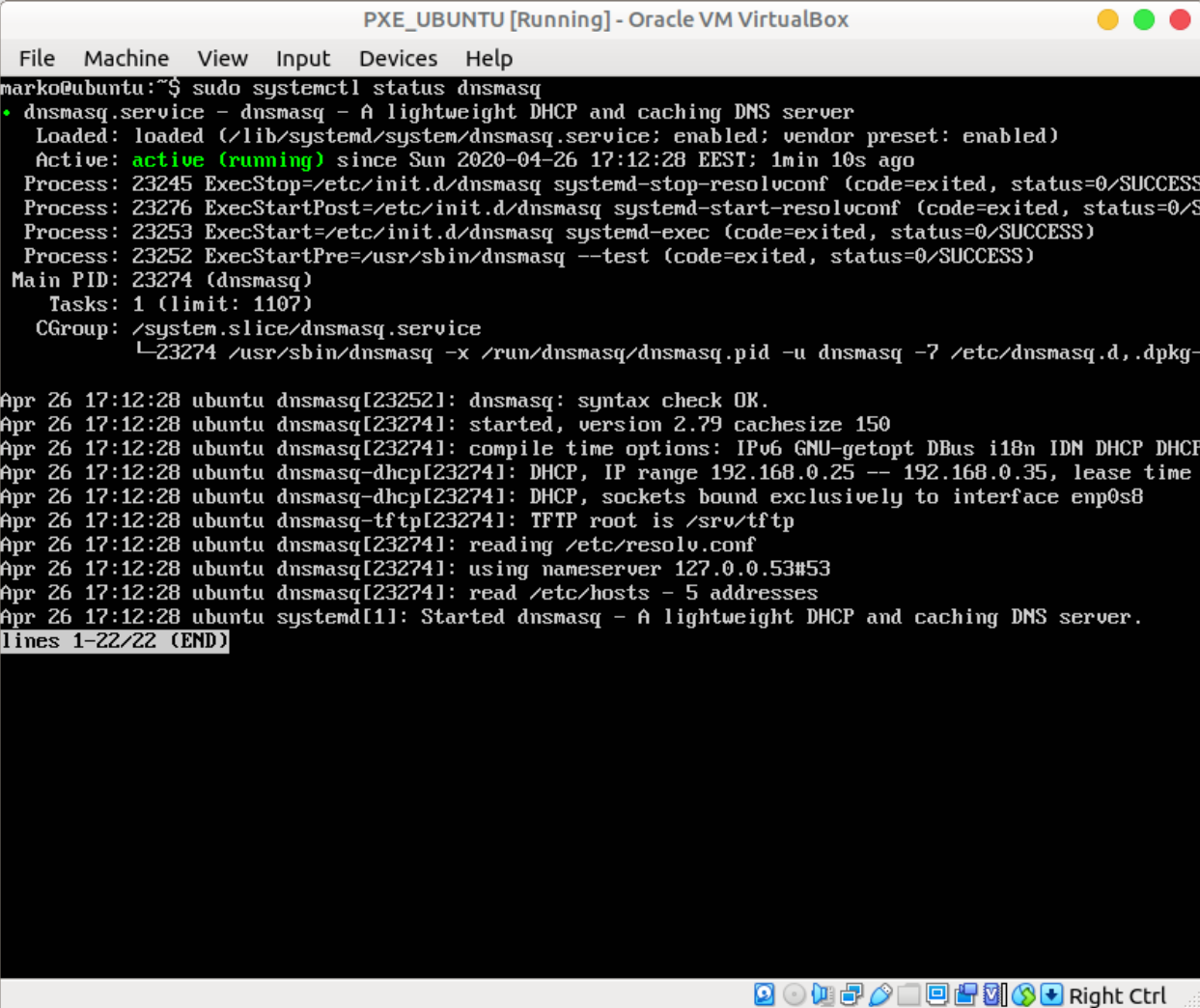
[ Wrote 685 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos ^T-U Undo
^X Exit ^R Read File ^W Replace ^U Uncut Text ^T To Spell ^_ Go To Line ^-E Redo
Right Ctrl
```

Kui muudatused on salvestatud, tuleb **dnsmasq** teenus taaskäivitada. Selle jaoks:

```
sudo systemctl restart dnsmasq
```

Ning veendumaks, et teenus töötab:

```
sudo systemctl status dnsmasq
```



```
PXE_UBUNTU [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
marko@ubuntu:~$ sudo systemctl status dnsmasq
• dnsmasq.service - dnsmasq - A lightweight DHCP and caching DNS server
  Loaded: loaded (/lib/systemd/system/dnsmasq.service; enabled; vendor preset: enabled)
  Active: active (running) since Sun 2020-04-26 17:12:28 EEST; 1min 10s ago
  Process: 23245 ExecStop=/etc/init.d/dnsmasq systemd-stop-resolvconf (code=exited, status=0/SUCCESS)
  Process: 23276 ExecStartPost=/etc/init.d/dnsmasq systemd-start-resolvconf (code=exited, status=0/SUCCESS)
  Process: 23253 ExecStart=/etc/init.d/dnsmasq systemd-exec (code=exited, status=0/SUCCESS)
  Process: 23252 ExecStartPre=/usr/sbin/dnsmasq --test (code=exited, status=0/SUCCESS)
  Main PID: 23274 (dnsmasq)
  Tasks: 1 (limit: 1107)
  CGroup: /system.slice/dnsmasq.service
          └─23274 /usr/sbin/dnsmasq -x /run/dnsmasq/dnsmasq.pid -u dnsmasq -7 /etc/dnsmasq.d,.dpkg-

Apr 26 17:12:28 ubuntu dnsmasq[23252]: dnsmasq: syntax check OK.
Apr 26 17:12:28 ubuntu dnsmasq[23274]: started, version 2.79 cachesize 150
Apr 26 17:12:28 ubuntu dnsmasq[23274]: compile time options: IPv6 GNU-getopt DBus i18n IDN DHCP DHCP
Apr 26 17:12:28 ubuntu dnsmasq-dhcp[23274]: DHCP, IP range 192.168.0.25 -- 192.168.0.35, lease time
Apr 26 17:12:28 ubuntu dnsmasq-dhcp[23274]: DHCP, sockets bound exclusively to interface emp0s8
Apr 26 17:12:28 ubuntu dnsmasq-tftp[23274]: TFTP root is /srv/tftp
Apr 26 17:12:28 ubuntu dnsmasq[23274]: reading /etc/resolv.conf
Apr 26 17:12:28 ubuntu dnsmasq[23274]: using nameserver 127.0.0.53#53
Apr 26 17:12:28 ubuntu dnsmasq[23274]: read /etc/hosts - 5 addresses
Apr 26 17:12:28 ubuntu systemd[1]: Started dnsmasq - A lightweight DHCP and caching DNS server.
lines 1-22/22 (END)
```

Sellelega on meil valmis võimalus, kuidas transportida alglaadimiseks vajalikud failid üle võrgu. Kuid lisaks sellele on vaja ka paigaldusprotsessi jaoks ülejäänud faile. Selle jaoks on tarvis **NFS-i (Network File System)** ehk võrgufailisüsteemi. See on samuti saadaval vaikimisi varamutes. Eelnevalt loome kausta, mida **NFS** hakkab kasutama paigaldusfailide hoiustamiseks:

```
sudo mkdir -v /srv/nfs
```

Paigaldamiseks:

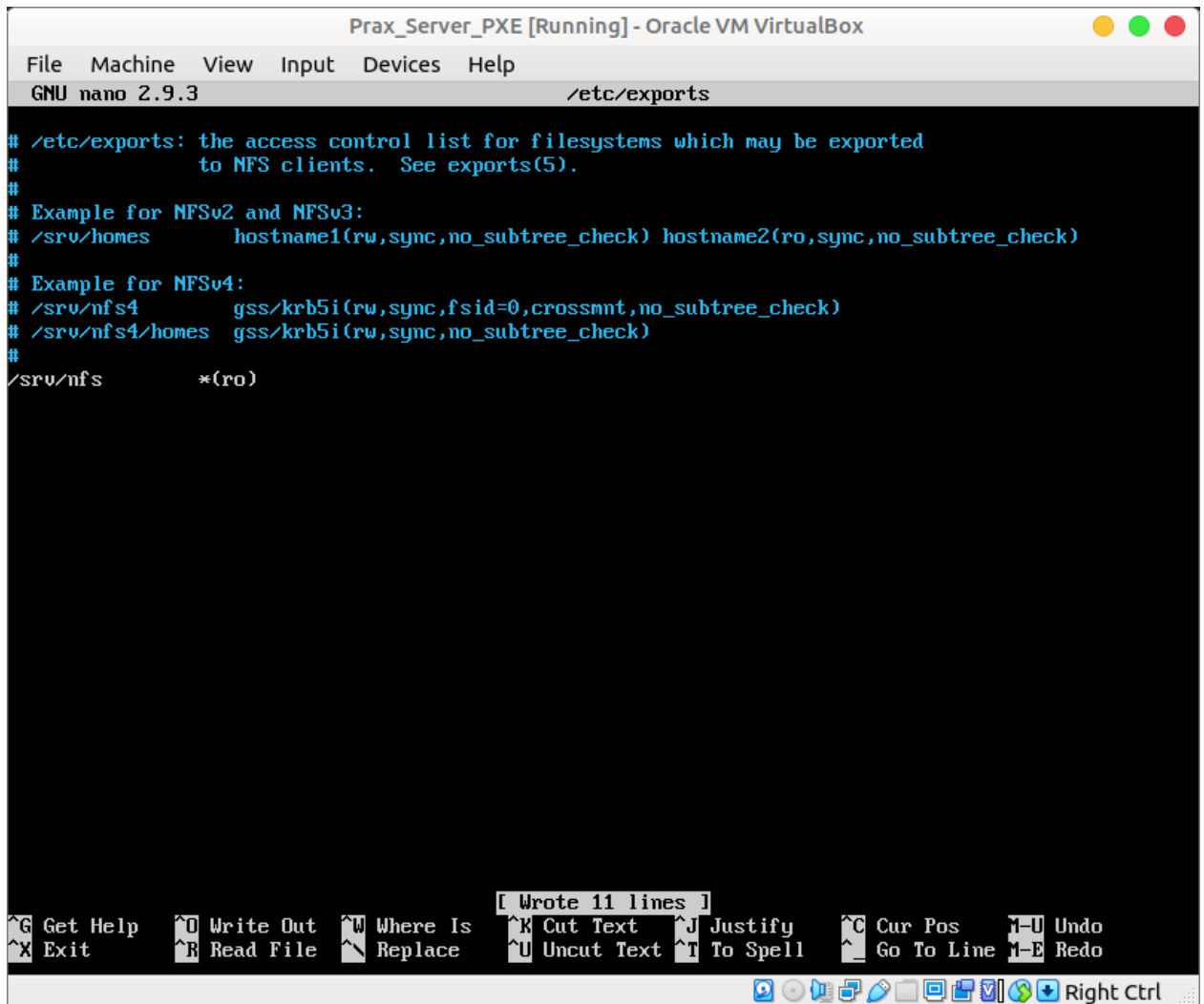
```
sudo apt update && sudo apt-get install -y nfs-kernel-server
```

Seejärel tuleb lisada eelnevalt loodud kataloog **NFS** poolt ligipääsetavate asukohtade hulka. Selle jaoks:

```
sudo nano /etc/exports
```

Sinna lisame eelnevalt loodud kataloogi absoluutse aadressiga asukohta. Lihtsuse mõttes ei piira ühendusi kindla aadressiga. Tärn (*) just seda tähendabki. Sulgudes olev **ro** tähendab, et antakse ainult lugemise õigus.

```
/srv/nfs *(ro)
```



```
Prax_Server_PXE [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
GNU nano 2.9.3 /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
# to NFS clients. See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4 gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
/srv/nfs *(ro)
```

The screenshot shows a nano editor window with the following content:

```
Prax_Server_PXE [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
GNU nano 2.9.3 /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
# to NFS clients. See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4 gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
/srv/nfs *(ro)
```

The bottom of the window shows a status bar with various keyboard shortcuts and a toolbar with icons for navigation and editing.

Pärast vajaliku rea lisamist salvestada. Lisaks salvestamisele tuleb muuta kataloog ka saadavaks. Lipp **-a** tähendab, et ühtegi kausta pole veel **NFS**-i lisatud, ehk siis kasutatakse esmakordsel lisamisel. Selle jaoks:

```
sudo exportfs -a
```

Järgmiseks on vaja paigaldada vajalikud algaaduri failid. Need saab kätte **syslinux** ja **pxelinux** pakettidest. Paigaldamiseks:

```
sudo apt-get install -y pxelinux syslinux
```

Seejärel kopeerida **/usr/lib/PXELINUX** kaustast **pxelinux.0** fail eelnevalt loodud **/srv/ftp** kausta.

```
sudo cp -v /usr/lib/PXELINUX/pxelinux.0 /srv/tftp
```

Samasse kausta tuleb kopeerida ka **/usr/lib/syslinux/modules/bios** kaustast failid: **ldlinux.c32, libcom32.c32, libutil.c32** ja **vesamenu.c32**.

```
sudo cp -v  
/usr/lib/syslinux/modules/bios/{ldlinux.c32, libcom32.c32, libutil  
.c32, vesamenu.c32} /srv/tftp
```

Edasi tuleb luua PXE alglaaduri konfiguratsiooni failide kaust ning sinna luua vaikimisi kasutatav seadistusfail. Selle jaoks:

```
sudo mkdir -v /srv/tftp/pxelinux.cfg  
sudo touch /srv/tftp/pxelinux.cfg/default
```

Lisaks on vaja alla laadida Ubuntu MATE 18.04.4 ISO, et saada vajalikud paigaldus failid nii **NFS**-i kausta kui ka **TFTP** kausta.

Allalaadimiseks vt [peatükki ISO-allalaadimine](#)

ISO-faili külgeühendamiseks:

```
sudo mkdir -v /mnt/ubuntu_iso
```

```
sudo mount -v ubuntu-mate-18.04.4-desktop-amd64.iso  
/mnt/ubuntu_iso
```

Järgnevalt luua 2 kausta, vastavalt **/srv/tftp/ubuntu_mate** ning **/srv/nfs/ubuntu_mate**.

```
sudo mkdir -v /srv/{tftp,nfs}/ubuntu_mate
```


Seejärel kopeerida rekursiivselt kogu **/mnt/ubuntu_iso** sisu **/srv/nfs/ubuntu_mate** kausta.

```
sudo cp -Rfv /mnt/ubuntu_iso/* /srv/nfs/ubuntu_mate
```

Lisaks sellele kopeerida **/srv/nfs/ubuntu_mate/casper** kaustast **vmlinuz** ja **initrd** failid **/srv/tftp/ubuntu_mate** kausta.

```
sudo cp -v /srv/nfs/ubuntu_mate/casper/{vmlinuz,initrd}  
/srv/tftp/ubuntu_mate
```

Kaustadele **tftp** ja **nfs** tuleb anda ka suuremad õigused. Lihtsuse mõttes:

```
sudo chmod -Rfv 777 /srv/{tftp,nfs}
```

Kuna Ubuntu tööjaama ISO pole rohkem tarvis, võib selle lahti ühendada.

```
sudo umount -v /mnt/ubuntu_iso
```

Viimase sammuna muuta PXE algladuri sätetefail vastavalt enda loodud süsteemile. Selleks:

```
sudo nano /srv/tftp/pxelinux.cfg/default
```

Sinna lisada oma menüü loomiseks järgnev:

```
default vesamenu.c32
```

```
label install  
menu label ^Ubuntu MATE Automaatpaigaldus  
menu default  
kernel ubuntu_mate/vmlinuz  
append file=/cdrom/preseed/autoinstall.seed automatic-ubiquity  
initrd=ubuntu_mate/initrd boot=casper netboot=nfs  
nfsroot=192.168.0.100:/srv/nfs/ubuntu_mate/ splash noprompt ---
```

```
Prax_Server_PXE [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
GNU nano 2.9.3 /srv/tftp/pxelinux.cfg/default

default vesamenu.c32

label install
menu label ^Ubuntu MATE Automaatpaigaldus
menu default
kernel ubuntu_mate/vmlinuz
append file=/cdrom/preseed/autoinstall.seed automatic-ubiquity initrd=ubuntu_mate/initrd boot=caspe$

^G Get Help      ^O Write Out    ^W Where Is    [ Read 7 lines ]
^X Exit          ^R Read File    ^M Replace    ^K Cut Text    ^J Justify     ^C Cur Pos    ^U Undo
^_              ^L Read File    ^~ Replace    ^U Uncut Text  ^I To Spell   ^G Go To Line ^E Redo
Right Ctrl
```

default vesamenu.c32 - kasutatakse lihtsa menüü akna loomiseks

label - menüübloki nimetus

menu label - kuvatav menüü tekst

menu default - vaikumisi määratud stiil menüüle

kernel - tuuma asukoht meediumil (antud olukorras siis PXE serveri TFTP kaustas)

netboot=nfs - kasutatakse võrgulaadimist ning väärtus näitab millist tüüpi (saab olla ka näiteks http)

nfsroot - näitab juurkataloogi asukohta, kus paiknevad paigaldusfailid

Sellega on PXE server valmis teostama alglaadimist üle võrgu. Et ka see paigaldus automaatseks muuta, tuleb luua analoogne *preseed* fail nagu praktikumi esimeses

osas sai loodud. Hea moodus on see tõsta isiklikku enose **public_html** kausta. Ning seejärel saab selle serverisse laadida.

Serveris tuleks liigutada see fail **/srv/nfs/ubuntu_mate/preseed** kausta. Kuna PXE server pakub võrgualglaadimist virtuaalmasinate vahelises võrgu, siis klient arvuti internetti paigalduse ajal ei saa. Seetõttu võime eemaldada **preseed** failist lisatarkvara paigaldus käsud. Kuid kuna

Ubuntu Desktop versiooniga esineb järgmine “bug”:

<https://bugs.launchpad.net/ubuntu/+source/debian-installer/+bug/1750597>

Tuleb loodud **preseedi** natukene muuta, ning lisada:

```
echo "auto lo" > /target/etc/network/interfaces; \  
echo "iface lo inet loopback" >> /target/etc/network/interfaces;
```

```
ubiquity ubiquity/success_command string \  
    echo "auto lo" > /target/etc/network/interfaces; \  
    echo "iface lo inet loopback" >> /target/etc/network/interfaces;
```

Esimene käsk kirjutab üle **interfaces** faili sisuga “auto lo” ning teine käsk lisab faili lõppu “iface lo inet loopback”. Vaikimisi on selles failis ainult “loopback” liidese seadistus. Kuid PXE paigalduse puhul kirjutatakse sinna kasutatava võrguliidese read juurde ning seetõttu puudub pärast paigaldust võrk.

Järgmine samm on virtuaalmasina loomine ning seejärel Ubuntu Linux paigaldus kasutades võrgualglaadimist. Nagu ka esimeses osas, saab seda teha käsitsi või skriptiga.

Kui esimeses osas kasutati **NAT** võrguliidest, siis nüüd tuleb valida **Internal Network**.

Muudetud VirtualBoxi skript:

```
#!/bin/bash
```

```
# MACHINENAME on muutuja, mille väärtus saadakse skripti  
käivitamisel esimesest argumendist ehk planeeritav  
virtuaalmasina nimi.
```

```
MACHINENAME=$1
```

Skript käivitatakse ühe muutujaga: ./skripti_nimi.sh "loodava virtuaalmasina nimi"

Kasutatud on materjale aadressilt:

<https://www.andreafortuna.org/2019/10/24/how-to-create-a-virtualbox-vm-from-command-line/>

Skriptis kasutatavatest käskudest saab detailsemalt lugeda VirtualBoxi dokumentatsioonist.

<https://www.virtualbox.org/manual/ch08.htm>

Virtuaalmasina loomine ning registreerimine VirtualBoxiga. Kasutades --register parameetrit ei pea käsitsi loodavat masinat eraldi registreerima VirtualBoxi tarkvara jaoks, vaid see toimub loomise ajal.

Kõik erinevad --ostype variandid saab kuvada käsuga:

"VBoxManage list ostypes"

--basefolder näitab, kus kaustas hakkab loodav virtuaalmasin paiknema.

VBoxManage createvm --name \$MACHINE_NAME --ostype "Ubuntu_64" --register --basefolder \$(xdg-user-dir DESKTOP)

I/O APIC lubamine

VBoxManage modifyvm \$MACHINE_NAME --ioapic on

Muutmälu ja videomälu määratlemine

VBoxManage modifyvm \$MACHINE_NAME --memory 2048 --vram 128

Graafikakontrolleri valik. Valikuvариandid:

-graphicscontroller none|vboxvga|vmsvga|vboxsvga

VBoxManage modifyvm \$MACHINE_NAME --graphicscontroller vmsvga

Võrguliidese tüübi valik: intnet=Internal Network. Võimalik lisada ka mitu. Formaat on järgmine: --nic<1-N> ning tüüp: none|null|nat|natnetwork|bridged|intnet|hostonly|generic

VBoxManage modifyvm \$MACHINE_NAME --nic1 intnet

```

# Virtuaalse kõvaketta loomine. --size parameeter on megabaitides.
# --variant Standard dünaamiliselt suurenev ketas.
VBoxManage createmedium disk --filename $(xdg-user-dir DESKTOP)/
$MACHINE/$MACHINE_DISK.vdi --size 25000 --format VDI --
variant Standard

# Kontrolleri lisamine mäluseadmete jaoks
VBoxManage storagectl $MACHINE --name "SATA Controller" --
add sata --controller IntelAhci

# Kõvaketta külge haakimine
VBoxManage storageattach $MACHINE --storagectl "SATA
Controller" --port 0 --device 0 --type hdd --medium $(xdg-user-
dir DESKTOP)/$MACHINE/$MACHINE_DISK.vdi

# Alglaadimisjärjekorra seadistamine
VBoxManage modifyvm $MACHINE --boot1 disk --boot2 net --
boot3 none --boot4 none

# Virtuaalmasina käivitus
VBoxManage startvm $MACHINE

```

Skripti käivitamine analoogne praktikumi esimeses osas loodud skriptiga.

Pärast paigaldust tuleb loodud virtuaalmasin käsitsi sulgeda, sest miskipärast viimane *reboot* või *poweroff* käsk ei lähe läbi. Seejärel saab muuta võrguliidese uuesti NAT variandiks. Uuesti käivitades on virtuaalmasin kasutuskõlblik ning interneti võimekusega.

Iseseisvaks uurimiseks

FAI - Fully Automatic Installation

<https://fai-project.org/features/>

FOG Project

<https://fogproject.org/>

Cobbler

<https://cobbler.github.io/>

CUBIC ISO Creator

<https://launchpad.net/cubic>

Tagasiside link veelkord:

<https://forms.gle/T9DCRQ9gFSxbz59z7>

Tulevikus täiendamiseks

- Mida teha teistmoodi, et kasutada UEFI masinate puhul
- Kuidas kasutada NFS asemel HTTP-d
- Käsitsi kettajagude defineerimine preseedis
- Väliste skriptide käivitamine preseedis
- Keskkonnamuutujate kasutamine alglaadimisel
- VirtualBoxi asemel KVM
- reboot/poweroff käsk üle võrgu
- Windowsi automaatpaigaldus

Lisa 2 – Tagasiside küsitlus

<https://forms.gle/7Gg6qAk9wrqH72j56>

1. Kuidas loodud materjal sobib õppeainega "Operatsioonisüsteemid ja nende haldamine" ? *

Sobib hästi
Ei oska öelda
Ei sobi hästi

2. Kui vastasite, et ei sobinud, siis miks ?

3. Kas praktikumimaterjal oli: *

Vähe mahukas
Paras
Liiga mahukas

4. Kui kaua umbes kulus antud praktikumi läbimiseks ? *

5. Kas loodud praktikumi läbi tegemine oli: *

Huvitav
Neutraalne
Igav

6. Kuidas hindate materjali arusaadavust ? *

Ei olnud arusaadav
Vähe arusaadav
Arusaadav
Hästi arusaadav

7. Kuidas hindate materjali keerulisust ? *

Liiga keeruline
Jõukohane
Liiga lihtne

8. Kui vastasite liiga keeruline või liiga lihtne, siis põhjendage palun

9. Kui, siis milline on Teie varasem kokkupuude operatsioonisüsteemi haldamisega?

10. Kui Teil on ettepanekuid materjalide parendamiseks või muutmiseks: