

TARTU ÜLIKOOL  
LOODUS- JA TEHNOLOOGIATEADUSKOND  
Tehnoloogiainstituut

Dmitri Pantšenko

**Lahendused asutustele avatud lähtekoodiga tarkvara näitel**

Bakalaureusetöö

Juhendaja: Artjom Lind

Tartu 2014

## SISUKORD

<u>1. Sissejuhatus.....</u>	<u>4</u>
<u>1.1. Avatud koodiga tarkvara.....</u>	<u>5</u>
<u>1.1.1. Sõltuvuse ja sellega kaasnevate riskide vähendamine.....</u>	<u>5</u>
<u>1.1.2. Stabiilsus, turvalisus ja usaldavus.....</u>	<u>6</u>
<u>1.1.3. Ühildavus teiste riistvaraplatformidega.....</u>	<u>6</u>
<u>1.1.4. Kohandamise lihtsus.....</u>	<u>6</u>
<u>1.1.5. Kokkuhoid IT kuludel ja kohaliku IT sektori stimuleerimine.....</u>	<u>7</u>
<u>2. Turul olevad kommertslahendused ja -tooted.....</u>	<u>8</u>
<u>2.1. Marsruutimine.....</u>	<u>8</u>
<u>2.1.1. Cisco Systems tooted.....</u>	<u>8</u>
<u>2.1.2. Hewlett-Packard tooted.....</u>	<u>9</u>
<u>2.1.3. Huawei tooted.....</u>	<u>10</u>
<u>2.1.4. MikroTik tooted.....</u>	<u>11</u>
<u>2.2. Riistvara virtualiseerimine.....</u>	<u>11</u>
<u>2.2.1. VMware vSphere.....</u>	<u>12</u>
<u>2.2.2. Microsoft Hyper-V.....</u>	<u>14</u>
<u>2.2.3. Citrix XenServer.....</u>	<u>15</u>
<u>2.3. Veebiserver.....</u>	<u>16</u>
<u>2.3.1. Microsoft Internet Information Services.....</u>	<u>17</u>
<u>2.4. E-postiserver.....</u>	<u>17</u>
<u>2.4.1. Microsoft Exchange Server.....</u>	<u>18</u>
<u>2.5. Kodukeskjaam.....</u>	<u>19</u>
<u>2.5.1. 3CX Phone System.....</u>	<u>20</u>
<u>3. Lahendused avatud lähtekoodiga tarkvara näitel.....</u>	<u>22</u>
<u>3.1. Marsruutimine.....</u>	<u>22</u>
<u>3.1.1. Marsruutimisprotokollid.....</u>	<u>22</u>
<u>3.1.2. Läbilaskevõime hindamine SSL-tüüpi virtuaalsetes privaatvõrkudes (VPN) .....</u>	<u>23</u>
<u>3.1.3. Näiteid Debian GNU/Linux distributsioonil.....</u>	<u>25</u>
<u>3.2. Riistvara virtualiseerimine.....</u>	<u>28</u>
<u>3.2.1. KVM.....</u>	<u>29</u>
<u>3.2.2. Haldustarkvara libvirt.....</u>	<u>30</u>
<u>3.3. Veebiserver.....</u>	<u>32</u>
<u>3.3.1. Veebiklastri seadistamine Nginx ja Apache HTTP Server abil.....</u>	<u>33</u>
<u>3.4. E-postiserver.....</u>	<u>36</u>
<u>3.4.1. E-postiserver Postfix, Dovecot ja RoundCube abil.....</u>	<u>36</u>

<u>3.5. Kodukeskjaam.....</u>	<u>40</u>
<u>3.5.1. Näiteid kodukeskjaama seadistamisest Asterisk abil.....</u>	<u>41</u>
<u>4. Kokkuvõte.....</u>	<u>46</u>
<u>5. Kasutatud kirjanduse loetelu.....</u>	<u>49</u>
<u>6. Summary.....</u>	<u>55</u>
<u>7. Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks.....</u>	<u>58</u>

## 1. SISSEJUHATUS

Ühiskonna arenguga tähtsustatakse järjest rohkem informatsiooni säilivust, levikut ja leviku kiirust. Selline nähtus on muutnud arvutusmasinad paljudele inimestele igapäevasteks töövahenditeks. Samuti ka paljudes ettevõtetes on infotehnoloogia muutunud äritegevuse lahutamatuks osaks. Ettevõtlus nõuab IT-lt kvaliteetseid lahendusi, kiire paigaldusega ja võimalikult odavalt.

Igal arvutil on kaks eraldiseisvat, kuid üks teisest sõltuvat komponenti - riistvara ja tarkvara. Riistvaraga tähistatakse elektroonilisi komponente, mis töödeldes elektroonilisi signaale võimaldavad manipuleerida andmetega. Tarkvara all mõistetakse programmide kogu, mis annab riistvarale juhiseid, kuidas töödelda andmeid.

Tänapäeval müüakse enamik üldotstarbelisi arvuteid juba eelnevalt paigaldatud tarkvaraga. Leidub ka arvutimudeleid, mida pole võimalik osta ilma operatsioonisüsteemita. Lisaks operatsioonisüsteemile uuel arvutil võib olla tootja poolt paigaldatud kontoritarkvarapakett ja viirusetõrje. Kuna paljudel arvutitootjatel on sõlmitud lepingud tarkvaratootjatega, siis eelnevalt paigaldatud tarkvara on üldjuhul tasuline. Operatsioonisüsteemi hind on tavaliselt uue arvuti hinna sees, kuid näiteks tekstitöötlusprogrammi käivitamisel või viirusetõrje juhtpaneeli avamisel selgub, et tarkvara on vaja veel välja osta. [1]

Eesti haridussüsteemis on ligikaudu 10000 arvutit ning enamikel nendest on paigaldatud kommertstarkvara. Endise Eesti haridus- ja teadusministri Jaak Aaviksoo teatel soovis tarkvaratootja Microsoft tõsta hinda enam kui 10 korda. Pikaajaliste läbirääkimiste tulemusena 2013. aasta lõpus jõuti kokkuleppele, et hindade tõus toimub järk-järguliselt aastani 2017. [2] Samuti tähendab kommertstarkvara kasutamine haridusasutustes ühiskonna süstemaatilist sõltuvust sellest.

2012. aastal viidi läbi uuring, millele vastas 31 Tallinna munitsipaalkooli. Uuringust selgub, et 10 kooli (32%) serveritel töötavad ainult Microsoft Windows perekonna operatsioonisüsteemid, 9 kooli (29%) serveritel töötavad nii Microsoft Windows perekonna kui ka avatud lähtekoodiga operatsioonisüsteemid ja 4 kooli (13%) serveritel ainult avatud lähtekoodiga operatsioonisüsteemid. Ülejäänud koolidele, millel puuduvad oma serverid, rendib Tallinna Haridusamet servereid ja ostab siise tugiteenust, kusjuures kasutatakse ainult kommertstarkvara. Samuti on uuringus märgitud, et baasteenused nagu DHCP ja DNS on

lahendatud avatud lähtekoodiga tarkvara abil ainult neljas koolis. E-postiteenuste osutamiseks kasutatakse avatud lähtekoodiga tarkvara ainult viies koolis, 19 koolis töötab Microsoft Exchange lahendus ja seitse kooli kasutab pilveteenuseid. [3]

Käesoleva töö eesmärgiks on näidata avatud lähtekoodiga tarkvara kasutamise ja juurutamise eeliseid asutustes kommertstarkvara ja -toodete ees. Töö on suunatud isikutele, kes omavad kogemusi asutuste IT infrastruktuuri haldamises.

## **1.1. Avatud koodiga tarkvara**

Avatud lähtekoodiga tarkvara arendatakse ja levitatakse erinevate litsentside all. Sõltumatult selle litsentsi tüübist ei ole kasutamine ja levitamine piiratud. Nimetatud nähtusest võib tekkida mulje, et tegemist on heategevusprojektiga. Tegelikuses on avatud lähtekoodiga tarkvaral oma ärimudel, mis sisaldab mitmeid varjatud tuluallikaid. Üheks selliseks tuluallikaks on tasuline seadistamis- või tugiteenus. [4] Teisest küljest on paljud riistvaratootjad huvitatud, et nende riistvara oleks võimalik kasutada ilma lisakulutusteta tarkvarale. Selliste riistvaratootjate hulka kuulub Intel, kes on suuruselt teine Linux-i tuuma arendaja ja *Linux Foundation*-i sponsor. [5] Antud ärimudeli eeliseks on kiiresti kasvav kasutajaskond, mis aitab kaasa tarkvara kiirele levikule.

Järgnevates alampeatükkides on kirjeldatud võimalused, mis kaasnevad avatud lähtekoodiga tarkvara kasutamisega.

### **1.1.1. Sõltuvuse ja sellega kaasnevate riskide vähendamine**

Ettevõtte vajavad tarkvara paljude kriitiliste ülesannete lahendamiseks nagu näiteks andmebaaside haldus, e-kaubandus, raamatupidamine ning erinevate andmete töötlemine ja analüüs. Ettevõtluses esinevad sageli ärisuundade arengud ja muutused, millega peavad kaasnema muutused tarkvaras. Igasugused muudatused võivad olla rahaliselt kulukad ja aeganõudvad, kuid sellise riskiga tuleb alati arvestada.

Kui ettevõtte on juba varem otsustanud kasutusele võtta kommertstarkvara, siis selle tootja ei pruugi alati nõustuma seda edasi arendama ning tulevased veaparandused on tootja kontrolli all. Samuti on kasulik meeles pidada, et iga toote tugi ei ole lõpmatu ning igal tootjal on õigus muuta oma toodete hindu ja kasutamistingimusi. [6] Sellised riskid kaasnevad kommertstarkvaraga ja võivad tähendada suurt kasumikadu.

### **1.1.2. Stabiilsus, turvalisus ja usaldavus**

Avatud lähtekoodiga tarkvara arendavad inimesed, kes seda ise kasutavad. Koostöö erinevate arendajate vahel tähendab kiiret eesmärkide saavutamist. Kuna tarkvara lähtekood on avatud, siis ka vigade ja turvaaukude tuvastamise protsess toimub tunduvalt kiiremini ja efektiivsemalt, sest lähtekoodiga on võimalus kõigil huvilistel tutvuda. Kommertstarkvara kasutamisel kasutajal tuleb teavitada arendajat veast, arendaja aga peab kinnitama vea, leidma lahenduse ja katsetama seda ning seejärel väljastama paranduse. Selline protsess võib võtta nädalaid või kuid, mis võib tekitada ettevõttele kahju. Lisaks piiravad paljud kommertstarkvara litsensid ja suletud lähtekood iseseisvate paranduste või täienduste tegemist. [7]

Paljudel kommertstarkvaratoodete litsents sisaldab väidet, et tootjal on õigus koguda andmeid kasutaja tegevusest toote arendamise eesmärgiga. Näiteks vaikeseadetega Internet Explorer saadab andmed kasutaja tegevusest Customer Experience Improvement Program, SmartScreen Filter ja Suggested Sites tehnoloogiate abil, kusjuures saadetavate andmetega ei ole võimalik tutvuda. [8] Leidub piisavalt palju selliseid kommertstooteid, mis ei luba statistika kogumist ja saatmist keelata. Avatud lähtekoodiga tarkvara puhul hoolitsevad arendajad alati ise selle eest, et nendel oleks valikuvõimalus.

### **1.1.3. Ühildavus teiste riistvaraplatvormidega**

Tänapäeval enamik üldotstarbelisi arvuteid ja suurim osa servereid töötavad arhitektuuridel x86 või x86\_64. Näiteks kui ettevõttel suurenevad vajadused arvutusjõudlusele või tekib mõni muu põhjus ühelt arhitektuurilt teisele kolimiseks, siis avatud koodiga tarkvara kompileerimine ja paigaldamine ei ole keeruline ega aega või muid ressursse nõudev protsess. Sellel samal põhjusel avatud lähtekoodiga tarkvara, mis on kõigile avalikult kättesaadav, on juurutatud NASA kosmosesüstikutesse, nutitelefonidesse ja navigatsiooniseadmetesse. [6]

### **1.1.4. Kohandamise lihtsus**

Sõltumata litsensi tüübist on enamik olemasolevaid tarkvaralisi lahendusi universaalsed, et tagada mingit funktsionaalsust suuremale kasutajate arvule. Võrreldes erilahendusega universaalne toode ei ole võimeline rahuldama kasutaja kõiki vajadusi. Juhul, kui on olemas osaliselt vajadusi rahuldavad tooted, mida on võimalik täiendada, ei ole otstarbekas arendada nende analooge algusest. Kasutades avatud lähtekoodiga tarkvara, on olemas võimalus muuta tarkvara olemasolevaid funktsioone ja täiendada uutege, kusjuures ajalised ja rahalised

kulutused kaasnevad ainult modifitseerimisele. Samuti saab integreerida antud tarkvara olemasolevatesse lahendustesse.

#### **1.1.5. Kokkuvõid IT kuludel ja kohaliku IT sektori stimuleerimine**

Avatud lähtekoodiga tarkvara kasutamine ja levitamine tasuta tähendab ettevõttele IT kulude vähenemist tarkvara litsentside arvelt. Avatud lähtekoodiga tarkvaral põhinevate lahenduste juurutamise hind võib olla suurem kommertstarkvaral põhinevatest, kuid tänu tarkvara maksumuse puudumisele on lahenduste lõpphind väiksem. Üleminek avatud lähtekoodiga tarkvarale suurendab nõudlust infosüsteemide arendamis- ja tugiteenustele, mis tähendab kohaliku IT sektori stimuleerimist ja uute töökohtade teket IT sektoris, lisaks ka impordi vähenemist. [9]

## 2. TURUL OLEVAD KOMMERTSLAHENDUSED JA -TOOTED

### 2.1. Marsruutimine

Tänapäeval on raske leida tegutsevat ettevõtet, millel puuduks vajadus internetiühendusele. Suure tõenäosusega on ettevõttel ka rohkem kui üks seade, mida on vaja ühendada andmesidevõrku. Selline olukord tähendab, et kohalik võrk on vaja eraldada Internetist, mis omakorda nõuab marsruuteri olemasolu.

Kui on tegemist ettevõttega, millel on kuni 25 töötajat, kes vajavad ligipääsu avalikele veebiressursidele ja e-postile, siis saab vaieldamatult piirduda kodu ja väikettevõtetele mõeldud toodetega. Sellised marsruuterid on saadavad kodutehnika poodidest ning nende hind jääb vahemikku 20 kuni 50 eurot.

Asutused, kus on palju töökohti, mis vahendavad suuri andmemahte, vajavad krüpteeritud virtuaalseid privaatvõrke või nõuavad lisateenuseid (näiteks veebiproksi ja võrguliikluse kontrollimine või jälgimine), ei saa oma nõudeid täita kodu ja väikettevõtetele mõeldud ruuteriga. Nimetatud olukordades on tavaliselt kaks tüüplahendust:

1. vajaliku funktsionaalsusega kommertstoote leidmine, ostmine ja seadistamine;
2. x86-x86\_64 platvormil üldotstarbelise arvuti ostmine ja seadistamine.

Järgnevad alampeatükid moodustavad ülevaate mõnedest turul olevatest marsruuteritest.

#### 2.1.1. Cisco Systems tooted

Ettevõtte Cisco Systems on loodud 1984. aastal ning on tegelenud asutamise algusest võrguseadmete arenduse ja tootmisega. Esimene müüki läinud marsruuter oli AGS (1986. aasta) ning selle põhierinevus konkureerivatest toodetest oli mitme erineva OSI mudeli kolmanda kihi protokollitugi. [10]

Cisco Systems rühmitab oma ruuterid kolme klassi:

1. *Branch* (harukontorid, väikse ja keskmise suurusega ettevõtted);
2. *Network Edge* (suured ettevõtted, linnavõrgud ja laivõrgud);
3. *Service Provider* (teenuspakkujad ja magistraalvõrgud).

Kuna käesolev töö ei puuduta suuri ettevõtteid ega magistraalvõrke, tehakse ülevaade ainult esimesest klassist. Esimene klass sisaldab mitu tooteseeriat: 800, 1900, 2900 ja 3900,



kusjuures iga tooteseeria sisaldab mitu mudelit [11]. Tooteseeriad erinevad peamiselt soovitava kasutajate arvult, läbilaske võimelt ja laiendusvõimalustelt. Allolevas tabelis on toodud välja esimese klassi tooteseeriade erinevused.

	<b>RV</b>	<b>800</b>	<b>1900</b>	<b>2900</b>	<b>3900</b>
100 Mbps WAN portide arv	0-2	1	0	0	0
1 Gbps WAN portide arv	0-2	0-1	2	2-3	3-4
Laienduskaartide (EHWIC) arv	0	0	2	4	3-4
Soovitatav internetiühenduse kiirus, Mbps	tundmatu	≤15	≤25	≤75	≤350
Läbilaskevõime, kadudeta, Mbps	90-800	197-1400	2770-2932	3114-5136	6903-8675
Läbilaskevõime, IPSec, Mbps	5-97	46-125	149-170	170-282	770-1503
Läbilaskevõime, SSL VPN, Mbps	0-25	10-25	50-75	75-150	200-500
Hinnavahe, EUR	70-370	270-1700	650-1900	1050-4400	≥5700

Tabel 1. Cisco Systems madala klassi marsruuterite seeriade võrdlus. [12], [13]

Mudelite peamiseks erinevaks on lisavõrguliideste arv ja tüüp (näiteks WLAN, DSL või 3G). Alates 800-seeriast kõikidel mudelitel on olemas riistvaraline kiirendi VPN ühenduste krüpteerimiseks järgmiste algoritmidega: DES, 3DES ja AES. Samuti kõik mudelid välja arvatud 800-seeria odavamad toetavad laia valiku VPN standarde: SSL VPN, IPSec, Easy VPN, Dynamic Multipoint VPN (DMVPN), Group Encrypted Transport VPN (GET VPN), V3PN, MPLS VPN. [11] Kõikidele eelnevalt nimetatud standarditele peale IPSec on Cisco Systems-il omandiõigus, nende kasutamine on piiratud litsentsidega ning nende kasutamiseks on vaja mõlemal pool Cisco Systems-i seadet või tarkvara. [11], [14]

### 2.1.2. Hewlett-Packard tooted

2010. aastal ostis Hewlett-Packard ettevõtte 3Com, mis oli maailmas üks juhtivaid võrguseadmete tootjaid. [15] Võrguseadmed moodustavad väga väikse osa ettevõtte Hewlett-Packard (edaspidi HP) toodangust, kuid vaatamata sellele on HP-l lai valik marsruutereid, mis on jaotatud järgmistesse klassidesse:

1. *Branch* (väikesed, keskmised ja suured ettevõtted);
2. *Campus* (suured ettevõtted, linnavõrgud ja laivõrgud);
3. *Data Center* (andmekeskused ja magistraalvõrgud).

Väikestele ja keskmise suurusega ettevõtetele sobivad järmised seeriad esimesest klassist: MSR900, MSR20 ja MSR30. Nimetatud seeriad erinevad läbilaskevõime võimepoolest ning kõik välja arvatud esimene võimaldavad kasutada laienduskaarte. Allolev tabel toob välja peamised erinevused.

	<b>MSR900</b>	<b>MSR20</b>	<b>MSR30</b>
Läbilaskevõime, Kpps	70-300	160-180	220-360
Läbilaskevõime, IPv4, Mbps	995	2000	3000
Läbilaskevõime, IPSec, Mbps	144-148	396	1902-3258
Hinnavahe, EUR	250-1250	600-1650	1050-6000

Tabel 2. Hewlett-Packard madala klassi marsruuterite seeriade võrdlus. [16], [17]

Igal tooteseerial on võimalik erinev võrguliiste konfiguratsioon (näiteks WLAN, DSL või 3G) ning kõik seeriad võimaldavad riistvaralist kiirendust IPSec ühenduse krüpteerimiseks. Lisaks IPSec VPN standardile on toetatud DVPN, mis on HP omand. Ruutimisprotokollidest on toetatud RIPv1, RIPv2, OSPF ja IS-IS.

### 2.1.3. Huawei tooted

Huawei on 1987. aastal loodud Hiina ettevõtte, mis toodab telekommunikatsiooniseadmeid. Lisaks väikese ja keskmise suurusega ettevõtetele toodab Huawei marsruutereid ka lai- ja magistraalvõrkudele. Tooteseeriad erinevad läbilaskevõime ja funktsionaalsuse poolest, seeria mudelid aga liideste konfiguratsiooni poolest. Allolev tabel kirjeldab väikse ja keskmise suurusega ettevõtetele suunatud Huawei marsruutereid. [18]

	<b>AR150</b>	<b>AR200</b>	<b>AR1200</b>	<b>AR2200</b>
LAN ethernet porte	4 FE	4 FE	8 FE	-
Läbilaskevõim, Kpps	300	450	450	350-5000
Läbilaskevõime, IPv4, Mbps	500	800	990	1900-5900
Läbilaskevõime, IPSec, Mbps	teadmata	40	340	1300-3000
Hinnavahe, EUR	300-750	400-1730	630-1460	≥800

Tabel 3. Huawei lihtsamate marsruuterite seeriade võrdlus. [18], [19]

Kõik seeriad toetavad IPSec, GRE ja L2TP VPN standarte, alates AR200 seeriast on toetatud ka SSL VPN. Samuti toetavad kõik marsruuterid RIP v1/v2, OSPF, IS-IS ja BGP ruutimisprotokolle. [18]

#### 2.1.4. MikroTik tooted

MikroTik on 1995. aastal registreeritud Läti ettevõtte, mis tegeleb nii kaabel- kui ka raadiovõrguseadmete arendamisega ja tootmisega. Tuntumad tooted on RouterBOARD – suure mudelivalikuga ruuterite peamoodulid ja RouterOS – laiade võimalustega marsruuteritele mõeldud operatsioonisüsteem, mis põhineb Linux tuumal. Enamikul mudelitest leidub mitu versiooni, mis erinevad liideste tüübilt ja arvult ning lisade poolest (näiteks USB port). Samuti on võimalik enamik mudeleid osta ilma korpuseta või valida tava-, välis- ja püstakukorpuse vahel. [20] Allolev tabel toob esile mõnede mudelite erinevused.

	433	435	450	493	2011	800	1016	1036
Ethernet portide arv	3	3	5	9	10	3	12	8-12
MiniPCI pesede arv	3	5	0	3	0	4	0	0
Läbilaskevõime, IPv4, Gbps	0,2	1	0,2-1	0,2	1,2	1,9	12,0	27,9
Hinnavahe, USD	90-165	190	70-100	130-200	80-100	360	625	1000-1300

Tabel 4. MikroTik marsruuterite mudelite võrdlus. [20]

MikroTik poolt arendatav RouterOS toetab erinevaid ruutimisprotokolle, näiteks RIP v1/v2, RIPng, OSPF v2/v3, BGPv4, mitmeid VPN standarte, sealhulgas IPSec, OpenVPN, PPTP L2TP ja SSTP ning erinevaid teenuseid nagu veebiproksi, veebi pöördproksi, SOCKS, DNS, Samba ja NTP. RouterOS on ametlikult toetatud x86 arhitektuuril. [21]

## 2.2. Riistvara virtualiseerimine

Kui on olemas vajadus osutada korraga mitu erinevat teenust nagu näiteks veeb, e-post ja VoIP, siis ei ole otstarbekas seadistada kõik ühele operatsioonisüsteemile. Sellega kaasneb mitmeid probleeme, näiteks ühe teenuse turvariskid või ülekoormus mõjutavad teiste teenuste turvalisust või kättesaadavust vastavalt, samuti on teenuste migreerimine tulevikus raskendatud. Üheks võimalikuks lahenduseks on iga teenuse paigutamine sõltumatule

riistvarale, mis kindlustab teenuseid sõltumatusena üksteisest, kuid samal ajal ka mitmekordistab kulusid riistvarale, selle ülalpidamiseks, elektrienergiaks ja jahutusele.

Tänu hoogsale riistvara arengule suure tõenäosusega ükski varem nimetatud teenustest ei kasuta riistvaralisi ressursse täies mahus. Veelgi enam, kui näiteks e-posti serveril on alla 100 postkasti, siis suurima osa ajast on sellepoolne ressurside kasutatavus tühikäigulähedane, ehk suurim osa riistvaralisi ressursse on kasutamata. Selline olukord tähendab vahendite raiskamist, ning probleemi leevendamiseks üks populaarne lahendus on virtualiseerimistehnoloogia juurutamine.

Riistvara virtualiseerimise mõte ei ole uus, selle juured pärinevad eelmise sajandi kuuekümnendatest. Vaatamata sellele virtualiseerimistehnoloogia toega x86 arhitektuuril protsessorid ilmusid ainult 2006. aastal. Vähem kui kümne aastaga on loodud mitmeid virtuaalmasinahaldureid ehk hüperviisoreid, mis erinevad peamiselt tüübi, litsensi ja erinevate funktsioonide/piirangute poolest. Viirtuaalmasinahaldurite tüüpe on kaks: esimene, kui hüperviisor on iseseisev operatsioonisüsteem ja teine, kui hüperviisor töötab mõne teise operatsioonisüsteemi peal. [22] Sõltumata ettevõtte tegevusalast tihtipeale ka suurte kogemustega süsteemiadministraatoril on keeruline teha valik saadaval olevatest virtuaalmasinahaldurite hulgast. Tavaliselt hinnatakse järmisi omadusi: stabiilsus, turvalisus, funktsionaalsus, paindlikkus ja toetavus. Järgnevad alampeatükid annavad ülevaate mõnedest kommertsilahendustest. [23]

### **2.2.1. VMware vSphere**

Ettevõtte VMware alustas tegevust 1998. aastal ning aasta hiljem esitles oma esimest toodet VMware Workstation, mis teostas riistvara virtualiseerimist ainult tarkvaraliselt. Mõned aastad hiljem siseneti ka serverite turule. Aastal 2011 ilmus vSphere 5.0, mille kaks tähtsamat komponenti on ESXi (hüperviisor) ja vCenter Server (juhtimistarkvara). ESXi 5 eelkäiad olid ülesehitatud operatsioonisüsteemil Red Hat Enterprise Linux, kuid viies põlvkond töötab omal tuumal nimega VMkernel. [24]

Lisaks virtuaalmasinahaldurile ja juhtimistarkvarale sisaldab vSphere veel mitmeid vahendeid, mis teevad serveri haldamise mugavamaks. Näiteks vMotion võimaldab tõsta töötavat virtuaalmasinat ühest füüsilisest serverist teise ilma virtuaalmasinat peatamata ja Storage vMotion võimaldab tõsta töötava virtuaalmasina virtuaalketast füüsiliste salvestusseadmete vahel. Sõltuvalt litsentsist vSphere võib sisaldada vahendid, mis

automatiseerivad koormuse ühtlast jaotust: Distributed Resources Scheduler – füüsiliste serverite vahel ja Storage Distributed Resources Scheduler – salvestusseadmete vahel. [24]

Erinevate vSphere toodete litsentse arvestatakse ühe füüsilise protsessori kohta, ehk kui serveris on kaks füüsilist protsessorit, siis tuleb osta kaks litsentsi. Samuti on olemas võimalus osta komplektina, mis sisaldab litsentsi tarkvarale vCenter ja kolmele füüsilisele serverile, igapähe kuni kaks füüsilist protsessorit. [25] Ülevaate vSphere toodete ja komplektide erinevusest annavad alloolevad tabelid.

	<b>Standard</b>	<b>Enterprise</b>	<b>Enterprise Plus</b>
Komponendid	ESXi, vMotion, Storage vMotion, High Availability and Fault Tolerance, Data protection and Replication, vShield Endpoint	Eelmsed ja Distributed Resource Scheduler and Distributed Power Management, Storage APIs for Array Multipathing, Big Data Extensions, Reliable Memory	Eelmsed ja Distributed Switch, Storage Distributed Resource Scheduler, I/O Controls, Host Profiles and Auto Deploy, Flash Read Cache, App HA
Litsentsi hind, EUR/1CPU	894,50	2585,00	3145,00
Kohustuslik tugi, EUR/aasta	245,51	543,18	660,09
Laiendatud tugi, EUR/aasta	290,47	646,60	785,99
vCenter hind, EUR	Foundation 1345,00 + tugi: kohustuslik 490,12 / laiendatud: 580,05 Standard 4495,00 + tugi: kohustuslik 943,37 / laiendatud: 1123,23		
Väikseim kulu, EUR	2975,13	5003,3	5640,21

Tabel 5. VMware vSphere toodete võrdlus, hinnad käbemaksuta.

	<b>Essentials Kit</b>	<b>Essentials Plus Kit</b>	<b>Essentials Plus Kit with vSphere Storage Appliance</b>
Komponendid	ESXi	Eelmised ja vMotion, High Availability, Data protection and Replication, vShield Endpoint	Eelmised ja Storage vMotion
vCenter	Essentials		
Litsentsi hind, EUR/1CPU	445,50	3105,00	3455,00
Kohustuslik tugi, EUR/aasta	58,45	653,06	1160,84
Laiendatud tugi, EUR/aasta	268,50	777,58	1382,22
Väikseim kulu, EUR	503,95	3758,06	4615,84

Tabel 6. VMware vSphere komplektide võrdlus, hinnad käibemaksuta. [24, 25]

Virtualiseerimistarkvaral vSphere on sõltumata litsentsist kõrged võimalused: kuni 2 TB mälu, 160 protsessorit ja 512 virtuaalmasinat ühel füüsilisel serveril ning kuni 32 virtuaalset protsessorit ja 1 TB mälu ühel virtuaalmasinal. [24]

### 2.2.2. Microsoft Hyper-V

Hyper-V on tarkvaratootja Microsoft arendatav virtualiseerimisplatvorm, mida on võimalik kasutada kolmel viisil [26]:

1. Iseseisva operatsioonisüsteemina (Windows Hyper-V Server 2008, 2008 R2, 2012 ja 2012 R2);
2. Windows Server 2008, 2008 R2, 2012 ja 2012 R2 rollina;
3. Windows 8/8.1 Professional või Enterprise täiendusena.

Esmakordselt ilmus Hyper-V 2008. aastal ning terve aasta ulatuses täiendati toodet klientide tagasiside alusel. [26] Selle tulemusena ilmus 2009. aastal Hyper-V 2008 R2 versioon, millele

lisandusid näiteks Live Migration – võimalus töötava virtuaalmasina tõstmiseks ühest füüsilisest serverist teise ilma esimest peatamata ja Dynamic Memory – võimalus eraldada virtuaalmasinale ainult nii palju mälu, kui see tegelikult kasutab. [26, 27]

Kasutades toodet iseseisva operatsioonisüsteemina (Windows Hyper-V Server) puudub sellel graafiline kasutajaliides Windows Explorer ning toodet saab hallata järgnevatel viisidel:

1. kohalikust käsureast (PowerShell);
2. kaudselt kasutades tarkvarapaketi Remote Server Administration Tools vahendeid;
3. kasutades kolmanda tootja vahendeid (nt Corefig või 5nine Manager for Hyper-V).

Kõik Microsoft Windows Hyper-V Server versioonid on tasuta ning nende paigaldusketastekujutised on saadavad tootja kodulehel.

Operatsioonisüsteem Windows Server omab graafilist kasutajaliidest ning graafilisi haldusvahendeid Hyper-V jaoks, kuid ei ole tasuta toode. Allolev tabel annab ülevaate Windows Server 2012 R2 väljaannete erinevusest ja maksumustest. [28]

	<b>Foundation</b>	<b>Essentials</b>	<b>Standard</b>	<b>Datacenter</b>
Hyper-V võimalus	puudub	olemas	olemas	olemas
Virtuaalkeskondade (VOSE) arv	0	1	2	piiramata
Litsentseerimise mudel	15 kasutajat	25 kasutajat	füüsiliste protsessorite arv ja CAL	füüsiliste protsessorite arv ja CAL
Hind, USD	OEM ainult	501	882	6155

Tabel 7. Microsoft Windows Server 2012 R2 väljaannete võrdlus. [28]

Uuendatud Microsoft Hyper-V virtuaalmasinahaldur paistab silma oma kõrgete tehniliste võimalustega: kuni 320 loogilist protsessorit, 4 TB mälu ja 1024 aktiivset virtuaalmasinat ühel füüsilisel serveril ning 64 virtuaalset protsessorit ja 1 TB mälu ühel virtuaalmasinal. [26, 28]

### **2.2.3. Citrix XenServer**

Citrix XenServer on iseseisev operatsioonisüsteem, mis töötab Linux tuumal ja sisaldab Xen virtuaalmasinahaldurit. Ettevõtte XenSource, mis tegeles Xen arendusega, oli asutatud 2003. aastal Cambridge Ülikoolis Ian Pratt eestvedamisel ning 2007. aastal ostis Citrix XenSource koos tarkvara Xen kaubamärgiga. Virtuaalmasinahaldur Xen on alati olnud tasuta ja avatud lähtekoodiga, kuid sellele lisaks sisaldas XenServer tasulisi ja suletud lähtekoodiga

tööriistu. [29] Sarnaselt VMware vSphere tarkvaraga oli Citrix XenServer lahendusel mitu väljaannet, mis erinesid tasuliste tööriistade olemasolu poolest.

Alates XenServer versioonist 6.2, mis ilmus 2013. aasta juunis, otsustas Citrix avada kõikide tasuliste vahendite lähtekoodi ja muuta need tasuvabaks. Tasuliseks on jäänud ööpäevaringne tugi, ligipääs teadmiste baasile ja mõned teised teenused. [30]

Tarkvara XenServer on võimalik hallata nii käsurealt kui ka graafilise haldusvahendiga XenCenter, mis töötab Microsoft Windows keskkonnas. Tööriistale XenCenter on olemas mitmeplatvormilisi ning veebipõhiseid alternatiive.

Viimane XenServer versioon on piiratud 160 loogilise protsessoriga, 1 TB mälega ja 500 aktiivse virtuaalmasinaga ühel füüsilisel serveril ning 16 virtuaalse protsessori ja 128 GB mälega ühel virtuaalmasinal.

### **2.3. Veebiserver**

Kahekümne esimese sajandi teistel aastakümnel on veebilehekülg muutunud ettevõtete visiitkaardiks. Palju on ka selliseid ettevõtteid, mille veebileht on peamine töökeskkond ja sidekanal klientidega. Veebilehe avamine nõuab mitme erineva teenuse sisseostu: materjalide koostamine, disainimine, programmeerimine ja majutamine, kusjuures viimane loetletutest on vajalik veebilehe terve tööaja ulatuses.

Veebilehe majutamiseks on kaks peamist võimalust - avaliku majutusteenuse või sõltumatu (virtuaalse) serveri kasutamine. Kaasaegsetel veebimajutustel on head tingimused: mõistlikud ketta- ja andmesidemahud, mõnede skriptimiskeelde toed, andmebaasi, e-posti majutuse ja toimingujasti võimalused ning madalad kuutasud. Samuti võivad veebimajutuse kasutamisel esineda mitmed probleemid: andmemahtude puudujääk, ressursside jagamine teiste kasutajatega, vajaliku lisatarkvara (näiteks PDF dokumentide loomiseks) ja erinevate võimaluste (näiteks turvasoklite kihiga kaitstud ühenduste) puudumine ning ligipääs tarkvarale ja andmetele teenusepakkujal. Eelnevalt nimetatud probleemid saab lahendada majutades veebilehe privaatsele serverile.

Ettevõtte Netcraft poolt 2014. aasta jaanuaris läbiviidud uuringu kohaselt jaguneb veebiserveri tarkvarade turuosad järgnevalt.



<b>Arendaja</b>	<b>Kõik leheküljed</b>	<b>Aktiivsed leheküljed</b>	<b>Suurima külastavusega</b>
Apache	41.64%	54.50%	55.45%
Microsoft	29.49%	11.97%	12.66%
Nginx	14.10%	11.61%	15.91%
Google	2.27%	8.54%	3.03%
Teised	12.50%	13.38%	12.95%

Tabel 8. Veebiserveri tarkvarade turujaotus jaanuaris 2014. [31]

Tabelis 8 väljatoodud ettevõtetest arendab kommertslahendusi Microsoft (IIS - Internet Information Services) ja Google (GWS - Google Web Server). Vaatamata sellele, et GWS on Apache edasiarendus, kasutatakse seda ainult ettevõttesiseselt. [32] Selliselt jääb ainsaks populaarseks kommertslahenduseks Microsoft IIS.

### **2.3.1. Microsoft Internet Information Services**

Microsoft IIS 1.0 ilmus täiendusena Windows NT 3.51 jaoks ning toetas selliseid teenuseid nagu HTTP, Gopher ja WAIS. Versioon 3.0 toetas skriptimiskeelt ASP 1.0, mis võimaldas luua dünaamilisi lehekülgi. Alates versioonist 5.0, mis ilmus operatsioonisüsteemiga Windows 2000, on IIS muutunud Windows komponendiks. [33]

Internet Information Services versioonid 6.0 ja uuemad omavad tähtsat erinevust võrreldes enamike rivaalidega – IIS HTTP ühenduste kuular töötab operatsioonisüsteemi tuuma tasemel ja leiab aset TCP/IP protokollipinu kõrval. Nimetatud lähenemine arendaja väitel tagab suurema päringute töötlemise kiiruse ja tõstab veebiserveri stabiilsust. Lisaks on asendatud skriptimiskeel ASP raamistikuga ASP.NET, mis toetab mitmeid keeli nagu näiteks C#, J# ja VB.NET. [34]

Alates IIS 8.0 on toetatud raamistik ASP.NET 4.5 ja lisandunud selliseid funktsioone nagu dünaamiline IP-aadressi blokeerimine ja arvutusjõudluse piiramine eraldi igale veebilehele. Samuti on toetatud ka CGI ja FastCGI liidesed, mis võimaldavad paigaldada teiste skriptimiskeelte (näiteks PHP ja Perl) tuge. [33]

## **2.4. E-postiserver**

Sarnaselt veebileheküljega omab igapäeva äritegevuses suurt tähtsust ka elektronpost. Kõige lihtsam viis e-posti aadressi saamiseks on registreerida see mõnes avalikus e-postiteenuses. Juhul, kui ettevõtte soovib saada vähemalt ühte e-posti aadressi omalt poolt valitud domeeninimega, tuleb osta e-posti majutusteenust. Selleks, et välistada ligipääs kirjadele

kolmandatel isikutel, kasutada ettevõttesiseseid autentimismehhanisme, kehtestada erilisi saatmis- ja vastuvõtmisreegleid ning korraldada tingimuslike suunamisi on lahehenduseks kasutada privaatset e-postiserverit.

Ettevõtte Cybercom poolt 2013. aasta jooksul läbiviidud uuringu kohaselt jagunevad teenuste SMTP ja IMAP serveritarkvara turuosad järgnevalt.

<b>SMTP</b>		<b>IMAP</b>	
<b>Nimetus</b>	<b>Turuosa</b>	<b>Nimetus</b>	<b>Turuosa</b>
Postfix	25,79%	Dovecot	52,87%
Exim	23,28%	Courier	27,14%
(tundmatu)	10,92%	Exchange	2,96%
Qmail	9,26%	IdeaImapServer	2,87%
PowerMTA	8,89%	UW-IMAP	2,32%
Exchange	8,19%	NetArt	1,56%
Sendmail	5,30%	MailEnable	1,43%
MailEnable	1,85%	Cyrus	1,02%
IdeaSmtpServer	1,69%	Zimbra	0,81%
Cisco Ironport	0,84%	hMailServer	0,79%

Tabel 9. SMTP ja IMAP teenuste serveritarkvara jaotus aastal 2013. [35]

Tabelist 9 on näha, et üle poole nii SMTP kui ka IMAP serveritest on loodud avatud lähtekoodiga tarkvaral. Vaatamata sellele on kasulik vaadelda ka kommertslahendusi nagu Microsoft Exchange Server.

#### **2.4.1. Microsoft Exchange Server**

Arendades tarkvara Exchange 2010 seadis Microsoft järgmised eesmärgid: suurendada toote paindlikust ja stabiilsust, lihtsustada selle haldamist ning opereerida 10 GB suuruste postkastidega sama kiiresti kui Exchange 2007 opereeris 1 GB postkastidega. Samuti on muudetud tarkvara struktuuri – tagaserver Store vastutab andmete säilivuse eest ja eesserver Client Access Server töötleb kasutajate päringuid. Mõlemad saavad töötada ühel ja samal operatsioonisüsteemil. [36, 37] Exchange 2013 arendusel lisandusid uued eesmärgid [37]:

- toetada kasutajaid erinevatest põlvkondadest (esimene põlvkond alustas elektronposti kasutamist 1980ndatel, kuid tänapäeval alustanud põlvkonnal on teised ootused teenuselt);

- lihtsustada üleminek uuele veersioonile kasutajatel, kes on harjunud töötama vanemate versioonidega;
- täiustada ühilduvust toodetega Microsoft SharePoint 2013 ja Microsoft Lync 2013;
- järgida kasutajate vajadusi eelmistest versioonidest saadud kogemuste põhjal ja investeerida funktsioonidesse proportsionaalselt kasutajate huviga;
- tagada ühildavus Store ja Client Access Server erinevate versioonide vahel.

Erinevalt viimastest eelkäiatest Microsoft Exchange Server 2013 on saadaval kahes väljaandes: Standard ja Enterprise. Standard väljaanne on odavam, suunatud väikse arvuga postkastidega ettevõtetele, sisaldab kõiki e-postiserveri funktsioone ja selle postkastiandmebaasi arv on piiratud viiega. Väljaanne Enterprise võimaldab kasutada kuni 100 postkastiandmebaasi, mis on ka ainus erinevus Standard väljaandest.

Exchange 2013 on võimalik paigaldada operatsioonisüsteemidele Windows Server 2008 R2 ja Windows Server 2012. Alates Exchange 2000 versioonist on vajalik töötav Active Directory teenus. Exchange 2013 võib töötada koos kataloogiteenusega Active Directory ühel ja samal operatsioonisüsteemil, kuigi jõudluse ja turvalisuse eesmärgil arendaja soovib seda vältida. [37]

## **2.5. Kodukeskjaam**

Sidevahendid tõstavad töajõu efektiivsust ettevõtetes ja telefoniside on olnud üle sajandi tähtis sidevahend. Viimastel aastakümnetel on muutunud populaarseks telefoniside kasutamine üle IP andmesidevõrkude, kuna see tõstab konfiguratsioonivõimalusi ja mobiilsust nii kasutajatel kui ka teenusepakkujatel.

Paljudel ettevõtetel on olemas klienditoe telefon. Ei ole haruldane, kui kliendid soovivad suhelda mitme erineva osakonnaga, näiteks müügiosakond, arveldusosakond ja tehnilise toe osakond, kusjuures igas osakonnas võib olla mitu klienditeenindajat, kes vastavad kõnedele. Sellises olukorras tekib vajadus jaotada sissetulevaid telefonikõnesid nii osakondade kui ka nendes töötavate inimeste vahel ettevõttes kehtestatud reeglite järgi. Lisaks on tihti olemas soov pidada arvestust nii sissetulevate ja väljaminevate kõnede järel kui ka salvestada neid.

Eelnevalt nimetatud ülesannete lahendamiseks kasutatakse kodukeskjaamu ehk PBX (private branch exchange). Kodukeskjaama saab sisseosta teenusena, paigaldada riistvaralise

seadmena või seadistada tarkvaralise lahendusena. Järgnevas alampeatükis vaadelakse tarkvaralist kommertslahendust 3CX Phone System.

### 2.5.1. 3CX Phone System

3CX Phone System oli üks esimestest tarkvaralistest lahendustest, mis töötab operatsioonisüsteemi Windows keskkonnas. Tarkvara arendaja positsioneerib oma toodet kui lihtsat ja mugavat lahendust ning lubab, et selle paigaldamisega ja seadistamisega saavad hakkama kõik, kes omavad algteadmisi võrgutehnoloogias. Tarkvara koosneb järgmistest komponentidest [38]:

- 3CX Phone System koosneb mitmest Windows-i teenusest nagu näiteks andmebaas, SIP/RTP proksi, faks-teenus, kõneposti teenus, konverentskõnede teenus ja teised;
- 3CX Assistant võimaldab jälgida hetkeseisu ja seadistada tarkvara;
- 3CX Call Reporter lubab koostada statistikat raportite ja graafikute kujul;
- 3CX WebServer on veebiserver, mille abil töötab veebiliides.

3CX Phone System on saadaval kolmes väljaandes – Free, Standard ja Pro. Free väljaanne on täiesti tasuta ja sisaldab järgmisi piiranguid: kuni kahte samaaegset kõnet, konverentskõnede keeld, kõnede salvestuse keeld ja mitmed teised piirangud. Tasuliste väljaannete hind sõltub lubatud samaaegsetest kõnede arvust. Täpsema ülevaate annab allolev tabel.

<b>Samaaegsete kõnede arv</b>	<b>Standard väljaande hind</b>	<b>Pro väljaande hind</b>
4	395	535
8	750	1015
16	1095	1480
32	1950	2630
64	3895	5260
128	6995	9440
256	11995	16195
512	19995	26995
1024	34995	47245

Tabel 10. 3CX Phone System väljaannete hinnad eurodes. [39]

3CX Phone System Pro väljaanne sisaldab erinevaid vahendeid, mis teevad klientidega töö mugavamaks ka väiksematel ettevõtetel. Näiteks on võimalik integratsioon Microsoft Exchange lahendusega, mille tulemusel otsitakse kataloogist klientide nimesid telefoninumbri

järgi. Kõnejärjekordade võimalus on olemas ainult Pro väljaandel ja sellel on palju valiku võimalusi – vastaja valik võib olla juhuslik, järjestikune, vähima kõneaja järgi, vähima vastatud kõnede arvu järgi või pikema ooteaja järgi. [38, 40]

### **3. LAHENDUSED AVATUD LÄHTEKOODIGA TARKVARA NÄITEL**

#### **3.1. Marsruutimine**

Tänu interneti püsühenduse levikule on olemas lai valik nii tarkvara kui ka riistvara, mis võimaldab lahendada internetiühenduse jagamisega seotud probleeme. Tarkvara saab olla nii avatud kui ka suletud lähtekoodiga, riistvara aga nii spetsialiseeritud kui ka üldotstarbeline. Marsruutimise ülesannete lahendamiseks mõeldud avatud lähtekoodiga operatsioonisüsteemid on võimalik jagada kaheks: eriotstarbelised ja üldotstarbelised distributsioonid.

Eriotstarbelised distributsioonid sisaldavad eelnevalt paigaldatud ja seadistatud tarkvara (näiteks DHCP ja DNS serverid, erinevad VPN standardid ja statistika kogumise vahendid) ning graafilist kasutajaliidest, mis on tihti veebipõhine. Tuntumad näited on DD-WRT ja OpenWRT (Linux-põhised) ning m0n0wall ja pfSense (FreeBSD-põhised). Paljudel distributsioonidel on olemas paketi haldur, mis erilahenduse vajadusel võimaldab kiirelt ja mugavalt paigaldada vajalikku tarkvara.

Üldotstarbelise operatsioonisüsteemi kasutamiseks valitakse tavaliselt sellised distributsioonid, mida on võimalik paigaldada minimaalse tarkvarapaketi arvuga (näiteks ilma graafilise kasutajaliideseta ega kompilaatorita). Populaarsemad distributsioonid on Debian GNU/Linux, CentOS ja FreeBSD. Sõltuvalt vajadustest paigaldatakse vajalikud tarkvarapaketid, näiteks võrgusildade tugi, VLAN tugi ning VPN ja muud teenused, mille seadistamine teostatakse muutes parameetreid konfiguratsioonifailides.

##### **3.1.1. Marsruutimisprotokollid**

Suurema seadmete arvuga võrke on mõistlik jaotada alamvõrkudeks kasvõi levikupiirkonna vähendamise eesmärgil. Tulemusena tekib mitu alamvõrku, mille marsruutidest peavad olema teavitatud kõik marsruuterid. Väiksema arvu alamvõrkude korral saab seadistada kõiki marsruutereid käsitsi, suurema arvu korral on kasulik kasutada marsruutimisprotokolle. Ülevaate kolme tarkvarapaketi BIRD, Quagga ja XORP marsruutimisprotokollide toest annab järgnev tabel.

Protokoll	BIRD	Quagga	XORP
BFG (tehnoloogia)	jah	jah	ei
BGP	jah	jah	jah
ISIS	ei	jah	ei
MPLS (tehnoloogia)	ei	jah	ei
OSPFv2	jah	jah	jah
OSPFv3 (IPv6)	jah	jah	jah
RIPv1	ei	jah	ei
RIPv2	jah	jah	jah
RIPng (IPv6)	jah	jah	jah

Tabel 10. BIRD, Quagga ja XORP marsruutimisprotokollide tugi [41, 42, 43].

### 3.1.2. Läbilaskevõime hindamine SSL-tüüpi virtuaalsetes privaativõrkudes (VPN)

Sageli valitakse marsruuteriks spetsialiseeritud kommertslahendus põhjusel, kuna see sisaldab riistvaralist krüpteerimisseadet, mis tõstab krüpteeritud ühendustel läbilaskevõimet. Vaatamata sellele nii kaasaegsed üldotstarbelised protsessorid kui ka nende energiasäästlikud versioonid suudavad konkureerida kommertsruuteritele nii läbilaskevõimekt kui ka hinnalt.

Järgnevad katsed on tehtud tingimustes, kus 10.10.0.0/24 ja 10.255.0.0/30 on füüsilised kohtvõrgud, mille vahele on paigaldatud marsruuter (mõlemad võrgud ühendatud eraldi 1 Gbps võrguliidestega). Võrgu 10.255.0.0/30 kaudu on tekitatud VPN (OpenVPN abil, UDP põhiselt), mille aadressiruum on 192.168.0.0/24. Marsruuter on seadistatud üldotstarbelisel arvutil, mille keskprotsessor on Intel Atom D2500 (taktsagedus 1.86 GHz, kaks füüsilist tuuma ja kokku kaks loogilist tuuma, 10 W nõutav jahutusvõimsus, 2011. aasta tehnoloogia) ja selle hind on vähem kui 150 eurot. Läbilaskevõimet mõõdetud vahendiga netio ja lisaks on märgitud ka keskmine keskprotsessori koormus marsruuteril katse ajal. Kokku on tehtud seitse katset, kõik võrgust 10.10.0.0/24, vaadeldud ka erinevaid pakettide suurusi ning saatmine/krüpteerimine (↑) ja vastuvõtt/dekrüpteerimine (↓) eraldi:

1. otseside marsruuteriga võrgus 10.10.0.0/24;
2. marsruutimine võrku 10.255.0.0/30;
3. marsruutimine ja võrguaadressi transleerimine (NAT) võrku 10.255.0.0/30;
4. marsruutimine VPN võrku 192.168.0.0/24 Blowfish-CBC krüpteeringuga;
5. marsruutimine VPN võrku 192.168.0.0/24 AES-128-CBC krüpteeringuga;
6. marsruutimine VPN võrku 192.168.0.0/24 AES-256-CBC krüpteeringuga;
7. marsruutimine VPN võrku 192.168.0.0/24 DES-EDE3-CBC krüpteeringuga.

	<b>1 KB paketid</b>	<b>2 KB paketid</b>	<b>4 KB paketid</b>	<b>8 KB paketid</b>	<b>16 KB paketid</b>	<b>32 KB paketid</b>	<b>CPU koormus</b>
Otseside	↑110.32 ↓82.70	↑110.71 ↓86.91	↑110.10 ↓111.91	↑110.64 ↓112.06	↑110.69 ↓112.00	↑110.48 ↓112.01	~8%
Marsruutimine	↑96.95 ↓96.95	↑97.34 ↓95.59	↑97.84 ↓95.93	↑97.58 ↓95.55	↑97.70 ↓95.83	↑98.51 ↓97.46	~2%
Marsruutimine ja NAT	↑96.38 ↓94.85	↑97.27 ↓95.22	↑97.14 ↓95.30	↑97.35 ↓95.97	↑97.17 ↓95.51	↑97.90 ↓96.92	~6%
Marsruutimine ja VPN Blowfish 128 bit	↑12.94 ↓64.15	↑11.85 ↓65.48	↑12.16 ↓65.16	↑12.26 ↓64.32	↑12.26 ↓65.35	↑12.30 ↓63.87	~50%
Marsruutimine ja VPN AES 128 bit	↑11.66 ↓63.73	↑10.36 ↓65.37	↑10.57 ↓65.98	↑10.55 ↓65.76	↑10.55 ↓65.46	↑10.56 ↓63.88	~50%
Marsruutimine ja VPN AES 256 bit	↑10.36 ↓64.55	↑9.10 ↓64.52	↑9.27 ↓64.70	↑9.27 ↓66.10	↑9.25 ↓66.05	↑9.29 ↓64.01	~50%
Marsruutimine ja VPN 3DES 192 bit	↑5.71 ↓61.44	↑4.66 ↓60.16	↑4.71 ↓59.47	↑4.70 ↓59.15	↑4.71 ↓59.67	↑4.71 ↓58.46	~50%

Tabel 12. Intel Atom D2500 võrguliikluse läbilaskevõime megabaitides sekundis.

Tabelist 12 on näha, et marsruutimise kiirused, sealhulgas koos võrguaadressi transleerimisega, lähenevad võrguliidese kiirusele ja keskprotsessori kasutavus on minimaalne. Otseside korral on keskprotsessori koormus suurem, sest ainult selles katses käivitati netio vahendi kuular just marsruuteril (teistes katsetes võrkudes 10.255.0.0/30 ja 192.168.0.0/24), mis vajab jõudlust võrguliikluse tekitamiseks ja mõõtmiseks. Krüpteeritud ühenduste korral on näha, et krüpteerimine nõuab 5-10 korda rohkem jõudlust kui dekrüpteerimine. Ligi 50 protsendine keskprotsessori koormus on seotud sellega, et OpenVPN kasutab ainult ühte keskprotsessori tuuma. OpenVPN toetab kõiki riistvaralisi krüpteerimisseadmeid, mida toetab OpenSSL - seega on võimalik ka energiasäästliku protsessoriga marsruuteri abil luua suurema läbilaskevõimega krüpteeritud virtuaalseid privaattõrke.



### 3.1.3. Näiteid Debian GNU/Linux distributsioonil

#### 3.1.3.1. Virtuaalkohtvõrkude, virtuaalvõrguliideste ja võrgusildade haldamine

Virtuaalkohtvõrgutehnoloogia ehk VLAN, mis tugineb protokollil 802.1Q, võimaldab loogiliselt jagada ühte füüsilist võrku mitmeks virtuaalseks. Nimetatud tehnoloogia kasutamise peamised põhjused on ARP-pakettide levikupiirkonna vähendamine ja üldise turvalisuse tõstmine. Samuti võimaldab VLAN tehnoloogia hoida kokku füüsiliste ühenduste ja võrguseadmete peal. Näiteks võib tuua olukorda, kui on olemas VLAN tehnoloogiat toetav hallatav kommutaator ja on vaja seadistada marsruuter riistvaral, millel on ainult üks võrguliides. Olgu võrguliidese nimetus eth0 ja laivõrgule eraldatud virtuaalkohtvõrgu ID on 10. Virtuaalkohtvõrkude haldamiseks on vaja paigaldada pakett vlan [44].

```
vconfig add eth0 10
```

Näide 1. Virtuaalkohtvõrgu lisamine käsurealt [44].

Virtuaalvõrguliideseid kasutatakse peamiselt virtuaalsetes privaatvõrkudes ehk VPN. UNIX'ilaadsetes operatsioonisüsteemides esineb kaks virtuaalvõrguliidest: TUN (marsruuditud võrkudeks) ja TAP (sillatud võrkudeks). Olukorras, kui on vaja ehitada kahte OSI mudeli teise kihi VPN-i sise- ja välisvõrguga, kasutatakse vahendit tunctl virtuaalvõrguliideste tap0 ja tap1 loomiseks. Virtuaalvõrguliideste seadistamiseks on vaja paigaldada pakett uml-utilities.

```
tunctl -t tap0  
tunctl -t tap1
```

Näide 2. OSI mudeli teise kihi virtuaalvõrguliideste lisamine käsurealt [45].

Lisaks võrkude ühendamisele võimaldab sildade kasutamine lihtsustada marsruuteri konfiguratsiooni. Tavaliselt sillatakse traadiga ja traadita sisevõrguliideseid, kuid silda saab kasutada ka laivõrgu liidestel, mis tekitab vabaduse vahetada võrguliideseid silla sees ilma muu tarkvara konfiguratsiooni muutmata. Selliselt olgu br0 sisevõrgu 10.10.0.0/24 sild ja br1 laivõrgu 213.180.31.64/28 sild, kurjuures sillaga br0 on ühendatud kohtvõrguliides eth0 ja virtuaalliides tap0 ning sillaga br1 on ühendatud kohtvõrguliides eth0 virtuaalkohtvõrguga 10 ja virtuaalliides tap1 (Näide 3). Võrgusildade kasutamiseks on vaja paigaldada pakett birge-utils.

```
brctl add br0
brctl addif br0 eth0
brctl addif br0 tap0
ifconfig br0 213.180.31.77/28 up
brctl add br1
brctl addif br1 eth0.10
brctl addif br1 tap1
ifconfig br0 10.10.0.1/24 up
```

Näide 3. Võrgusildade lisamine ja seadistamine käsurealt [46].

Distributsioonis Debian GNU/Linux on võimalik kõik eelnevalt sisestatud parameetrid kirjeldada konfiguratsioonifailis /etc/network/interfaces järgmisel kujul.

```
auto eth0
iface eth0 inet manual

iface eth0.10 inet manual
    vlan_raw_device eth0

auto tap0
iface tap0 inet manual
    pre-up tuncctl -t tap0
    up ifconfig tap0 up
    down ifconfig tap0 down

auto tap1
iface tap1 inet manual
    pre-up tuncctl -t tap1
    up ifconfig tap1 up
    down ifconfig tap1 down

auto br0
iface br0 inet static
    address 10.10.0.1
    netmask 255.255.255.0
    bridge_ports eth0 tap0

auto br1
iface br1 inet static
    address 213.180.31.77
    netmask 255.255.255.240
    gateway 213.180.31.65
    bridge_ports eth0.10 tap1
```

Näide 4. Virtuaalkohtuvõrkude, virtuaalvõrguliidest ja sildade konfiguratsioon failis /etc/network/interfaces [44, 45, 46].

### 3.1.3.2. Võrguliikluse suunamine liidest vahel, tulemüür ja NAT

Iga marsruuteri peamine ülesanne on suunata võrguliiklust erinevate liidest vahel, olgu need füüsilised või virtuaalsed. Selleks tuleb muuta vastav operatsioonisüsteemi tuuma parameeter ja kirje konfiguratsioonifailis.

```
sysctl net.ipv4.ip_forward=1
sed -i '/net.ipv4.ip_forward/c\net.ipv4.ip_forward=1' /etc/sysctl.conf
```

Näide 5. Võrguliikluse suunamise lubamine käsurealt ja konfiguratsioonifailis [47].

Võrguliikluse piiramist (tulemüür), suunamist ja transleerimist (NAT) seadistab iptables nimeline vahend. Järgnevaga lubame sissetulevad ühendused ainult TCP pordile 22 (SSH teenus) ja keelame suunatavad paketid portide vahemikus 6881-6889 (Näide 6).

```
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
iptables -P INPUT DROP
iptables -A FORWARD -p tcp --dport 6881:6889 -j DROP
```

Näide 6. Kõikide sissetulevate ühenduste keelamine välja arvatud SSH ja suunatavate pakettide keelamine TCP portide vahemikus 6881-6889 [48].

Eeldusel, et välise silla br1 IP aadress on 213.180.31.77 ja sild br0 on ühendatud sisevõrguga 10.10.0.0/24, lubame sisevõrgus NAT-i, suuname TCP pordi 25 aadressile 10.10.0.104 ning suuname veebiliikluse sisevõrgust kohalikule TCP pordile 3128, millel võiks töötada HTTP proksi.

```
iptables -t nat -A POSTROUTING -s 10.10.0.0/24 -o br1 -j SNAT --to-source 213.180.31.77
iptables -t nat -A PREROUTING -d 213.180.31.77/32 -p tcp -m tcp --dport 25 -j DNAT --to-destination 10.10.0.104:25
iptables -t nat -A PREROUTING -s 10.10.0.0/24 -p tcp --dport 80 -j REDIRECT --to-port 3128
```

Näide 7. NAT lubamine, SMTP teenuse suunamine sisevõrgu aadressile ja veebiliikluse suunamine veebiproksile [48].

Eelnevalt sisestatud iptables parameetrid tühistuvad taaskäivitusel ning nende salvestamiseks ja taastamiseks on olemas vahendid iptables-save ja iptables-restore vastavalt. Parameetrite salvestamiseks sisestame järgneva käsu.

```
iptables-save > /etc/iptables.conf
```

Näide 8. Iptables parameetrite salvestamine konfiguratsioonifaili [49].

Eelnevalt salvestatud parameetrite taastamiseks saab sisestada järgneva käsu või seadistada see automaatselt käivitatava skriptina.

```
iptables-restore < /etc/iptables.conf
```

Näide 9. Iptables parameetrite taastamine konfiguratsioonifailist [49].

### 3.1.3.3. Nimeserver (DNS)

Kohaliku nimeserveri kasutamisel on mitmeid eeliseid: vastused sellelt on kiiremad, selle abil on võimalik jälgida tehtud nimepäringuid, keelata domeene, muuta nimedele vastavaid aadresse ja seadistada vahemälu kasutamise reegleid. UNIX'i-laadsetel operatsioonisüsteemidel populaarseim nimeserveritarkvara on BIND, mille 9. versiooni saab paigaldada operatsioonisüsteemile Debian GNU/Linux paketi bind9. Lahendus on kasutatav ka vaikimisi seadetega. Olukorras, kus on vaja muuta kohalikele kasutajatele domeeninimele d-sys.org vastavaid IP-aadresse (näiteks suunata päringud e-postiserverile mail.d-sys.org sisevõrgu IP-aadressile) lisatakse järgmised read konfiguratsioonifaili /etc/bind/named.conf.local eeldusel, et /etc/bind/zones/d-sys.org-local.db on eelnevalt seadistatud tsoonifail.

```
acl "local"      { 127.0.0.1; 10.10.0.0/24; };
zone "d-sys.org" { type master; file "/etc/bind/zones/d-sys.org-local.db"; allow-query { local; }; }
```

Näide 10. Ainult sisevõrgust kättesaadava tsooni lisamine nimeservertarkvaras BIND [50].

### 3.1.3.4. Dünaamilise hostikonfiguratsiooni protokoll (DHCP)

Suure tõenäosusega on vaja sisevõrgus jagada IP-aadresse, kusjuures mõnedele seadetele staatilisi, teistele dünaamilisi. Probleemi lahendamiseks aastaid tõestanud ennast ISC DHCP Server, mida on võimalik paigaldada paketi isc-dhcp-server. Sisevõrgu 10.10.0.0/24 jaoks sobib järgmine näide, milles e-postiserverile on antud staatiline IP ja on loodud dünaamiliste IP aadressite vahemik.

```
ddns-update-style none;
default-lease-time 3600;
max-lease-time 7200;
authoritative;

subnet 10.10.0.0 netmask 255.255.255.240 {
    option routers 10.10.0.1;
    option domain-name-servers 10.10.0.1;
    option domain-name "ds.local";

    group {
        host mail { hardware ethernet 00:1A:2B:3C:4D:5E; fixed-address 10.10.0.104; }
    }
    pool {
        allow unknown-clients;
        range 10.10.0.151 10.10.0.250;
    }
}
```

Näide 11. Tarkvara ISC DHCP Server konfiguratsioon, milles esineb üks staatiline kirje ja dünaamiliste IP-aadressite vahemik [51].

## 3.2. Riistvara virtualiseerimine

Alates riistvara virtualiseerimise toega x86 arhitektuuril protsessorite turule tulekust on loodud mitmeid nii tasulisi kui ka avatud lähtekoodiga vahendeid, mille abil on võimalik lahendada riistvara virtualiseerimisega seotud probleeme. Tasulise tarkvara tootjad on positsioneerinud oma lahendusi kui mugavalt hallatavaid, töökindlaid ja paljude lisavõimalustega. Kuigi avatud lähtekoodiga lahendustele on raske omistada turuliidrite positsioone, nende arengus on toimunud märkimisväärseid edusamme.

Avatud lähtekoodiga riistvara virtualiseerimise serveritele mõeldud lahendusi jagatakse kaheks üksteisest sõltumaks komponendiks: virtuaalmasinahaldur ja juhtimistarkvara. Tuntumad virtuaalmasinahaldurid on KVM ja Xen, populaarsemad juhtimistarkvarad on Libvirt ja oVirt.

### 3.2.1. KVM

KVM (Kernel-based Virtual Machine) on hüperviisor, mis koosneb täiendusest Linux tuumale (annab ligipääsu virtualiseerimisressursidele operatsioonisüsteemi kasutajaruumist) ja rakendusest (täiendatud QEMU emulaator, millele on lisatud x86 virtualiseerimistehnoloogia tugi). KVM arendust alustas ettevõtte Qumranet ja esimene väljalase ilmus aastal 2007 koos Linux versiooniga 2.6.20. Aastal 2008 ostis RedHat Qumranet koos projektiga KVM [52].

KVM suudab emuleerida erinevat riistvara [53]:

- protsessoreid (alates Intel 486 kuni Sandy Bridge mikroarhitektuurini);
- kettakontrollereid (IDE, AHCI, SCSI);
- graafikaadptereid (Cirrus Logic GD5446, VGA koos VBE-ga ja VMWare SVGA-II);
- võrguadptereid (NE2000 PCI, Intel PRO/100 ja PRO/1000 seeriad, Realtek RTL8139, AMD Am79C970A);
- helikaarte (Creative Sound Blaster 16, Crystal CS4231A, Yamaha YM3812, Gravis Ultrasound GF1, Intel 82801AA AC97 Audio, ENSONIQ AudioPCI ES1370, Intel HD Audio);
- USB (UHCI, EHCI, jaotur ja mitmed seadmed);
- jada- ja rööpporte.

Samuti toetab KVM järgmisi kettakujutisvorminguid [53]:

- toorandmed;
- kohalike kettaid – kõvakettaid, optilisi ja floppi kettaid;
- cow, qcow ja qcow2 – copy-on-write erinevaid versioone;
- vdi – Tarkvara VirtualBox kettakujutisvormingut;
- vmdk – VMWare toodetes kasutatav kettakujutisvormingut;
- vpc – Microsoft VirtualPC kettakujutisvormingut;
- parallels – Parallels toodete kettakujutisvormingut;
- Bochs emulaatori kettakujutisvormingut;
- iSCSI tüüpi võrguressurse.
- erinevaid võrguprotokolle (TFTP, FTP, FTPS, HTTP, HTTPS)

Kuna virtuaalmasinahaldur toetab kettakujutist toorandmete kujul, siis on lihtne muuta füüsilisi masinaid virtuaalseteks (P2V) ja vastupidi (V2P), lisaks on kerge monteerida kettakujutistes olevaid partitsioone kohalikku tööpunkti. Sõltumata kettakujutise vormingust

on võimalik hoida seda kolmel erineval tasemel: failina failisüsteemil, füüsilise ketta jaotusel või otse füüsilisel kettal.

KVM ei sisalda vahendeid, mis automatiseeriks käivitust ega haldust, seega käivitatakse virtuaalmasinad käsurealt andes ette parameetritena virtuaalse riistvara konfiguratsiooni (näide 12).

```
/usr/bin/kvm -S -M pc-1.1 -enable-kvm -m 256 -smp 1,sockets=1,cores=1,threads=1 -name MAIL \  
-uuid f5b8c05b-9c7a-3211-49b9-2bd635ea0004 -no-user-config -nodefaults \  
-chardev socket,id=charmonitor,path=/var/lib/libvirt/qemu/MAIL.monitor,server,nowait \  
-mon chardev=charmonitor,id=monitor,mode=control -rtc base=localtime -no-shutdown -boot order=c,menu=on \  
-device ahci,id=ahci0,bus=pci.0,addr=0x9 -device piix3-usb-uhci,id=usb,bus=pci.0,addr=0x1.0x2 \  
-drive file=/data/vm/mail.img,if=none,id=drive-sata0-0-0,format=raw \  
-device ide-hd,bus=ahci0.0,drive=drive-sata0-0-0,id=sata0-0-0 \  
-drive if=none,id=drive-ide0-1-0,readonly=on,format=raw \  
-device ide-cd,bus=ide.1,unit=0,drive=drive-ide0-1-0,id=ide0-1-0 \  
-netdev tap,fd=20,id=hostnet0 -device e1000,netdev=hostnet0,id=net0,mac=ea:00:00:00:04,bus=pci.0,addr=0x3 \  
-vnc 0.0.0.0:4,password -vga cirrus -device virtio-balloon-pci,id=balloon0,bus=pci.0,addr=0x8
```

Näide 12. Virtuaalmasina koos virtuaalse riistvara konfiguratsiooniga käivitamine KVM abil käsurealt [53].

Töötavaid virtuaalmasinaid on võimalik hallata (peatada, taaskäivitada, muuta virtuaalse riistavara konfiguratsiooni) läbi virtuaalse terminali, kuid lihtsam ja efektiivsem on kasutada haldustarkvara.

### 3.2.2. Haldustarkvara libvirt

Tarkvarakogu libvirt on hüperviisorite haldustarkvara, mis koosneb järgmistest komponentidest: API teegid, teenus libvirtd ja käsurida virsh. Tarkvara arendus algas aastal 2005 ettevõtte RedHat initsiatiivil ning peamine eesmärk oli luua vahend, mis võimaldaks juhtida erinevaid virtuaalmasinahaldureid ühiste käskudega. Versioon 1.2.2 toetab järgmisi hüperviisoreid, emulaatoreid [54]:

- KVM ja QEMU;
- Xen;
- LXC (Linux operatsioonisüsteemi virtualiseerimine);
- OpenVZ (Linux operatsioonisüsteemi virtualiseerimine);
- User-mode Linux (Linux operatsioonisüsteemi virtualiseerimine);
- Oracle VirtualBox;
- VMware GSX, ESX, ESXi, Workstation, ja Player;
- Microsoft Hyper-V;
- IBM PowerVM;
- Parallels;
- Bhyve.

Tarkvara libvirt toetab kasutajaõiguste määramist ja autentimismehhanisme Kerberos ja SASL [55]. Virtuaalmasinate konfiguratsiooni salvestatakse XML-vormingus failide kujul (Näide 13).

```
<domain type='kvm'>
  <name>MAIL</name>
  <uuid>f5b8c05b-9c7a-3211-49b9-2bd635ea0004</uuid>
  <memory unit='KiB'>262144</memory>
  <currentMemory unit='KiB'>262144</currentMemory>
  <vcpu placement='static'>1</vcpu>
  <os>
    <type arch='x86_64' machine='pc-1.1'>hvm</type>
    <boot dev='hd' />
    <bootmenu enable='yes' />
  </os>
  <features>
    <acpi />
  </features>
  <clock offset='localtime' />
  <on_poweroff>destroy</on_poweroff>
  <on_reboot>restart</on_reboot>
  <on_crash>destroy</on_crash>
  <devices>
    <emulator>/usr/bin/kvm</emulator>
    <disk type='block' device='disk'>
      <driver name='qemu' type='raw' />
      <source dev='/data/vm/mail.img' />
      <target dev='sda' bus='sata' />
      <address type='drive' controller='0' bus='0' target='0' unit='0' />
    </disk>
    <disk type='block' device='cdrom'>
      <driver name='qemu' type='raw' />
      <target dev='hdc' bus='ide' />
      <readonly />
      <address type='drive' controller='0' bus='1' target='0' unit='0' />
    </disk>
    <controller type='ide' index='0'>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x01' function='0x1' />
    </controller>
    <controller type='sata' index='0'>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x09' function='0x0' />
    </controller>
    <controller type='usb' index='0'>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x01' function='0x2' />
    </controller>
    <interface type='bridge'>
      <mac address='ea:00:00:00:04' />
      <source bridge='br0' />
      <model type='e1000' />
      <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0' />
    </interface>
    <input type='mouse' bus='ps2' />
    <graphics type='vnc' port='5904' autoport='no' listen='0.0.0.0'>
      <listen type='address' address='0.0.0.0' />
    </graphics>
    <video>
      <model type='cirrus' vram='9216' heads='1' />
      <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x0' />
    </video>
    <memballoon model='virtio'>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x08' function='0x0' />
    </memballoon>
  </devices>
</domain>
```

Näide 13. Virtuaalmasina konfiguratsioon tarkvaras libvirt XML kujul [56].

Üksuste haldamine käsurealt ei ole kõikides olukordades mugav, mis tõttu on loodud mitmeid graafilise kasutajaliidesega abivahendeid: virt-manager, virt-viewer, VM Manager (Android)

ja oVirt (veebipõhine). Taskvara libvirt on integreeritud paljudesse teistesse lahendustesse näiteks pilvandmetöötlusplatvormidesse OpenStack ja Eucalyptus. [57]

### 3.3. Veebiserver

Kaasaegsed veebimajutused võimaldavad hoida üleval ettevõtte kodulehekülge väga väikeste ülalpidamiskuludega, kuid omavad teatud nõrkusi:

- puudub võimalus paigaldada lisatarkvara;
- ressursid (näiteks arvutusjõudlus ja ligipääs kettale) on jagatud teiste kasutajatega;
- piiratud andmeedastusmaht ja kettamaht;
- võimalik juurdepääs kolmandatel isikutel.

Piirangute vähendamiseks on võimalik vahetada veebimajutuse paketti, tellida asemele virtuaalserver või viia üle majutus oma serverile. Juba renditud virtuaalserveril tekib vabadus valida veebiserveri tarkvara ja selle lisasid. Tasuta ja avatud lähtekoodiga veebiservereid on märkimisväärne valik, milles populaarseimad on Apache HTTP Server, Nginx, lighthttpd ja Apache Tomcat.

Seoses internetikasutajate arvu märkimisväärse kasvuga on muutunud tavaliseks nähtus, kui ühel veebilehel on nii palju külastajaid, et üks veebiserver ei ole võimeline õigeaegselt töötleva kõiki päringuid. Nimetatud probleemil on kaks lahendussuunda – vertikaalne ja horisontaalne skaleerimine. Vertikaalne skaleerimine tähendab riistvara täiendamist või selle vahetamist võimsama vastu. Horisontaalse skaleerimise all mõeldakse süsteemi laiendamist mitmele sõlmele. Mõned horisontaalse skaleerimise eelised on näiteks töökindluse tõusmine ja skaleerimise käigus süsteemi seiskamise vajaduse puudumine.

Veebiserveri horisontaalse skaleerimise tulemust nimetatakse veebiklastriks. Veebiklaster koosneb pöördproksist (eeskomponent) ja sõlmedest veebiserverite kujul (tagakomponendid). Klastril töötava veebilehe külastajad saavad päringuid pöördproksile, mis edastab need ühele sõlmedest, kusjuures sõlmede valimine toimub kindlate reeglite järgi ning kindla sessiooni ühendused luuakse alati ühe ja sama sõlmega. Lisaks koormuse tasakaalustamisele täidab pöördproksi ka teisi ülesandeid: staatilise sisu puhverdamine ja pakkimine, SSL krüpteerimine (ka riistvaraliselt), tegelike veebiserverite varjamine turvalisuse eesmärgil. Mõned populaarsetest avatud lähtekoodiga tarkvaralistest lahendustest, mis suudavad täita pöördproksi funktsiooni, on Nginx, Squid ja Apache Traffic Server.



Kuna pöördproksi peab töötleva kõiki sissetulevaid ja väljaminevaid päringuid, siis on kasulik valida minimaalse jalajäljega kiire tarkvara. Nimetatud omadusi omab Nginx. Veebiserveriks valitakse tarkvara, millel on olemas vajalike programmeerimiskeelte ja lisamoodulite võimalus, samuti on kasuks laiad konfiguratsioonivõimalused. Kirjeldatud funktsioonid esinevad tarkvaras Apache HTTP Server. Järgnevas alampeatükis tutvustatakse Nginx ja Apache HTTP Server abil veebiklastri seadistamist.

### **3.3.1. Veebiklastri seadistamine Nginx ja Apache HTTP Server abil**

Enamikele Linux distributsioonidele saab paigaldada Apache HTTP Serveri ühe käsuga, kusjuures pakettide allalaadimine ja paigaldus kestevad alla ühe minuti, mille järel käivitub veebiserver. Üldjuhul on vaikimisi töödeldavaks kataloogiks `/var/www`. Lihtsamal stsenaariumis saab eelnevalt nimetatud kataloogi paigutada veebilehe sisu, ja veebiserver on valmis külastajate teenindamiseks. Tavaststsenaariumis tuleb paigaldada töödeldava veebilehe, sisuhaldussüsteemi või muu tarkvara poolt kasutatavate programmeerimiskeelte toed (näiteks PHP, Perl ja Python) ja lisavõimaluste pakettid (näiteks MySQL, IMAP ja LDAP toed). Leidub veebilehti, mis nõuavad veebiserveri mõnede vaikimisi paremeetrite muutmist, näiteks üleslaetava faili suuruse piirangut. Ei ole haruldane, kui üks ja sama veebiserver peab teenindama mitu erinevat veebilehte mitmel domeenil. Nimetatud probleemi lahenduseks on virtuaalhosting, mis võimaldab samal IP-aadressil teenindada mitu domeeni erineva konfiguratsiooniga, sealhulgas töödeldavate kataloogidega.

Kuna veebiklastris on rohkem kui üks sõlm (veebiserver), siis tekib vajadus automatiseerida või lihtsustada sõlmede seadistamist. Paljud paketi haldurid lubavad paigaldada mitu paketti ühe käsuga. Apache HTTP Serveri konfiguratsioonifailide muutmine on automatiseeritav kas ühe rea kaupa või terve faili asenduse kujul. Nimetatud viisidel on seadistatav ka virtuaalhostingu konfiguratsioon.

Omaette küsimust moodustab sisu paigutus. Kui tegemist on sisuga, millel ei esine muutusi, ja igas sõlmes saab sisu hoida eraldi andmekandjal, siis on kasulik paigutada sisu kohalikel andmekandjatel. Suure tõenäosusega on sisu muutuv, seega tuleb see paigutada võrguressursile (näiteks NFS või iSCSI). Analüüsides sisu on lihtne tuvastada muutuvaid ja muutumatuid sisu komponente – vastavalt sellele on võimalik kombineerida eelnevalt kirjeldatud sisu paigutusviise.

### 3.3.1.1. Sõlmede määramine

Veebiklastris on tähtis roll pöördproksil kui eeskomponendil, kuna see jaotab kõik külastajate päringud sõlmede vahel ja edastab vastused külastajatele. Pöördproksi ei pea omama lisamooduleid, mida kasutab veebileht, kuid virtuaalhostingu kasutamisel tuleb seadistada see ka pöördproksil. Selleks tuleb pöördproksi ja sõlmede tasemel jaotada veebilehed kas portide või nimede alusel. Jaotuses portide abil seadistatakse kindel veebileht ühele ja samale portile kõikidel sõlmedel. Nimepõhises jaotuses seadistatakse sõlmedel virtuaalhosting nii, et kindlal veebilehel on kindel nimi, kuid sõlmede gruppeerimise eesmärgil on nime ees metamärk. Nimelahenduseks pöördproksil võib kasutada nii hostifaili (/etc/hosts) kui ka võrgusisest DNS teenust. Eeldades, et infrastruktuuris on sõlmele seadistatud nimelahendus, järgnevad read seadistavad Nginx tarkvara ülesvoolu nimega NODES\_d-sys, mis kasutab sõlmi nimedega \*.d-sys.webnode.local (Näide 14).

```
upstream NODES_d-sys {
    server 01.d-sys.webnode.local:80;
    server 02.d-sys.webnode.local:80;
    server 03.d-sys.webnode.local:80;
    server 04.d-sys.webnode.local:80;
}
```

Näide 14. Ülesvoolu konfiguratsioon tarkvaras Nginx [58].

Eelnevalt määratud ülesvoolu on võimalik kasutada terves Nginx konfiguratsioonis.

### 3.3.1.2. Suunamine turvatud ühendusele

Juhul, kui on vaja kasutada turvasoklite kihiga kaitsud ühendusi, siis on mõistlik korraldada suunamine nendele just pöördproksil. Järgnevad read korraldavad suunamist HTTP ühenduselt HTTPS ühendusele (Näide 15).

```
server {
    listen 213.180.31.77:80;
    server_name      d-sys.org www.d-sys.org;

    if ($host !~* ^www\.){
        rewrite      ^ https://www.$server_name$request_uri? permanent;
    }
    if ($host ~* ^www\.){
        rewrite      ^ https://$server_name$request_uri? permanent;
    }
}
```

Näide 15. Veebiliikluse suunamine HTTPS ühendusele tarkvara Nginx mooduli rewrite abil [59].

Suunamisel on kasutatud tingimusi, mis vajadusel lisavad domeeninime ette www.

### 3.3.1.3. Vahemälu kasutamine

Sõlmede koormuse vähendamiseks on kasulik seadistada vahemälu pöördproksil. Väiksema koormusega klastritel saab selleks kasutada monteeritud kõvakettajaotust, suuremate koormuste korral saab viimast asendada operatiivmälu. Järgnevad read defineerivad vahemälu nimega ram, mis kasutab kataloogi /mnt/ramdisk ja mille andmemahu piiranguks on 1024 MB (Näide 16).

```
proxy_buffering on;
proxy_cache_path /mnt/ramdisk levels=1:2 keys_zone=ram:32m inactive=7d max_size=1024m;
```

Näide 16. Vahemälu defineerimine tarkvaras Nginx [60].

Edaspidiselt on võimalik kasutada ram-nimelist vahemälu kõikidel virtuaalhostidel.

### 3.3.1.4. Virtuaalhosti määramine

Sarnaselt tarkvaraga Apache HTTP Server seadistatakse tarkvaras Nginx SSL ühenduse parameetrid nagu sertifikaadid ja šifrid virtuaalhosti konfiguratsioonis. Sõltuvalt sõlmedel töötavast tarkvarast on kasulik päistes edastada sõlmedele kliendi IP-aadress ja vajadusel mõned muud parameetrid. Eeldades, et on olemas töötav ülesvool nimega NODES\_d-sys, pöördproksi peab vajadusel lisama www domeeninime algusesse, kasutama vahemälu ram ja lisama päistesse mõned parameetrid on võimalik seadistada turvasoklite kihiga kaitstud virtuaalhosti tarkvaras Nginx järgnevalt (Näide 17).

```
server {
    listen 213.180.31.77:443;
    server_name      d-sys.org d-sys.org;

    if ($host !~* ^www\.) {
        rewrite      ^ https://www.$server_name$request_uri? permanent;
    }

    client_max_body_size 32M;

    ssl on;
    ssl_certificate /etc/cert/www_d-sys_org.crt;
    ssl_certificate_key /etc/cert/www_d-sys_org.key;

    ssl_session_timeout 5m;

    ssl_protocols SSLv3 TLSv1;
    ssl_ciphers ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv3:+EXP;
    ssl_prefer_server_ciphers on;

    location / {
        proxy_pass      http://NODES_d-sys;
        proxy_cache      ram;
        proxy_cache_valid 10m;
        proxy_set_header Host                $host;
        proxy_set_header X-Real-IP          $remote_addr;
        proxy_set_header X-Forwarded-For    $proxy_add_x_forwarded_for;
        proxy_set_header HTTPS              1;
    }
}
```

Näide 17. Ülesvooluga ja vahemäluga HTTPS virtuaalhosti konfiguratsioon tarkvaras Nginx [59, 61].

Ei ole välistatud, et üks pöördproksi ei ole võimeline teenindama kõiki külastajaid. Horisontaalsel laienemisel lisatakse pöördproksiservereid ning neid saab tasakaalustada jagades nende vahel sisu (tekst, pildid, video) või seadistades nende IP-aadressite kordamööda vahelduvust DNS tasandil. Lahendusi on võimalik kombineerida.

### **3.4. E-postiserver**

Paljud elektronposti töötlemiseks mõeldud kommertsilahendused (näiteks Microsoft Exchange ja IBM Notes/Domino) sisaldavad grupitöö korraldamiseks mõeldud vahendeid (näiteks kontaktide andmebaas, ühine kalender koos ülesannetega ja teised). Kuigi eksisteerivad konkurentsivõimelised avatud lähtekoodiga lahendused, ei ole kommertsilahenduste autentimismehhanismid alati ühilduvad asutustes kasutuses olevatega. Samuti võivad puududa kliendid mobiilsetele platvormidele või lahenduste veebileidesed ei ole piisavalt mugavad mobiilsetel seadmetel kasutamiseks.

Asutusel võib olla mitu domeeni, millel peab töötama e-post, ja lisaks töötaja e-postiaadressitele võivad kasutuses olla üldised (näiteks info, arved, müük või hooldus). Üldised e-postiaadressid võivad olla suunatud kas samanimelistele postikastidele või töötaja nimelistele postikastidele. Nimetatud olukord tähendab, et on vaja lahendust, mis lubaks dünaamiliselt defineerida reegleid. Ühilduvuse tõstmiseks on järgnes alampeatükis vaadeldud lahendus, mis töötab avatud lähtekoodiga tarkvaral ja kasutab IETF (Internet Engineering Task Force) poolt RFC (Request for Comments) dokumentides kirjeldatud standarte.

#### **3.4.1. E-postiserver Postfix, Dovecot ja RoundCube abil**

E-postiinfrastruktuuri peamised komponendid on:

- kasutajaagent (MUA) – kasutaja arvutis töötav tarkvara, mis võimaldab saata ja lugeda kirju;
- edastusagent (MTA) – serverarvutis töötav tarkvara, mis edastab kirju teisele edastusagentile või kohaletoietusagendile;
- kohaletoietusagent (MDA) – serverarvutis töötav tarkvara, mis edastab kirju postkasti;
- IMAP (või POP3) server – serveriarvutis töötav tarkvara, mis üle võrgu annab ligipääsu postkastile.

On olemas lai valik kasutajaagente, mis erinevad kasutajaliidese, funktsioonide, litsentsi ja toetatud operatsioonisüsteemide poolest. Populaarsemad avatud lähtekoodiga kasutajaagendid, millel leidub rohkem konfiguratsioonivõimalusi, on Mozilla Thunderbird (Windows, Mac OSX, Linux), K-9 Mail (Android) ja RoundCube (veebipõhine). RoundCube on asendamatu vahend, kui kasutaja soovib saata või lugeda elektronkirju, kuid puudub võimalus kasutada eelnevalt seadistatud kasutajaagendiga arvutit. RoundCube on mugav seadistada mõnele asutuse alamdomeenile, näiteks mail.d-sys.org. Postfix on populaarne avatud lähtekoodiga ja suurte konfiguratsioonivõimalustega meiliedastusagent, mille arenduse üks peamistest prioriteetidest on turvalisus. Dovecot on avatud lähtekoodiga ja suurte konfiguratsioonivõimalustega POP3 ja IMAP server, mis sisaldab ka kohaletoitetusagenti.

#### 3.4.1.1. Andmebaasi seadistamine

Kuna e-postiserveris on vaja dünaamiliselt kirjeldada kasutajaid ja kirjade edastusreegleid, siis on mugav hoida nimetatud parameetreid relatsioonilises andmebaasis. Autentimist saab korraldada ka teisiti, näiteks kasutades asutuse IT infrastruktuuris olemasolevaid autentimismehhanisme nagu Kerberos või kataloogiteenust LDAP. Relatsioonilise andmebaasi korral on vajalik tabel, millel oleks kaks veergu: kasutajanimi või e-postiaadress ja parool. Parool võib olla nii lahtisel kui ka krüpteeritud kujul. Andmebaasijuhtimissüsteemi (edaspidi ABJS) MySQL korral võib luua eelnevalt kirjeldatud tabelit järgmise käsuga (Näide 18).

```
CREATE TABLE `users` ( `user` varchar(64) NOT NULL, `password` varchar(32) NOT NULL, PRIMARY KEY (`user`) ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Näide 18. Kasutajate tabeli loomine ABJS-s MySQL e-postiserveri jaoks.

Edastusagent Postfix nõuab, et oleksid määratud kõik domeenid, millele saadetud kirjad tuleks edastada kohaletoitetusagendile. Sellist nimekirja on võimalik hoida nii tekstifailis kui ka tabelina andmebaasis. MySQL korral on sobiv järgmise käsuga loodav tabel (Näide 19).

```
CREATE TABLE `domains` ( `domain` varchar(128) NOT NULL, PRIMARY KEY (`domain`) ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Näide 19. Domeenide tabeli loomine ABJS-s MySQL e-postiserveri jaoks.

Viimaseks peab olema ka allikas, mis kirjeldab suunamisi. Suunamine koosneb unikaalsest lähteadressist, millele saadetakse kiri, ja sihtaadressitest, mis määravad kohalikke postkaste või kaudseid e-postiaadresse. Sihtaadressite tõlgendamiseks kasutab Postfix komaga eraldatud väärtustega sõnet, kuid on võimalik kasutada teisi seost 1:n realiseerivaid struktuure. Tabelit,

milles sihtaadressid on eraldatud komaga, on võimalik luua ABJS-s MySQL järgmise käsuga (Näide 20).

```
CREATE TABLE `aliases` ( `src` varchar(64) NOT NULL, `dst` text NOT NULL, PRIMARY KEY (`src`) )  
ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Näide 20. Suunamiste tabeli loomine ABJS-s MySQL e-postiserveri jaoks.

Pärast tabelite loomist on võimalik vajalikud andmed sisestada käsurealt käsitsi, kasutada graafilist andmebaasihaldusvahendit (näiteks phpMyAdmin), importida (teisest andmebaasist või tabelarvutusprogrammist) või lisada haldamiseks vajalikud funktsioonid asutuse infosüsteemi.

### 3.4.1.2. Postfix

Turvalisuse tõstmiseks tuleks lubada kirjade saatmist SMTP läbi turvasoklite kihi (SMTPS, port 465) ja ainult autenditud kasutajatelt. Turvasoklite kihi lubamiseks on vaja tellida või genereerida SSL sertifikaadid, seadistada need failis main.cf ning lubada SMTPS ühendused ja keelata kirjade saatmised ühendusel SMTP konfiguratsioonifailis master.cf.

Kasutajate autentimiseks puudub võimalus pöörduda otse MySQL andmebaasi poole. Probleemi lahendamiseks saab kasutada tarkvarapakette PAM (Pluggable Authentication Modules) ja SASL (Simple Authentication and Security Layer). Mõlemal tarkvarapaketil seadistatakse andmebaasiserveri aadress, kasutajanimi, parool, andmebaasi nimi, tabeli nimi, veergude nimed või SQL päring ning parooli krüpteerimisfunktsioon. Samuti tuleb Postfix konfiguratsioonifailis lubada SASL autentimine (Näide 21).

```
smtpd_sasl_auth_enable = yes  
broken_sasl_auth_clients = yes  
smtpd_sasl_authenticated_header = yes
```

Näide 21. SASL abil autentimise lubamine tarkvara Postfix konfiguratsioonifailis [62].

Selleks, et Postfix suudaks kasutada MySQL andmebaasis olevaid suunamisi ja domeene, peavad olema loodud konfiguratsiooni abifailid, milles on määratud andmebaasiserveri aadress, kasutajanimi, parool, andmebaasi nimi ja SQL päring. Postfix konfiguratsioonis määratakse nimetatud failid järgnevalt (Näide 22).

```
virtual_alias_maps = proxy:mysql:/etc/postfix/sql/aliases.cf, mysql:/etc/postfix/sql/users.cf  
virtual_mailbox_domains = proxy:mysql:/etc/postfix/sql/domains.cf  
virtual_mailbox_maps = proxy:mysql:/etc/postfix/sql/mailboxes.cf
```

Näide 22. MySQL andmebaasi kirjeldavate abifailide määramine tarkvara Postfix konfiguratsioonis [63].

### 3.4.1.3. Dovecot

Sarnaselt Postfix edastusagendiga on IMAP (kui ka POP3) teenustel kasulik lubada turvasoklite kihiga kaitstud ühendused. Erinevalt SMTP teenusega võib lülitada välja turvamata ühendused. SSL ühendused saab lubada ja määrata sertifikaadid konfiguratsioonifailis `dovecot.conf`.

Dovecot võimaldab pärida autentimiseks vajalikke andmeid otse MySQL andmebaasist, mille tõttu ei ole kohustuslik kasutada autentimismehhanisme nagu PAM ja SASL. Andmebaasiga ühendamiseks tuleb määrata konfiguratsiooni abifail, mis sisaldab andmebaasiserveri aadressi, andmebaasi nime, kasutajanime, parooli, SQL päringut ja parooli krüpteerimisfunktsiooni. Kirjeldatud abifail tuleb määrata konfiguratsioonifailis `dovecot.conf` järgnevalt (Näide 23).

```
passdb {  
    driver = sql  
    args = /etc/dovecot/sql.conf  
}
```

Näide 23. Andmebaasi kasutajate tabelit kirjeldava abifaili määramine tarkvara Dovecot konfiguratsioonis [64].

Tarkvaral Dovecot on mitmeid lisavõimalusi nagu meilikvoodid, Sieve (kirjade filtreid kirjeldamise keel) ja ManageSieve (võimaldab kirjeldada filtreid üle võrgu). Samuti on toetatud erinevad postkastivormingud ja on võimalik seadistada kirjade indekseerimist.

### 3.4.1.4. RoundCube

RoundCube on programmeerimiskeeles PHP loodud veebipõhine meili kasutajaagent, mida ettevõtte töötajad saavad kasutada muud tarkvara seadistamata ega paigaldamata üle võrgu. Vaatamata väiksele jalajäljele saab tarkvaras kasutada spellereid, otsida kirju, korraldada identiteete ja malle, vahetada kasutajaliidese keelset ja kasutada ühiseid IMAP kaustu. RoundCube kasutajaliides on tõlgitud kümnetesse keeltesse, sealhulgas ka eesti. [65]

Tarkvara käivitamiseks on vajalik veebiserver (Apache, Lighttpd või Nginx) PHP toega ja andmebaas (MySQL, PostgreSQL, SQLite või MSSQL). Paigaldamiseks tuleb laadida alla tarkvara pakk ja pakkida lahti see veebiserveri poolt töödeldavasse kataloogi. Järgnevalt saab avada veebilehitsejas RoundCube installeerija, mis aitab luua ühenduse andmebaasiga ja seadistada tarkvara.

#### **3.4.1.5. Horisontaalne skaleerimine**

Ei ole välistatud, et üks e-postiserver ei ole võimeline vastu võtma kõiki sissetulevaid kirju ega andma ligipääsu kõikide postkastidele. Probleemi lahendamiseks on võimalik elektronpostiteenuseid laiendada horisontaalselt.

Peamine nõue vertikaalsel skaleerimisel on postkastide paigutus jagatud võrguressursil, mis peab olema kiire. Meiliedastusagentide horisontaalseks skaleerimiseks kasutatakse DNS kirjeid MX, kusjuures on võimalik määrata igale serverile erinev prioriteet. IMAP serverite skaleerimiseks on võimalik seadistada IP aadressite kordamööda vahelduvust DNS tasandil.

#### **3.4.1.6. Andmete varudamine**

Sõltumata sellest, kas on teostatud horisontaalne laienemine või ei ole, võib jagada e-postiteenuste poolt kasutatavad andmed kolmeks: staatilised konfiguratsioonifailid, dünaamilised andmed andmebaasis ja kasutajate postkastid, mis säilitavad kirju failide kujul.

Konfiguratsiooni varundamiseks piisab lihtsast failide kopeerimisest või pakkimisest. Andmebaasi varundamiseks sisaldab ABJS vastavat vahendit, näiteks MySQL tööriista mysqldump, mis koondab andmeid tekstifaili. Postkastide varundamisel võib tekkida suurte andmemahutudega seotud probleeme, kuid tänu failipõhisele struktuurile on võimalik teostada lisanduvat või eristuvat varundamist.

### **3.5. Kodukeskjaam**

Telefoniside ettevõttes sageli tähendab kõnede reeglitepõhist jaotamist, arvestamist ja salvestamist. Kõnedejaotusreeglid võivad olla ajapõhised, mis võimaldavad sõltuvalt ajast jaotada sissetulevaid kõnesi erinevatele töötajate vahel või teavitavad helistajat lahtiolekuaegadest töövälisel ajal. Ei ole haruldane, kui kasutatakse sissetulevate ja väljaminevate kõnede jaoks erinevaid kanaleid, kuna ühtedel pakub operaator paremini meeldejäätavaid telefoninumbreid, teistel on odavam kõneminuti hind. IP telefoniside keskpunktiks on digitaalne kodukeskjaam, mille ülesandeks on lahendada kõiki eelnelt kirjeldatud probleeme. Avatud lähtekoodiga kodukeskjaamadest on populaarsed Asterisk ja FreeSWITCH, mõlemad on konfiguratsioonivõimalusterohked ja sisaldavad palju mooduleid. Järgnevas alampeatükis on toodud näited, kuidas on võimalik Asterisk abil lahendada reeglitepõhist kõnede jaotust, arvestust ja salvestust.



### 3.5.1. Näiteid kodukeskjaama seadistamisest Asterisk abil

Tarkvaras Asterisk defineeritakse kõnede töötlemise reeglid helistamisplaanis, mida sisaldab konfiguratsioonifail `extensions.conf`. Järgnevates lahendustes on kasutatud sissetulevateks kõnedeks laiendit `6000000`.

#### 3.5.1.1. Kelleajast tingitud reeglid

Kuna kõikidel ettevõtetel telefonitoeosakonnad ei tööta 24/7, siis on kasulik töövälisel ajal teavitada klienti lahtiolekuaegadest. Eeldades, et klienditoe telefoninumbrile vastab laiend `6000000`, lahtiolekuaeg on esmaspäevast reedeni kella vahemikus 8:00 kuni 17:00 ja laupäeval 10:00 kuni 15:00 ning teavitus lahtiolekuaegedast on salvestatud faili `/data/sounds/closed`, võib helistamisplaan algada järgnevalt (Näide 24).

```
exten => _6000000,1,GotoIfTime(8:00-17:00,mon-fri,*,*?open)
exten => _6000000,n,GotoIfTime(10:00-15:00,sat,*,*?open)
exten => _6000000,n,Answer
exten => _6000000,n,Wait(1)
exten => _6000000,n,Playback(/data/sounds/closed);
exten => _6000000,n,Hangup
exten => _6000000,n(open),NoOp
```

Näide 24. Ajast tingitud reeglite määramine tarkvara Asterisk helistamisplaani konfiguratsioonis [67].

Toodud näites töövälisel ajal mängitakse helistajale teavitus ning katkestatakse kõne, lahtiolekuaegadel hüpatakse teavituse mängimisest üle ja jätkatakse helistamisplaani täitmist.

#### 3.5.1.2. Helistaja nime otsing

Juhul, kui asutuses on olemas andmebaas või mõni muu allikas, mis sisaldab klientide nimesid ja telefoninumbreid, siis on võimalik otsida helistaja nime telefoninumbri kaudu. Selle lihtsustamiseks tuleb luua käsureaskript, mis võtab esimeseks argumendiks telefoninumbrit ja tagastab kliendi nime. Eeldusel, et kirjeldatud skripti asukoht on `/data/scripts/getname`, lisame helistamisplaani järgmise rea (Näide 25).

```
exten => _6000000,n,Set(CALLERID(name)=$(SHELL(/usr/bin/getname ${CALLERID(num)}))
```

Näide 25. Helistaja nime päring skripti abil tarkvara Asterisk helistamisplaani konfiguratsioonis [67].

Kuna operaator võib edastada telefoninumbrit koos riigikoodiga, siis on kasulik elimineerida riigikood skriptis.

#### 3.5.1.3. Kõnede salvestus

Enne kõnede salvestamist on kasulik määrata salvestuste nimede vorming, mis võib sisaldada kuupäeva, kellaega, suunda ja telefoninumbrit. Salvestuste andmemahu vähendamiseks saab

kasutada mõnda tihendamise algoritmi nagu MP3. Kuna paljudel Asterisk versioonidel on olemas MP3 enkooder, kuid võib puududa dekodeer, on võimalik kasutada välist dekodeerit. Näites 26 defineeritakse failinimede vorming, salvestatakse kõne, tihendatakse see vormingusse MP3 ja kustutakse tihendamata salvestus.

```
exten => _6000000,n,Set(RECFN=/data/rec/${STRFTIME(${EPOCH},,%Y%m%d-%H%M%S)}_in_${CALLERID(num)})
exten => _6000000,n,MixMonitor(${RECFN}.wav,b,lame --quiet --resample 16 -b 16 ${RECFN}.wav ${RECFN}.mp3 && rm ${RECFN}.wav)
```

Näide 26. Kõnede salvestamine ja tihendamine MP3 vormingusse tarkvara Asterisk helistamisplaani konfiguratsioonis [67].

Eelnevas näites on kasutatud avatud lähtekoodiga MP3 enkooderit Lame. Tihendatud helifaili diskreetimissageduseks on 16 KHz ja bitikiiruseks 16 Kbit/s. Kodeerimisprotsessi järel kustutatakse kodeerimata helifail.

#### 3.5.1.4. Kõnejärjekorrad

Kaks peamist kõnejärjekorra funktsiooni on jaotada sissetulevaid kõnesi klienditeenindajate vahel ja leevendada helistaja ooteaega muusika abil. Kõnejärjekorras määratakse selle liikmed kasutajanimedena ja üks järgmistest kõnede jaotusstrateegiatest: kõik liikmed kuni keegi vastab, kõige rohkem aega tagasi vastanud liige, kõige vähema vastatud kõnede arvuga liige, juhuslik valik ja teised. Järgnev näide, mida defineeritakse konfiguratsioonifailis queues.conf, sisaldab järjekorda nimega support, mille liikmeteks on 5 klienditeenindajat ja jaotusstrateegiaks on kõige vähema vastatud kõnede arvuga liige (Näide 27).

```
[support]
strategy=fewestcalls
member => SIP/1011
member => SIP/1012
member => SIP/1013
member => SIP/1014
member => SIP/1015
```

Näide 27. Kõnejärjekorra seadistamine ja selle liikmete määramine tarkvaras Asterisk [67].

Lisaks järjekorra defineerimisele tuleb järgneva käsuga suunata kõned kõnejärjekorda helistamisplaanis (Näide 28).

```
exten => _6000000,n,Queue(support)
exten => _6000000,n,Hangup
```

Näide 28. Kõnede suunamine kõnejärjekorda tarkvara Asterisk helistamisplaani konfiguratsioonis [67].

Kui puudub vajadus käskude töötlemiseks pärast kõne lõpetamist, siis on kasulik väljuda helistamisplaanist käsuga Hangup.

Kõnejärjekordasid on võimalik luua käsitsi erinevate tingimuste alusel. Näiteks kui asutuse infosüsteemis on klient seotud kindla klienditeenindajaga, siis on võimalik kliendi kõned suunata otse klienditeenindajale.

#### 3.5.1.5. Kõnede arvestus

Kõnede arvestuse pidamiseks on mitmeid põhjuseid, näiteks sissetulevate kõnede kuupäevaline või ajaline statistika ja klienditeenindajate aktiivsuse statistika. Vaikimisi salvestab Asterisk kõnede üksikasju CSV faili, mille vormingut saab ka muuta, kuid CSV failist statistika arvutamine nõuab lisavahendeid.

Kogutud andmete lihtsamaks töötlemiseks on mõistlik kasutada relatsioonilist andmebaasi. Kõnede üksikasju on võimalik salvestada MySQL andmebaasi Asterisk mooduli cdr\_mysql abil. Paraku moodul cdr\_mysql ei võimalda käsitsi defineerida andmebaasitabeli veerge, et salvestada lisaks näiteks kõnes osalejate IP aadresse, kõnele vastaja kasutajanime ja teisi parameetreid. Laiemate võimalustega alternatiiviks, millel puuduvad eelnevalt kirjeldatud piirangud, on moodul cdr\_odbc, mis kasutab ODBC vahevara andmebaasiga suhtlemiseks. Eeldades, et ODBC ja moodul cdr\_odbc on seadistatud ning muutuja RECFN sisaldab salvestatud kõne failinime, saab selle edastada andmebaasitabeli veergu recfn järgneva helistamisplaani reeglina (Näide 29).

```
exten => _6000000,n,Set(CDR(recfn)=${RECFN})
```

Näide 29. Helistamisplaani muutuja sisu edastamine andmebaasi kõnederegistri tabelisse tarkvara Asterisk helistamisplaani konfiguratsioonis [67].

Eelnevas näites toodud kujul on võimalik salvestada suvalise helistamisplaani muutuja sisu suvalise nimega andmebaasitabeli veergu. Eeldusel, et kõnede arvestus salvestatakse MySQL andmebaasi tabelisse cdr, milles veerg start sisaldab kõnede algusaegu ja veeru dcontext parameeter elion vastab sissetulevate kõnede kontekstile, saab pärida sissetulevate kõnede ajalist jaotust vahemikus 01.01.2014 kuni 01.02.2014 järgnevalt (Näide 30).

```
mysql> SELECT RIGHT(LEFT(start, 13), 2) as Time, COUNT(*) as Count FROM cdr
WHERE dcontext='elion' AND start>'2014-01-01' AND start<'2014-02-01' GROUP BY Time;
+-----+-----+
| Time | Count |
+-----+-----+
| 07 | 8 |
| 08 | 67 |
| 09 | 94 |
| 10 | 117 |
| 11 | 142 |
| 12 | 135 |
| 13 | 149 |
| 14 | 142 |
| 15 | 128 |
| 16 | 81 |
| 17 | 32 |
| 18 | 5 |
| 19 | 4 |
| 20 | 2 |
+-----+-----+
```

Näide 30. SQL päring ja selle vastus, mis näitab sissetulevate kõnede ajalist jaotust.

Sarnaselt on võimalik pärida klienditeenindaja kasutajanimede poolt vastatud kõnede arvu, aega minutites ja keskmist kõne kestvust sekundites. Eeldusel, et kõnede arvestust salvestatakse MySQL andmebaasi tabelisse cdr, milles veerg ans sisaldab kõnele vastanud klienditeenindaja kasutajanime, veerg billsec kõne kestvusi sekundites, veerg start kõnede algusaegu, veeru dcontext parameeter elion vastab sissetulevate kõnede kontekstile ja veeru disposition parameeter ANSWERED viitab vastatud kõnele, saab pärida vastatud kõnede arvu, kestvust minutites ja ühe kõne keskmist kestvust sekundites klienditeenindajate kasutajanimede kaupa vahemikus 01.01.2014 kuni 01.02.2014 järgnevalt (Näide 31).

```
mysql> SELECT ans AS Username, COUNT(*) AS Calls, ROUND(SUM(billsec)/60) AS Minutes, ROUND(SUM(billsec)/COUNT(*)) AS SPC
FROM cdr WHERE dcontext='elion' AND disposition='ANSWERED' AND start>'2014-01-01' AND start<'2014-02-01' GROUP BY Username;
+-----+-----+-----+-----+
| Username | Calls | Minutes | SPC |
+-----+-----+-----+-----+
| 1011 | 192 | 787 | 246 |
| 1012 | 261 | 692 | 159 |
| 1013 | 209 | 724 | 208 |
| 1014 | 183 | 680 | 223 |
| 1015 | 218 | 731 | 201 |
+-----+-----+-----+-----+
```

Näide 31. SQL päring ja selle vastus, mis näitab vastuvõetud kõnede arvu, kestvust minutites ja ühe kõne keskmist kestvust sekundites klienditeenindajate kasutajanimede kaupa.

Eelnevas näidetes toodud statistika päringuid saab automatiseerida või integreerida asutuse infosüsteemi.

### 3.5.1.6. Teavitus vastamata kõnedest

Kuigi vastamata kõnesid on võimalik jälgida CSV failist või andmebaasitabelist, on olukordi, kui nendele tasub pöörata erilist tähelepanu. Selleks on kasulik kirjutada käsuraaskript, mis võtab argumendiks sissetuleva kõne telefoninumbri (vajadusel ka teised parameetrid) ja saadab teavituse e-postiga või impordib sündmuse asutuse infosüsteemi. Eeldades, et

eelnevalt kirjeldatud skripti asukoht on /data/scripts/notifymissed, lisatakse helistamisplaani lõppu järgmised käsud (Näide 32).

```
exten => h,1,GotoIf(["${DIALSTATUS}" = "ANSWER"]?end)
exten => h,n,System(/data/scripts/notifymissed ${CALLERID(num)})
exten => h,n(end),Hangup
```

Näide 32. Skripti käivitamine kõnele vastamata jätmisel tarkvara Asterisk helistamisplaani konfiguratsioonis.

Juhul, kui asutuse infosüsteemis on kliendile määratud kindel klienditeenindaja, siis võib skript vastamata kõnest teavitada otse klienditeenindajat.

### 3.5.1.7. Väljuvate kõnede reeglid ning jaotus

Sarnaselt sissetulevate kõnedega on võimalik väljuvatel kõnedel kasutada ajalisi tingimusi näiteks kõnede keelamiseks, teostada kõnede salvestust, arvestust ja teisi operatsioone. Väljuvate kõnede jaotus on kasulik olukordades, näiteks kui ettevõtte kasutab piiratud kõneajaga teenuspakette, millel on vaja tasakaalustada väljuvaid kõnesid. Kõnede tasakaalustust on võimalik korralda nii kõnede arvu, kõneaja kui ka teiste tingimuste järgi. Järnevas näites tasakaalustatakse väljuvaid kõnesid kanalitel SIP/out.1 ja SIP/out.2 vaheldumisi (Näide 33).

```
exten => _X.,1,GotoIf(["${DB_EXISTS(support-out/next)}" = "0"]?er1)
exten => _X.,n,Goto(er${DB(support-out/next)})
exten => _X.,n(out1),Set(DB(support-out/next)=2)
exten => _X.,n,Dial(SIP/out.1/${EXTEN})
exten => _X.,n,ExecIf(["${DIALSTATUS}" = "BUSY"]?Hangup)
exten => _X.,n,GotoIf(["${DIALSTATUS}" = "CHANUNAVAIL"]?er${DB(support-out/next)})
exten => _X.,n(out2),Set(DB(support-out/next)=1)
exten => _X.,n,Dial(SIP/out.2/${EXTEN})
exten => _X.,n,ExecIf(["${DIALSTATUS}" = "BUSY"]?Hangup)
exten => _X.,n,GotoIf(["${DIALSTATUS}" = "CHANUNAVAIL"]?er${DB(support-out/next)})
```

Näide 33. Väljuvate kõnede tasakaalustamine kahe kanali vahel tarkvara Asterisk helistamisplaani konfiguratsioonis.

Eelnevalt toodud näites kasutatakse Asterisk moodulit DB, mis kujutab endast lihtsustatud andmebaasi. Võrreldes helistamisplaani muutujaga säilitab DB väärtused ka helistamisplaanist väljumisel ehk kõne lõppemisel. Selliselt salvestatakse kanali number, mida tuleb kasutada järgmisena. Samuti on helistamisplaani lisatud tingimused, mis vahetavad kanalit, kui helistamine nurjub tehnilistel põhjustel.

## **Lahendused asutustele avatud lähtekoodiga tarkvara näitel.**

Dmitri Pantšenko

### **4. KOKKUVÕTE**

Antud diplomitöö eesmärgiks on näidata avatud lähtekoodiga serveripoolse tarkvara alahinnatud potentsiaali ning selle ilmseid eeliseid kommertstarkvara ja -toodete ees. Analüüs on tehtud tüüpilise asutuse IT infrastruktuuri näitel.

Diplomitöö raames viidi läbi uurimus eeltoodud väite tõestamiseks. Esmase ülesandena koguti statistikat IT infrastruktuuridest. Andmed pärinevad Eesti Haridussüsteemist ja mitmetest Tallinna gümnaasiumitest. Tulemused on koondatud esimesse peatükki ning näitavad, et avatud lähtekoodiga tarkvara kasutatakse vähesel määral.

Edaspidi uuritakse põhjalikumalt asutuste IT infrastruktuurides enimkasutatud kommertstarkvara ja -tooteid. Teine peatükk on rühmitatud vastavalt probleemile: marsruutimine, virtualiseerimine, veebiserver, e-postiserver ja IP-telefoniside. Igas rühmas on esitatud toodete näited ja toodetel on analüüsitud funktsionaalsust, jõudlust, halduse keerukust ning toote hinda. Peatükis nimetatud toodete põhilisteks puudujäägiks on piiratud funktsionaalsus ja kõrge litsentsihind.

Kolmandas peatükis tuuakse välja teises peatükis nimetatud toodete avatud lähtekoodiga alternatiivid. Iga tootele on lisatud peale kirjelduse ka konfiguratsiooni näited ja juhendid. Nimetatud toodete põhilisteks puudujääkideks on puudulik dokumentatsioon ja konfiguratsiooni liiasus. Eelisteks on aga usaldusväärsem lähtekood, paindlik konfiguratsioon ja tasuta litsentsid.

Kuigi avatud lähtekoodiga tarkvaral võib esineda puudusi, nagu täiusliku dokumentatsiooni puudumine, selle kasutamine vähendab mitmeid riske. Näiteks kommertstarkvara kasutamisel võib selle arendaja lõpetada toote arenduse ja toe, mis võib nõuda nii tarkvara enda kui ka sellest sõltuvate lahenduste väljavahetust. Avatud lähtekoodiga tarkvara on õigus kõigil muuta ja täiendada. Selliselt võib iga kasutaja kohandada olemasolevaid lahendusi vastavalt oma vajadustele. Kommertstarkvaral põhinevate lahenduste kvaliteetne haldamine võib nõuda arendaja poolt väljaõpetatud personali, kelle otsingud võivad tekitada raskusi. Avatud lähtekood lubab kiiresti tuvastada ja parandada vigu ja turvaauke tänu lähtekoodile ligipääsu

omavate inimeste arvule. Avatud lähtekood garanteerib varjatud kahjulike mehhanismide puudumist. Ühte ja sama lähtekoodi on võimalik kompileerida erinevatele arhitektuuridele. See tähendab, et vajadusel saab kolida ühelt arhitektuurilt teisele (näiteks armhf-lt x86\_64-le) tarkvara vahetamata, kusjuures konfiguratsioon on üldjuhul ühilduv. Nii kommertstarkvaral kui ka avatud lähtekoodiga põhinevad lahendused nõuavad kulutusi nende paigaldamiseks ja ülalpidamiseks. Kommertstarkvara arendajad hoolitsevad selle eest, et nende toode oleks dokumenteeritud ja mugavam, mille tõttu paigaldus- ja ülalpidamiskulud võivad olla väiksemad. Avatud lähtekoodiga tarkvara paigaldus- ja ülalpidamiskulude vahe kommertstarkvara omadega on üldjuhul väiksem kui kommertstarkvara litsentsi hind. Selliselt saab avatud lähtekoodiga tarkvara abil kokku hoida IT kuludel ja samaaegselt stimuleerida kohaliku IT sektorit.

Marsruutimisega seotud probleemide lahendamisel selgub, et avatud lähtekoodiga tarkvara funktsionaalsus, sealhulgas ka marsruutimisprotokollide toe olemasolu, on vähemalt sama hea kui kommertstoodetel. Krüpteerimisvõimaluste poolest on üldotstarbelised arvutid isegi nõrkade ja odavate protsessoritega nagu Intel Atom D2500 on konkurentsivõimelised paljude kommertstoodetega. Suuremate kiiruse saavutamiseks alati võimalik kolida võimsamale riistvarale konfiguratsiooni muutmata või lisada krüpteerimisseadmeid.

Riistvara virtualiseerimiseks on lai valik avatud lähtekoodiga vahendeid, mille hulka lisandus XenServer. Avatud lähtekoodiga virtuaalmasinahaldurite (nagu KVM) funktsionaalsus võib olla halvem kui tasulistel konkurenttoodetel: näiteks puudub võimalus tõsta töötaval virtuaalmasinal kettakujutisi füüsiliste salvestusseadmete vahel. Tugevateks külgedeks on lai valik virtualiseeritavat riistvara ja kettakujutistevorminguid, kusjuures viimaste seas on omandiõigusega standarditega, mille sisule on olemas ligipääs. Kuigi avatud lähtekoodiga haldustarkvaral libvirt puuduvad koormuse jaotust automatiseerivad vahendid, see toetab märkimisväärset arvu virtuaalmasinahaldureid ja kasutab seadete salvestamiseks avatud standardeid (libvirt korral XML), mida on võimalik töödelda mitmete vahenditega või integreerida selle töötlemist asutuse infosüsteemi. Lisaks eksisteerivad graafilise kasutajaliidesega abivahendid, mis muudavad virtuaalmasinate haldamise mugavamaks.

Lihtsa veebiserveri loomine Apache abil eristub tasulistest konkureerivatest toodetest seadistamise lihtsuse ja kiiruse poolest, märkimisväärne arv lisamoduleid annab tarkvarale suurt hulka funktsionaalseid võimalusi, mille tulemusena omab tarkvara turuliidri positsiooni. Veebiklastri loomine Nginx pöördproksi ja Apache sõlmede abil tähendab piiramatut laienemisvõimalusega lahendust, millel puuduvad kulud tarkvara litsentsidele.

Kuigi e-postiserveritarkvara turul leiduvad lahendused nagu Microsoft Exchange, millel on lihtsustatud autentimismehhanismide seadistamine ja horisontaalne skaleerimine, domineerivad lahendused, mis koosnevad avatud lähtekoodiga komponentidest. Viimaste paigaldus ja haldamine nõuab manipuleerimist kõikide komponentidega, kuid samaaegselt lubab luua lahendusi avatud standarditel, mida saab horisontaalselt skaleerida ja lihtsalt varundada. Lisaks eksisteerivad mugavad veebipõhised kasutajaagendid nagu RoundCube, mis sarnanevad tuntud avalikele teenustele nagu Gmail ja Outlook.com.

Kodukeskjaamade turul leidub nii tarkvaralisi kui ka riistvaralisi kommertslahendusi, mille odavate litsentsidega võivad kaasneda suuremad piirangud. Avatud lähtekoodiga tarkvara kasutamisel võivad piirangud olla tingitud ainult riistvaralisest võimekusest. Samuti on näidatud, et avatud lähtekoodiga kodukeskjaam nagu Asterisk on konfiguratsiooni- ja integratsioonivõimalusterohke – kõik helistamisplaani etapid on seadistatavad ja mitmeid võimalusi ühendamiseks andmebaaside ja käsureaga.

Ülaltoodud fakte silmas pidades on antud töö loogiline kokkuvõtte järgmine: vabavara tarkvara võimsust alahinnatakse. Osapooled on huvitatud kasutusvalmis toodetes, mis võivad olla hoolduse seisukohalt odavamad. Vabavara vähene kasutus on samuti motiveeritud väärusel veendumusel, et vabavara seadistamise ja hoolduse keerukus nõuavad kvalifitseeritud eksperti. Kolanda peatüki näited tõestavad vastupidist. Selline põhjendamatu vabavaratoodetest loobumine on tänapäeval tüüpiline ja seda tuleks vähendada. Antud tööd saab kasutada vabavara kasutuselevõtmist argumenteerimiseks.



## 5. KASUTATUD KIRJANDUSE LOETELU

1. K. Thomas, *Beginning Ubuntu Linux: From Novice to Professional* (Berkeley, CA, 2006)
2. M. Teder, Jaak Aaviksoo: Ministeerium võitles haridusasutustele välja Windowsi soodushinnad, 19.12.2013
3. I. Horm, Vaba tarkvara kasutamine võrguteenuste osutamisel Tallinna munitsipaalkoolides, bakalaureusetöö (Tallinna Tehnikaülikool, 2012)
4. J. Lerner, J. Tirole, Some simple economics on open source, *Journal of Industrial Economics* (2002)
5. Intel-i kodulehekül, Open Source at Intel: Building with the Ecosystem, <http://software.intel.com/sites/oss/linuxkernel.php> [3.12.2012]
6. J. W. Eckert, M. J. Schitka, *Linux+ Guide to Linux Certification, Second Edition* (Canada, 2006)
7. R. Jaques, "Open source code quality is as good as proprietary software," (2012), <http://www.theinquirer.net/inquirer/news/2154870/source-code-quality-proprietary-software>
8. Microsoft kodulehekül, Windows Internet Explorer 8 Privacy Statement, <http://windows.microsoft.com/en-CA/internet-explorer/products/ie-8/privacy-statement> []
9. D. Riehle, "The Economic Motivation of Open Source Software: Stakeholder Perspectives," (2007), <http://www.riehle.org/computer-science/research/2007/computer-2007-article.html> [11.12.2012]
10. Cisco Systems kodulehekül, Cisco Systems Corporate Timeline, [http://newsroom.cisco.com/dlls/corporate\\_timeline.pdf](http://newsroom.cisco.com/dlls/corporate_timeline.pdf) [18.11.2013]
11. Cisco Systems kodulehekül, Cisco Product Quick Reference Guide 2011, <http://www.cisco.com/en/US/prod/collateral/netmgtsw/cpqrg.pdf> [19.11.2013]
12. Cisco Systems kodulehekül, Enterprise routing solutions for a borderless network, [http://www.cisco.com/en/US/prod/collateral/routers/ps10536/Routing\\_Poster.pdf](http://www.cisco.com/en/US/prod/collateral/routers/ps10536/Routing_Poster.pdf) [19.11.2013]

13. Cisco Systems kodulehekülg, Cisco Integrated Services Routers - Performance Overview, [https://supportforums.cisco.com/servlet/JiveServlet/download/3667060-130325/white\\_paper\\_c11\\_595485.pdf](https://supportforums.cisco.com/servlet/JiveServlet/download/3667060-130325/white_paper_c11_595485.pdf) [19.11.2013]
14. Cisco Systems kodulehekülg, Cisco Site-to-Site VPN Technologies Comparison, [http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6586/ps6635/ps7180/prod\\_brochure0900aecd80582078.pdf](http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6586/ps6635/ps7180/prod_brochure0900aecd80582078.pdf) [20.11.2013]
15. Hewlett-Packard kodulehekülg, HP Completes Acquisition of 3Com Corporation, <http://www8.hp.com/us/en/hp-news/press-release.html?id=342187#.Uofdf0MS3v4> [25.11.2013]
16. Hewlett-Packard kodulehekülg, A brief technical overview of HP MSR Series of routers, <http://h20195.www2.hp.com/v2/GetPDF.aspx%2F4AA4-3841ENW.pdf> [26.11.2013]
17. Hewlett-Packard kodulehekülg, HP MSR Series Router Performance Evaluation and Feature Validation, <http://www8.hp.com/h20195/v2/GetPDF.aspx%2F4AA4-9069ENW.pdf> [26.11.2013]
18. Huawei kodulehekülg, Routers – Huawei Products, <http://enterprise.huawei.com/en/products/network/router/index.htm> [2.12.2013]
19. Huawei kodulehekülg, Huawei Router Performance and Reliability, [http://huawei.com/ilink/cnenterprise/download/HW\\_127517](http://huawei.com/ilink/cnenterprise/download/HW_127517) [2.12.2013]
20. MikroTik kodulehekülg, RouterBOARD Product Catalog 4Q 2012, <http://download2.mikrotik.com/Q4.pdf> [12.12.2013]
21. MikroTik kodulehekülg, RouterOS features, [http://wiki.mikrotik.com/wiki/Manual:RouterOS\\_features](http://wiki.mikrotik.com/wiki/Manual:RouterOS_features) [13.12.2013]
22. F. L. Camargos, G. Girard, B. des Ligneris, Virtualization of Linux servers (Ottawa, Canada 2008)
23. V. Josyula , M. Orr , G. Page, Cloud Computing: Automating the Virtualized Data Center (Indianapolis, IN, 2012)
24. R. Troy, M. Helmke, VMware Cookbook Second Edition, (Sebastopol, CA, 2012)

25. VMware kodulehekülg, VMware vSphere with Operations Management and VMware vSphere Licensing, Pricing and Packaging,  
<http://www.vmware.com/files/pdf/products/vsphere/VMware-vSphere-Pricing-Whitepaper.pdf> [18.01.2014]
26. L. Carvalho, Windows Server 2012 Hyper-V Cookbook (Birmingham, UK, 2012)
27. A. Finn, P. Lownds, Michel Luescher, Damian Flynn, Windows Server 2012 Hyper-V Installation and Configuration Guide (Indianapolis, IN, 2013)
28. Microsoft, Volume Licensing reference guide for Windows Server 2012 R2 (Redmond, WA, 2013)
29. D. Tosatto, Citrix XenServer 6.0 Administration Essential Guide (Birmingham, UK, 2012)
30. Citrix kodulehekülg, Citrix Extends Open Source Strategy with XenServer,  
<http://www.citrix.com/news/announcements/jun-2013/citrix-launches-open-source-xenserver.html> [26.01.2014]
31. Netcraft kodulehekülg, January 2014 Web Server Survey,  
<http://news.netcraft.com/archives/2014/01/03/january-2014-web-server-survey.html>  
[24.02.2014]
32. C. Metz, Google mystery server rooted in Apache,  
[http://www.theregister.co.uk/2010/02/03/google\\_web\\_server\\_based\\_on\\_apache/](http://www.theregister.co.uk/2010/02/03/google_web_server_based_on_apache/) [24.02.2014]
33. K. Schaefer, J. Cochran, S. Forsyth, D. Glendenning, B. Perkins, Professional Microsoft IIS 8 (Indianapolis, IN, 2012)
34. J. P. Mueller, Mastering IIS 7 Implementation and Administration (Indianapolis, IN, 2007)
35. Cybercom Group AB, Open email survey (2013), <http://www.openemailsurvey.org/>  
[28.02.2014]
36. T. Redmond, Microsoft Exchange Server 2010 Inside Out (Redmond, WA, 2010)
37. T. Redmond, Microsoft Exchange Server 2013 Inside Out: Mailbox and High Availability (Redmond, WA, 2013)
38. M. M. Landis, R. A. Lloyd, The 3CX IP PBX Tutorial (Birmingham, UK, 2010)

39. 3CX kodulehekülg, 3CX Phone System Pricing,  
<http://www.3cx.com/ordering/pricing/> [5.03.2014]
40. 3CX kodulehekülg, Features Comparison,  
<http://www.3cx.com/phone-system/edition-comparison/> [5.03.2014]
41. BIRD kodulehekülg, BIRD User's Guide: Protocols,  
[http://bird.network.cz/?get\\_doc&f=bird-6.html](http://bird.network.cz/?get_doc&f=bird-6.html) [15.03.2014]
42. Quagga kodulehekülg, Quagga Documentation,  
<http://www.nongnu.org/quagga/docs/docs-info.html> [15.03.2014]
43. XORP kodulehekülg, XORP User Manual,  
[http://www.xorp.org/releases/1.8-CT/docs/user\\_manual/user\\_manual.pdf](http://www.xorp.org/releases/1.8-CT/docs/user_manual/user_manual.pdf) [15.03.2014]
44. Debian kodulehekülg, Network Configuration,  
<https://wiki.debian.org/NetworkConfiguration> [17.03.2014]
45. R. P. Pollei, Debian 7: System Administration Best Practices (Birmingham, UK, 2013)
46. Debian kodulehekülg, Bridging Network Connections,  
<https://wiki.debian.org/BridgeNetworkConnections> [17.03.2014]
47. RedHat kodulehekülg, FORWARD and NAT Rules,  
[https://access.redhat.com/site/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/html/Security\\_Guide/sect-Security\\_Guide-Firewalls-FORWARD\\_and\\_NAT\\_Rules.html](https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Security_Guide/sect-Security_Guide-Firewalls-FORWARD_and_NAT_Rules.html) [18.03.2014]
48. Debian kodulehekülg, Debian Firewall, <https://wiki.debian.org/DebianFirewall>  
[18.03.2014]
49. Debian kodulehekülg, iptables, <https://wiki.debian.org/iptables> [18.03.2014]
50. C. Liu, P. Albitz, DNS and BIND, 5th Edition (Sebastopol, CA, 2006)
51. Internet Systems Consortium, ISC DHCP Server 4.3.0 Documentation,  
<http://www.isc.org/wp-content/uploads/2013/05/DHCP-Distribution-Documentation-20-4-7.pdf> [19.03.2014]

52. RedHat kodulehekül, The Evolution of Virtualization: Qumranet joins Red Hat, <http://www.redhat.com/promo/qumranet/> [24.03.2014]
53. QEMU kodulehekül, QEMU Emulator User Documentation, <http://wiki.qemu.org/download/qemu-doc.html> [24.03.2014]
54. libvirt kodulehekül, libvirt hypervisor drivers, <http://libvirt.org/drivers.html> [25.03.2014]
55. libvirt kodulehekül, libvirt authentication, <http://libvirt.org/auth.html> [26.03.2014]
56. libvirt kodulehekül, libvirt domain XML format, <http://libvirt.org/formatdomain.html> [27.03.2014]
57. libvirt kodulehekül, Applications using libvirt, <http://libvirt.org/apps.html> [28.03.2014]
58. Nginx kodulehekül, Module ngx\_http\_upstream\_module, [http://nginx.org/en/docs/http/ngx\\_http\\_upstream\\_module.html](http://nginx.org/en/docs/http/ngx_http_upstream_module.html) [3.04.2014]
59. Nginx kodulehekül, Module ngx\_http\_rewrite\_module, [http://nginx.org/en/docs/http/ngx\\_http\\_rewrite\\_module.html](http://nginx.org/en/docs/http/ngx_http_rewrite_module.html) [5.04.2014]
60. Nginx kodulehekül, Nginx Content Caching, <http://nginx.com/resources/admin-guide/caching/> [7.04.2014]
61. Nginx kodulehekül, Setting Up a Virtual Server, <http://nginx.com/resources/admin-guide/web-server/> [7.04.2014]
62. Postfix kodulehekül, Postfix SASL Howto, [http://www.postfix.org/SASL\\_README.html](http://www.postfix.org/SASL_README.html) [10.04.2014]
63. Postfix kodulehekül, Postfix Virtual Domain Hosting Howto, [http://www.postfix.org/VIRTUAL\\_README.html](http://www.postfix.org/VIRTUAL_README.html) [11.04.2014]
64. Dovecot kodulehekül, Authentication: master users/passwords, <http://wiki2.dovecot.org/Authentication/MasterUsers> [16.04.2014]
65. Roundcube kodulehekül, Roundcube released features, <http://roundcube.net/about/#features> [21.04.2014]
66. Roundcube kodulehekül, Roundcube Wiki, <http://trac.roundcube.net/wiki> [21.04.2014]

67. R. Bryant, L. Madsen, J. Van Meggelen, *Asterisk: The Definitive Guide*, 4th Edition  
(Sebastopol, CA, 2013)

## **Open-source software-based solutions for institutions.**

Dmitri Pantšenko

### **6. SUMMARY**

The goal of this thesis is to show the underestimated potential of open-source servers-side software and its obvious advantages over commercial on example of IT-environment of typical organization.

To prove the declared statement there was a research conducted in the scope of this thesis. As a primal subject of the case study the IT-infrastructure statistics surveys were selected. The corresponding reports are covering Estonian Educational System and a number of high-schools in Tallinn. The results were summarized in the first chapter of the thesis and show clear lack of open-source software (OSS) usage in those organizations.

Further in thesis there is a detailed study of commercial software and hardware that is typically present in the IT-infrastructures of the organizations. The second chapter is subdivided into sections by purpose of the typical IT-infrastructure components: routing, virtualization, webserver, mailserver and VoIP. In every group there are a number of examples given with detailed analysis including functionality, performance, maintenance complexity and product prices. The general disadvantages of the listed commercial products are limited functionality and high license price.

In order to convince the audience in the advantages of the OSS, the third chapter of the thesis illustrates the corresponding open-source solutions (to the ones described in the second chapter). In addition to the evaluation of each product the example configurations and manuals are given. The general disadvantage of listed products are lack of documentation and configuration overhead. The advantages are however much more reliable code (due to its openness), highly customizable configuration and free licenses.

Although OSS might have drawbacks like lack of documentation, using OSS may reduce many risks. For example, at some point commercial software vendors may stop the product development and support, this situation requires to look for a new solution and software replacement. In case of OSS, everyone is allowed to modify and update the software. This also means that everyone is allowed to adapt the existing OSS solutions to their needs.

Administration of commercial software based solutions may require vendor trained workforce whose searches can be hard. The principles of OSS speed up identification and correction of errors and vulnerabilities in the source code due the number of people who have access and guarantees absence of harmful mechanisms. The same source code can compiled for different architectures. This means that there is always an opportunity to migrate from one architecture to another (for example from armhf to x86\_64) without replacing software itself while the configuration is generally compatible. Both commercial and open-source software-based solutions require spendings on their installation and maintenance. Commercial software vendors assure that their product would be documented and user-friendly, what should decrease the installation and maintenance costs. However, the difference between the open-source and the commercial software installation and maintenance price is usually not as big as the commercial software licence price. As a result using OSS can save IT costs and at the same time stimulate local IT sector.

Considering problems related to the routing, it appears that the OSS functionality, including the support of routing protocols, is at least as good the as commercial products'. In terms of encrypted the traffic throughput the general-purpose computers even with weak and cheap CPUs like Intel Atom D2500 are competitive with many commercial products. In order to achieve better performance, there are always possibilities to migrate to more powerful hardware without changing software configuration, or add encryption hardware.

Hardware virtualization can be done with a wide range of open-source tools, which now include XenServer. The functionality of open source hypervisors like KVM may be worse than the possibilities of the commercial competitor products. For example, OSS hypervisors do not have storage live migration. The advantages of OSS hypervisors are a good choice of emulated hardware and supported disk image formats, whereby the last ones include open standarts which provide an access to content. Open-source management tools like Libvirt do not have automatic load balancing functions, but do support notable number of hypervisors and save their configuration in open standarts (Libvirt in XML) which can be processed by many tools or the processing may be integrated in instituion's information system. In addition, there are several tools available with GUI that make management of virtual machines easier.

Building simple web server using Apache distinguishes from usage of commercial competitor products with simplicity and quickness, notable number of addons gives Apache almost infinite possibilities, what make Apache market leader. Building a web cluster with Nginx as reverse proxy and Apache as nodes provides unlimited scaling capabilities.



Although on the e-mail server software market there are available solutions like Microsoft Exchange, which have simplified setup of authentication mechanisms and horizontal scaling, dominate solutions built from open-source components. Installation and maintenance of the last ones require manipulation of all dependant components, but they allow to build solutions based on open standarts, whjch can be scaled horisontaly and easily backed-up. In addition, there are convenient web-based mail user agents like RoundCube witch look similar to public services like Gmail and Outlook.com.

Finally, there are software and hardware commercial products on the PBX market, however their cheaper licences often stands forhigher restrictions. Using OSS, the limitations are mostly applied by the hardware capabilities. In the thesis it was shown that open-source PBXs like Asterisk are highly configurable and integratable – all dialplan steps are adjustable and there are many ways for using database and shell.

Considering the above mentioned facts there is a logical conclusion of the thesis: the power of open-source software is underestimated and it's limited use mostly motivate by short term expenses-savings. The decision makers are interested in quick and ready-to-use products which might be cheaper from the maintenance point of view. Limited usage of open-source software is also motivate by faulty belief of it's complex configuration and maintenance demanding qualified expert. The practical examples in 3rd chapter show the opposite. Such an unreasonable refusal of open-source products is somewhat typical nowadays and needs to be reconsidered. This thesis can be used as valid argument supporting decisions towards the use of open-source software.

## **7. LIHTLITSENTS LÕPUTÖÖ REPRODUTSEERIMISEKS JA LÕPUTÖÖ ÜLDSUSELE KÄTTESAADAVAKS TEGEMISEKS**

Mina, Dmitri Pantšenko,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose

Lahendused asutustele avatud lähtekoodiga tarkvara näitel

mille juhendaja on Artjom Lind

1.1. reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;

1.2. üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace'i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.

3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartus, 27.05.2014