TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Martin Mihkelsaar

IPv6 juurutamine Astro Baltics näitel

Bakalaureusetöö (9 EAP)

Juhendaja: Toomas Lepik
Kaasjuhendaja: Artjom Lind

Tartu 2021

IPv6 juurutamine Astro Baltics näitel

Lühikokkuvõte:

Käesoleva bakalaureusetöö eesmärk on ettevõttes Astro Baltics võrgu infrastruktuur üle viia IPv4 protokolli pealt täielikult IPv6 protokolli peale lähtudes ettevõtte eripäradest. IPv6 on järglane IPv4 protokollile, mis pakub paremat turvalisust, kiirust ja on palju tulevikukindlam kuna pakub ka rohkem aadresse. Teema on aktuaalne, sest järjest rohkem on maailmas hakatud üle minema IPv4 pealt IPv6 tehnoloogiale ja seda peetakse tänapäeva protokolliks. Üleminek peab olema ette valmistatud ja eelnevalt läbi testitud, et teenus oleks võimalikult vähe häiritud ning ei tekiks pikaajalisi katkestusi. Areneval ettevõttel on kasulik üleminek teostada võimalikult varakult, sest mida suuremaks ja keerulisemaks infrastruktuur läheb, seda raskem on üleminekut tulevikus teostada. Antud töö käigus antakse ülevaade IPv6-st ja tema erinevatest ülemineku meetoditest. Töö tulemusena hakkab ettevõtte infrastruktuur kasutama IPv6-te koos kasutuselevõtu plaaniga.

Võtmesõnad:

IPv6, IPv4, võrguprotokoll, IPv4-st IPv6-le üleminek

CERCS: T120 Süsteemitehnoloogia, arvutitehnoloogia

Deploying IPv6 Astro Baltics Case Study

Abstract:

This thesis aims to transition the network infrastructure of Astro Baltics from IPv4 to IPv6 based on the specifics of the company. IPv6 is the successor of IPv4 that offers better security, is faster, and more futureproof as it provides more addresses to use. The topic is relevant because many are switching from IPv4 to IPv6 technology as it is considered a more modern protocol. Before the transition, everything must be prepared and tested to keep downtime to services as low as possible. For a growing company, it is better to do it as soon as possible rather than later because the more extensive and complex the network gets, the harder it is to transition later. The thesis will give a brief overview of IPv6 and IPv6 transition methods. As a result of this thesis, Astro Baltics will receive an implementation plan and start using IPv6.

Keywords: IPv6, IPv4, internet protocol, IPv4 to IPv6 transition

CERCS: T120 Systems engineering, computer technology

Sisukord

Lühendite loetelu

IP (*Internet Protocol*) – Sideprotokoll ehk reeglistik, mida jälgitakse andmepakettide saatmisel internetis kasutavate võrguseadmete vahel.

IPSec (*Internet Protocol Security*) – Protokoll, mida kasutatakse andmete krüpteerimseks seadmete vahel.

DNS (*Domain Name System*) – Võrguteenus, mis tõlgendab domeeninimed (nt. ut.ee) IP aadressiteks.

DHCP (*Dynamic Host Configuration Protocol*) – Protokoll, mis võimaldab seadmetel (DHCP klient) küsida võrguaadressi DHCP serveri käest.

NAT (*Network Address Translation*) – Meetod, mis tõlgendab sisevõrgu aadressid avalikuks IP aadressiks ja vastupidi.

Dual Stack – Ülemineku meetod, kus pannakse kaks erinevat IP (IPv4 ja IPv6) võrku kõrvuti (seadmetel mõlemas võrgus aadress suhtluseks).

AAAA kirje – DNS kirje, mis näitab domeeninime tõlgendust IPv6 aadressina.

A kirje – DNS kirje, mis näitab domeeninime tõlgendust IPv4 aadressina.

Võrguliides (*Network Interface)* – Riistvaraline seadme osa, mis ühendab arvuti arvutivõrguga.

MAC-aadress (*Media Access Control address*) – Unikaalne riistvaranumber, mis määratakse võrguliidese seadme tuvastamiseks võrgus.

DUID (*DHCP Unique Identifier*) – Seadme unikaalne identifikaator võrgus.

Kommutaator (*network switch*) – Võrguseade, mis ühendab seadmed omavahel arvutivõrgus ja saadab lähtehostist tulnud info ainult sihthostini viiva teekonna kaudu.

RA (*Router Advertisement*) – Marsruuteri poolt saadetav võrgu prefixi, prefixi eluiga, vaikelüüsi ja võrgu aadressi jagamise poliitika info.

ND (Neighbor Discovery) – Protokoll, mis tuvastab teiste seadmete olemasolu, nende IP aadressi, suudab otsida ka võrgu marsruutereid ja peab järge teiste seadmete kättesaadavause üle.

SLAAC (*Stateless address autoconfiguration*) – Aadressi saamise viis, kus seade ise genereerib endale IP aadressi vastavalt võrgu prefixile, mida ta saab RA kaudu.

1. Sissejuhatus

Tänapäeva interneti aluskihiks ehk vundamendiks võib lugeda IP protokolli pakette, mis võimaldavad arvutitel omavahel suhelda. Nad meenutavad oma vormi poolt ümbrikuid, kus on kirjas saatja ja saaja aadress koos sisuga. Praegu on suurem osa maailmas kasutusel paketi standard nimega IPv4.

Laaneoks (2010) on kirjutanud, et suurimaks IPv4 probleemiks hetkel on aga aadresside puudus. IPv6 protokoll on järglane IPv4 protokollile, mis toob endaga kaasa paketi struktuuri muudatuse. Suuremaks muutuseks on näiteks paketi suurus. IPv6 on 128 bitti pikk, samal ajal kui IPv4 on ainult 32 bitti pikk. See aga võimaldab mahutada paketi sisse rohkem aadresse ja sisu. Lisaks on IPv6 turvalisem ja kiirem. Turvalisemaks teeb IPv6 protokolli aga IPsec tugi ehk andmed on saatja ja saaja vahel krüpteeritud. Kiirusele aitab aga kaasa paketi muutumine lihtsamaks, mis tähendab, et võrguseadmed ei pea palju aega kulutama paketi käsitlemise peale.

Seega ettevõttele tähendaks IPv4 sisevõrgu pealt IPv6-le üleminek paremat turvalisust ja kiirust. Lisaks ka tulevikukindlust, et aadressid ei saa otsa juhul kui ettevõtte kasvab veelgi suuremaks. Antud töö eesmärk on ettevõtte võrgu infrastruktuur üle viia IPv4 protokolli pealt täielikult IPv6 protokolli peale ilma, et tekiks suuri teenuse katkestusi. Antud töö annab ülevaate IPv6-st ja selle kasutuselevõtmise protsessist.

Oluline on üleminek teha praegu kui hiljem, sest ajapikku muutub ülemineku tegemine ainult raskemaks. Selle taga on pidevalt muutuv võrk, mis ainult suureneb ja läheb lahendustega keerulisemaks. Ettevõtte üleviimise muudab keerulisemaks juba praegu asjaolu, et IPv6 on täielikult keelatud võrgus ja protokolli lubamiseks peab kõik seadmed võrgus ümber häälestama.

Antud teema on aktuaalne, sest järjest rohkem on maailmas hakatud üle minema IPv4 pealt IPv6 tehnoloogiale. Näiteks, kui jaanuaris 2010 oli Google-it külastavate IPv6 kasutajate osakaal maailmas 0.28 % , siis nüüdseks, jaanuar 2021 on see tõusnud umbes 30% peale. (Cisco 6lab veebilehekülg, 2021)

Töö on jaotatud kaheks osaks. Esimeses osas antakse ülevaade IPv6-st ja erinevatest ülemineku meetoditest. Teises osas antakse ülevaade kuidas plaaniti ja alustati IPv6 üleminekut. Kokkuvõttes analüüsitakse antud töö käiku ja võimalikke täiendusi tulevikus.

2. Tausta ülevaade

Antud peatükk annab ülevaate IPv6 erinevusest IPv4-st ja erinevatest ülemineku meetoditest. Peatükk jaotub kaheks alampeatükiks, adresseerimine ja ülemineku meetodid.

2.1 Adresseerimine

Selleks, et seadmed saaksid omavahel suhelda on seadmetel vaja võrgus oma aadressi. Seda jagab näiteks IPv4 võrgu puhul välja keskne olekuga DHCP server. Olekuga DHCP server jagab välja aadresse MAC-aadressi põhjal koos alamvõrgu infoga. Avalikule võrgule (internet) ligipääsemiseks jagab olekuga DHCP server lisaks ka infot vaikelüüsi ja DNS serveri kohta. Kui võrgus pole keskset olekuga DHCP serverit siis peab käsitsi IPv4 võrgus määrama kõik need parameetrid. (Laaneoks, 2010) . Frankel jt. (Frankel, Graveman, Pearce, & Rooks, 2010) on kirjutanud, et IPv6 võrguprotokolli kasutades suudab ise genereerida endale IP aadresse ehk olekuta aadresse määrata. IPv6-el kasutab selle jaoks ND (neighbor discovery) protokolli.

Neighbor discovery (ND)

Narten jt. (Narten, Nordmark, Simpson, & Soliman, 2007) on kirjutanud, et neighbor discovery ehk ND protokolli kasutavad võrguseadmed, et välja selgitada informatsiooni võrgu kohta millega nad parasjagu ühenduses on. ND kaudu saab võrguseade infot teiste võrgus olevate seadmete link-local aadresside kohta ja jälgib, et need oleksid koguaeg korrektsed. ND abil saab seade ülesse leida ka marsruutereid, mis on võimelised pakette edastama teistesse võrkudesse. Samuti jälgitakse ND abil aktiivselt, et mis seadmed on kättesaadavad, mis seadmed ei ole kättesaadavad ja tuvastatakse teiste seadmete link-layer aadressite muutusi. ND abil on võimalik ka seadmel teada saada kas aadress on juba kasutusel.

Router Advertisement (RA)

Narten jt. (Narten, Nordmark, Simpson, & Soliman, 2007) on kirjutanud, et Router advertisement ehk RA sõnumeid saadab marsruuter võrgu parameetrite ja enda olemasolust teadaandmiseks perioodiliselt või RS (Router Solicitation) sõnumi vastamiseks. RA sõnumid sisaldavad prefix infot mida kasutatakse IP aadressi kasutuseloleku kontrolliks ja IP aadressi määramiseks. Lisaks sisaldab ka RA sõnumid infot võrgu adresseerimis poliitika kohta. Näiteks on võimalik seadmetele teada anda kas kasutada DHCPv6-te ja/või olekuta automaatkonfiguratsiooni.

Olekuga DHCPv6

Mrugalski jt. (Mrugalski, et al., 2018) on kirjutanud, et olekuga DHCPv6 töötab sarnaselt IPv4 olekuga DHCP serverile, kus igale DHCP kliendile jagatakse välja kindel IP aadress. Seega jälgib DHCP server ise väljajagatud IP aadresse vastavalt DUID järgi. DUID on seadme unikaalne identifikaator võrgus. See asendab IPv4 MAC-aadressi, mis on võrguliidese põhine. Eriti hea on antud funktsionaalsus mitme võrguliidesega seadmete jaoks nt. sülearvutid, sest see võimaldab neil nii juhtmevaba võrgus kui ka kaablivõrgus saada kätte täpselt sama IP aadress.

Olekuta DHCPv6

Mrugalski jt. (Mrugalski, et al., 2018) on kirjutanud, et olekuta DHCPv6-te kasutatakse siis, kui soovitakse jagada DHCP kliendile ainult ühte või rohkem lisa võrgu parameetrit, näiteks DNS serveri info. Seega DHCP server ise IP aadresse välja ei jaga. Üldiselt kasutatakse olekuta DHCPv6 serverit koos SLAAC-iga, mis edastab kõik ülejäänu info, mis seadmel interneti pääsemiseks vaja on.

Olekuta automaatkonfiguratsioon (SLAAC)

Thomson jt. (Thomson, Narten, & Jinmei, 2007) on kirjutanud, et olekuta automaatkonfigureerimise puhul genereerivad endale seadmed ise IP aadresse. See tähendab, et DHCP serverit pole üldse võrgus vaja. Interneti pääsemiseks peab ainult marsruuter reklaamima RA (Router Advertisement) abil infot prefixi kohta. Seadmed genereerivad ise endale IPv6 avaliku aadressi vastavasse prefixi ja pääsevadki marsruuteri kaudu internetti.

2.2 Nimelahendus

Antud alampeatükk annab ülevaate DNS (Domain Name System) ehk nimelahenduse teenusest IPv4 ja IPv6 võrgus.

Selleks, et võrguteenuseid kasutada peab kasutaja teadma teenuse täpset IP aadressi. Küllaga on IP aadresse kasutajal keeruline meelde jätta. Frankel jt. (Frankel, Graveman, Pearce, & Rooks, 2010) on kirjutanud, et kasutaja kogemust lihtsamaks muuta kasutatakse DNS-i, mis tõlgendab domeeninimed ümber IP aadressiks. Ühel domeeninimel on aga võimalik määrata nii IPv4 kui ka IPv6 aadress. Selle jaoks kasutatakse domeeni kirjeid. IPv4 aadressi määratakse 32 bitise A-kirjetena. IPv6 puhul aga 128 bitise AAAA-kirjetena. DNS server vastab ainult selle kirje, mida tema käest parasjagu küsitakse. Seega kasutaja arvuti

ise peab otsustama, kas küsib A kirje, AAAA kirje või mõlemad. Vastavalt vastusele teab siis arvuti, kas teenusele on võimalik ligi pääseda.

2.3 Ülemineku meetodid

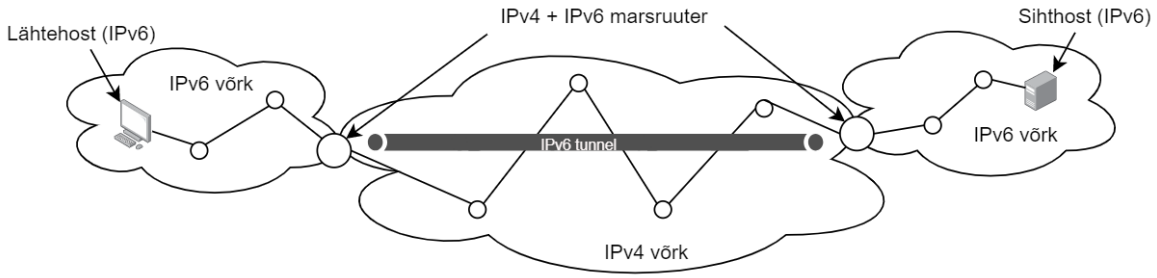
Antud alampeatükk annab ülevaate erinevatest ülemineku viisidest. Peatükk jaotub veel omakorda kolme eraldi alampeatükki: tunnelid, teisendamine ja dual-stack.

Abubakar Isa jt (2019) on maininud, et kuna IPv4 ja IPv6 on täiesti eraldiseisvad protokollid, siis ei suuda IPv4 ja IPv6 võrgud omavahel otse suhelda. Selleks, et lahendada antud probleem on väljatöötatud erinevad ülemineku meetodid. Need on jaotatud kolme peamisesse gruppi: tunnelid, teisendamine ja dual-stack.

Tunnelid

Abubakar Isa jt (2019) on kirjutanud, et tunnel on meetod, kus IPv6 pakett ümbritsetakse IPv4 paketti sisse, võimaldades liiklust edastada üle IPv4 võrgu. Saaja aga peab IPv4 protokolliga paketi lahti pakkima tagasi IPv6 paketiks. Siis saab IPv6 pakett sihtkohta edasi minna. (vt. Joonis 1) Laaneoksa (2010) sõnul on tunnelit võimalik häälestada käsitsi või automaatselt kasutades 6to4 meetodit. Kahjuks ei saa 6to4 meetodit kasutada NAT-iga ja selle kasutamise osas sõltud avalikest lüüsidest ehk vahendajatest, mis teeb side aeglasemaks ja lahenduse vähem töökindlamaks.

Meetodi illustratsiooni on näha Joonis 1 peal, kus on käsitsi häälestatud VPN tunnel. VPN tunnel luuakse marsruuteri poolt, millel on IPv4 ja IPv6 võimekus olemas. Lähtehost ja sihthosti vahelise IPv6 ühenduse toimimiseks loovad sihthosti ja lähtehosti marsruuterid VPN tunneli üle IPv4 võrgu. Seega kui lähtehost saadab pakette sihthostile, siis lähtehosti marsruuter ümbritseb IPv6 paketti IPv4 paketti sisse ja saadab üle IPv4 võrgu selle sihthosti marsruuterile, mis pakib IPv4 paketti tagasi lahti IPv6 pakettiks. Lahtipakitud pakett saadetakse edasi sihthostile.



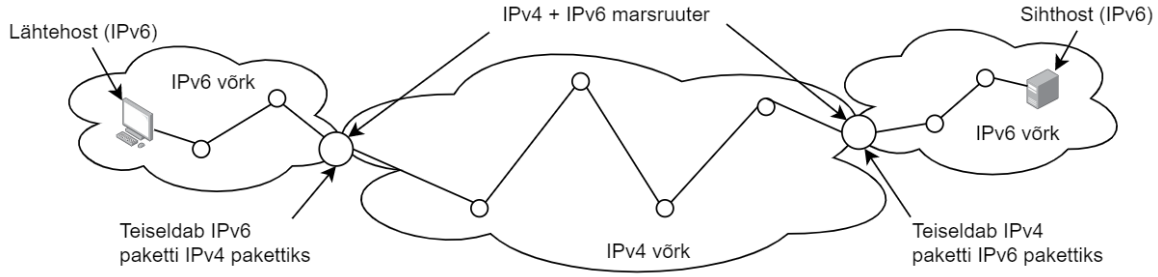
Joonis 1: IPv6 tunneli kasutamine üle IPv4 võrgu

Teisendamine

Abubakar Isa jt (2019) on kirjutanud, et teisendamine on meetod, kus IPv6 pakett teisendatakse ümber IPv4 pakettiks. See lahendab ära IPv6 ja IPv4 võrgu vahelise suhtlusprobleemi, sest keegi on vahepealne tõlkija.

Marcelo Bagnulo jt (2012) on kirjutanud, et seda on võimalik kasutusele võtta NAT64 ja DNS64 abiga. NAT64 tõlgendab paketid otse IPv4 ja IPv6 vahel. Samal ajal kui DNS64 tõlgendab domeeni nimetusi (nt. neti.ee) IPv6 aadressiks. Juhul kui domeenil on AAAA kirje olemas siis edastatakse see kohe IPv6 seadmele. Kui domeenil seda pole, siis DNS64 hangib domeeni A kirje ja teisendab selle ise AAAA kirjeks.

Meetodi illustratsiooni on näha Joonis 2 peal, kus IPv6 lähtehosti poolt saadetud paketid tõlgitakse IPv4 ja IPv6 võimekusega marsruuteri poolt. Marsruuter teisaldab NAT64-ga IPv6 paketti IPv4 võrgu jaoks sobilikuks, et oleks võimalik jõuda sihthosti marsruuterini, kuna vahepealne võrk toetab ainult IPv4 ühendusi. Sihthosti marsruuter teisaldab kasutades NAT64 IPv4 paketti tagasi IPv6 võrgule sobivaks ja saadab selle edasi IPv6 sihthostini.



Joonis 2: Teisendamise meetodi kasutamine

Dual-Stack

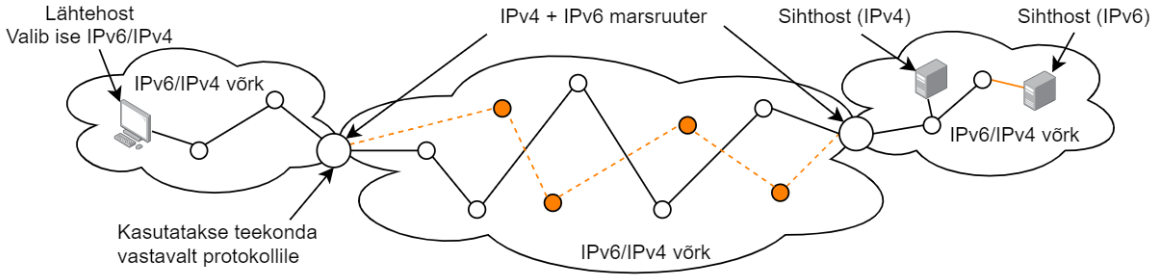
Abubakar Isa jt (2019) sõnul on Dual-Stack meetod, kus IPv6 võrk ja IPv4 võrk eksisteerivad kõrvuti. Seega on seadmetel võimalik kasutada mõlemat võrku.

Küllaga on Samheeth Malay jt. (2018) avastanud, et kahjuks Dual-Stack seadmed teevad alati AAAA ja A kirjete päringu, et valida sobiv aadress. See aga tähendab, et teenuse kasutamiseks läheb rohkem aega, sest ühe päringu asemel tehakse kaks. Alati oodatakse kuni mõlemad päringud saavad vastuse.

Sellisel juhul sobiks kasutada Laaneoksa (2010) poolt mainitud puhvermälu nimeserverit. Antud DNS server jätab vahemällu meelde vahendatud veebilehtede DNS päringud. See tähendab, et populaarsemate veebilehtede päringud, mida on varem juba külastatud, salvestatakse ajutiselt vahemälusse. Sellisel juhul kui võrgus olev seade soovib juba

vahemälus olevat veebilehte külastada, ei pea DNS server päringut tegema uuesti, et teada saada IP aadress. Selle asemel annab ta vastuseks vahemälus oleva IP aadressi.

Meetodi illustratsiooni on näha Joonis 3 peal, kus lähtehost omab IPv4 ja IPv6 aadressi. Tänu sellele peab lähtehost ise otsustama kumba protokolli kasutada. Lähtehost otsustab selle peamiselt DNS kirjete järgi, selle jaoks teeb ta nii AAAA kui ka A DNS päringu, et teada saada nii IPv4 kui ka IPv6 aadressi. Kui mõlemad kirjed on olemas siis kasutab lähtehost eelistatud protokolli, mis on defineeritud operatsioonisüsteemi enda poolt. Kui ainult üks kirje tagastab aadressi, siis kasutab lähtehost seda protokolli. Lähtehosti marsruuter kasutab teekonda, mis vastab lähtehosti poolt valitud protokollile. Sihthosti marsruuter samamoodi edastab saabunud paketi vastavale sihthostile saabuva paketti protokolli põhjal. Antud näitel on võimalik ühenduda kahe erineva sihthostiga, sest IPv4 või IPv6 toega sihthost ei pea otseselt olema sama masin. Dual-Stack võrk lubab mõlemale pakette saata.



Joonis 3: Dual-Stack meetodi kasutamine

Antud peatükk andis ülevaate IPv6-st, tema erinevusest IPv4-st ja erinevatest IPv6 ülemineku meetoditest. Järgmine peatükk annab ülevaate missugune ettevõtte infrastruktuur oli enne üleminekut ja kuidas hakkas üleminek IPv6 peale ettevõttes toimuma.

3. Üleminek

Antud peatükk räägib täpsemalt ettevõtte algseisust ja kuidas täpsemalt hakkas üleminek ettevõttes toimuma. Peatükk jaotub kaheks alampeatükiks, ettevõtte võrgu infrastruktuuri ülevaade ning ettevalmistus ja kasutuselevõtt.

Ettevõte otsustas, et sobiv meetod üleminekuks oleks Dual Stack. Dual Stacki topelt DNS päringute probleemi vastu hakkas aitama kohalik DNS server, mis puhverdab populaarsed päringud. Tunneli meetodit kasutusele aga ei plaanita võtta, sest see vajaks rohkem ressurssi haldamiseks ja lisaks liiga palju sõltuvust avalike lüüside peale. Teisaldamise meetodit ei saanud kasutusele võtta, sest Mikrotik võrguseadmete IPv6 pakett ei toeta NAT64 teenust (Mikrotik wiki - IPv6 overview, 2021). See tähendab, et võrguseadmed ei oska teisaldada IPv6 liiklust IPv4 liikluseks ja vastupidi.

3.1 Ettevõtte võrgu infrastruktuuri ülevaade

Antud alampeatükk annab ülevaate ettevõtte infrastruktuurist ennem IPv6 kasutuselevõttu.

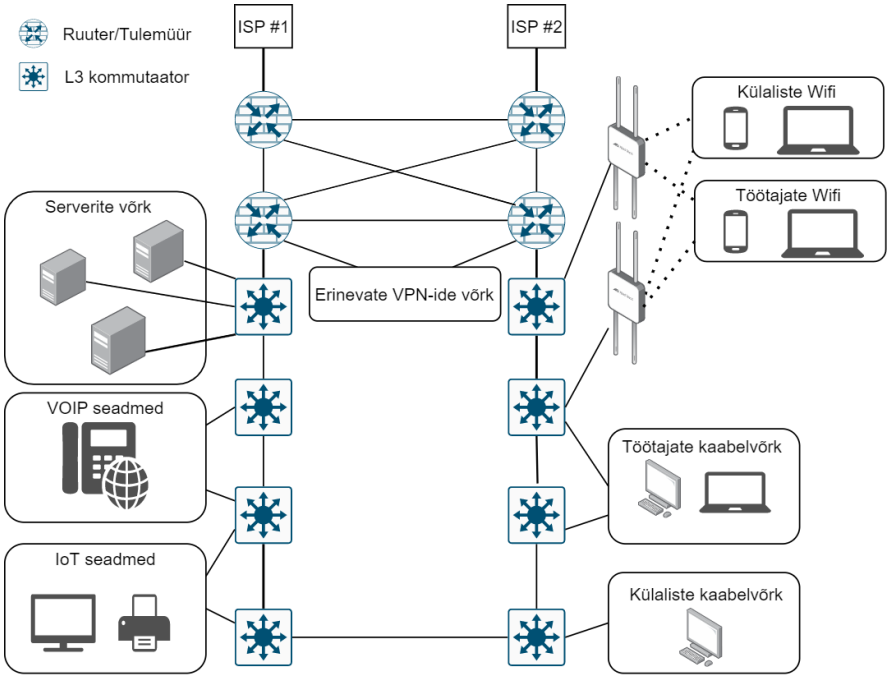
Ettevõtte kasutada on 20 IPv4 välisvõrgu aadressi. Kõik need aadressid on aktiivselt kasutusel ja kasutatakse rohkem veebilehtede ära mahutamiseks Proxy-sid.

Võrk ise toetab üle 50 töötaja, üle 65 erineva arvuti ja üle 100 erineva virtuaalserveri, mis on jagatud kolme füüsilise serveri peale. Lisaks on võrgus palju VOIP telefone, IoT seadmeid ja muid võrguseadmeid, mis on seotud ettevõtte arendustöö ja tegevusalaga. Ettevõtte võrk on virtuaalselt VLAN-ide abil segmenteeritud neljateistkümneks erinevaks virtuaalvõrguks. Nendeks on: ISP1 serverite võrk, ISP2 serverite võrk, IoT seadmed, töötajate kaabelvõrk, külaliste kaabelvõrk, töötajate wifi, külaliste wifi, VOIP seadmed, võrguseadmete haldusvõrk, töötajate VPN ja veel 4 erinevat serveriteenuste/vpn-ide jaoks mõeldud väikest võrku. Võrguseadmete toojaks on Mikrotik.

Lisaks on olemas kohalik DNS server koos tõrkesiirdega, mis on võrgus virtuaalmasinatena. Kohalikku DNS serverit ei saa ainukesena kasutada külalistele mõeldud võrgud ja VOIP telefonid, nendel on selle jaoks avalikud DNS serverid. IoT seadmed on aga täiesti internetist isoleeritud. Ettevõtte kasutab välise teenusepakkuja meiliserverit, millele puudub täielik ligipääse haldamiseks, ettevõtte haldab ainult kasutajaid ja meililiste.

Võrgu digitaalne topoloogia on väljatoodud Joonis 4 peal, kus on näha nelja tulemüüri ja kommutaatorid. Kommutaatorid on kõik L3 hallatavad ehk marsruudivad IP aadressi põhjal ja neid on võimalik ka konfigureerida. Lisaks on võrgus ka olemas juhtmevaba võrkude

jaoks wifi ligipääsupunktid ja on disainitud tõrkesiirde saavutamise põhimõttel. Joonisel on ka näha suuri kaste, mis iseloomustavad erinevaid virtuaalvõrke, mis on ettevõttel kasutusel.



Joonis 4: Astro Balticsi võrgu inftrastruktuur

Ennem kasutuselevõttu oli kogu võrgus IPv6 manuaalselt väljalülitatud ja keeltatud kõikidel võrgus olevates seadmetel. Põhjuseks on hea turvalisuse tagamise tava. Selle kohta on kirjutanud Sheila Frankel jt. (Frankel, Graveman, Pearce, & Rooks, 2010), et ettevõtted, kes veel ei kasuta IPv6, peaksid blokeerima kõik IPv6 liikluse ja lülitama välja IPv6 toe võrgus olevatel seadmetel. Ettevõttes on Windows 10 ja MacOS arvutid, serverid kasutavad Debian Linuxi. Linux serverite puhul oli IPv6 tugi väljalülitatud /etc/sysctl.conf failis

```
1.  net.ipv6.conf.all.disable_ipv6 = 1
```

parameetriga, Windowsi ja Mac masinate puhul oli käsitsi võrgu adapteritel IPv6 tugi väljavõetud. Võrguseadmetel aga polnud IPv6 paketti paigaldatud, seega ei osanud võrguseadmed IPv6 marsruutida ja liiklus ei läinud võrguseadmetest edasi.

3.2 Ettevalmistus ja kasutuselevõtt

Antud alampeatükk annab ülevaate IPv6 ülemineku jaoks tehtud ettevalmistusest ja kasutuselevõtust. Ülemineku osadeks saavad olema järgmised grupid: võrguteenused, võrguseadmed ja kasutajate seadmed.

Võrguteenused

Kuna meiliserveri teenust pakub ettevõttele väline teenusepakkuja, siis sõltub meiliserverile IPv6 toe lisandumine välise teenusepakkuja IPv6 kasutuselevõtmise ajakavast. Peamisteks kohalikeks võrguteenusteks saab ettevõtte võrgus olema DNS, DHCP, andmebaasiserverid, veebiserverid ja failiserver.

Kuna ettevõttel on algusest peale olnud kohalik DNS server, siis kõikide serverite aadressid on peidus DNS nime taga. Tänu sellele kasutatakse aktiivselt domeeninimesid. Ülemineku jaoks on vajalik ainult IPv6 päringutele hakata vastama ja lisada kohalikele domeenidele AAAA kirjed serverite jaoks, mis jooksvalt saavad IPv6 peale üle viidud. Testimise käigus tõrkeid DNS serveri teenustega ei esinenud, piisas sellest, et lülitasime sisse IPv6 toe DNS serveri operatsioonisüsteemil ja DNS teenusel.

DHCP serverina otsustas ettevõtte, et võrk peaks pigem hakkama kasutama olekuta DHCPv6-te. DHCP on ikkagi vajalik kuna ettevõtte soovib, et DNS ja domeeni info jõuaksid seadmeteni. Olekuga DHCP kasuks kahjuks ei saanud otsustada, sest Mikrotik ei toeta seda. Mikrotik toetab ainult IPv6 prefixite delegeerimist. (Mikrotik wiki - IPv6 overview, 2021)

Kuna ettevõtte peamiseks tegevusalaks on majandustarkvara arendus, siis on võrgus väga palju erinevate eesmärkidega andmebaasiservereid. Peamiselt on andmebaasimootoriks Firebird, kuna ettevõtte poolt arendatud majandustarkvara ise kasutab seda andmete hoidmiseks. Firebirdi dokumentatsioonis (FirebirdSQL Docs, 2016) on kirjas, et IPv6 aadressi kasutamiseks peab IPv6 aadressi ümbritsema kantsulgudega või kui kasutatakse domeeni nime, siis proovib Firebird automaatselt kõiki tagastatud IP aadresse kuni saavutab ühenduse andmebaasiga. IPv6-te aga toetab Firebird alates 3.0 versioonist. Seega vanemad andmebaasiserverid ei saa täiesti IPv6 peale üle minna. Lisaks kuna ettevõte kasutab serveriga ühendamiseks domeenimesid, siis ei pea lisatööd tegema, et saavutada ühendus. Uuemad (3.0 alates) Firebird andmebaasiserverid aga töötasid edukalt edasi IPv6 peal.

Veebiserverite puhul kasutab ettevõtte nii Apache-t kui ka NGINX platvormi. Apache ja NGINX oli ettevõttel kuulamas kõikide võrguliideste ja IP aadressite peal. Tähtis oli see, et päring tuleks õige domeeniga, kuna veebiserverid olid määratud „VirtualHost“-idena. Kuna sama konfiguratsiooni põhimõtet kasutati kõikides veebiserverites, siis oli vaja veebiserveritel ainult IPv6 lubada operatsioonisüsteemi tasemel ja veebiserveri teenusele restart teha, et teenus hakkaks IPv6 liidese peal ka kuulama.

Failiserverina kasutab ettevõtte peamiselt Samba-t, failserver asub töötajatega samas VLAN võrgus, ühtegi avalikku võrguketast pole, võrguketastega ühendamiseks on vaja autentida ennast. Vaikimise kuulab teenus kõiki võrguliideseid teatud pordi peal. Seega polnud antud teenuse puhul muudatusi vaja ja testides töötas teenus edukalt ka IPv6 kasutades.

Sidevõrguseadmed

Selleks, et võrgus oleks võimalik IPv6 võrguteenustele, nt failiserver ligi pääseda, peaks ennem olemas olema teekond seadmest seadmeni. Kommutaatorid on võrguseadmed, mida kasutatakse mitme seadme ühendamiseks võrgus. Kommutaatorid on just need seadmed, mis leiavad kõige sobivama tee teise seadmeni. Sellest tulenevalt saab esimeseks sammuks olema võrgukommunikaatorite häälestus IPv6 toetama. Mikrotik võrguseadmetel on olemas IPv6 pakett, mis lülitab sisse IPv6 toe võrguseadmetes. Juba konfigureeritud seadme puhul IPv6 paketti sisselülitamisel aga avastasime, et sellega ei tule kohe kaasa vaike konfiguratsiooni. Konfiguratsiooni saamiseks, pidime aga test seadmele tehase konfiguratsiooni taastamise tegema. Samal ajal kui IPv6 pakett oli sisselülitatud. Siis tekkis vaike konfiguratsioon, mida saime väljastada käskude nimekirjana.

```
1.  export file=test
```

Käskude nimekirjast sisestasime test võrguseadmele ainult IPv6 pakettiga seonduvad käsud, mis lisasid seadmesse IPv6 pakettiga seonduvad vaikesätted. IPv6 pakettiga seonduvad vaikesätted on väljatoodud töö lisana (Lisa 1 – Mikrotiki IPv6 tulemüüri vaikekonfiguratsioon). Antud meetod tuleb ka kasuks kasutusel olevate seadmete puhul, kuna saame testimises väljatöötatud konfiguratsiooni lisada olemasolevale konfiguratsioonile.

Kasutajate seadmed

Viimaseks ja kõige tähtsamaks osaks on võrgus kasutaja seadmed. Kasutaja seadmeteks loetakse tavaliselt arvuteid, mille taga inimesed töötavad ja kasutavad erinevaid võrguteenuseid ja sidevõrguseadmeid. Kõikidel kasutajate seadmetel tuleb aga IPv6 tugi sisse lülitada, et kasutajad saaksid kasu IPv6 võrgu olemasolust.

Windows seadmed

Kuna suur osa ettevõtte arvutitest kasutavad Windows operatsioonisüsteemi, siis oli vaja meetodit kuidas kõige efektiivsemalt Windows arvutitel IPv6 tugi tagasi sisse lülitada. Kaks varianti, kuidas Windowsis sätteid muuta oleks käsitsi kasutajaliidese kaudu või kasutades

automaatskripte/käskude abi. Kuna ettevõttel oli olemas Windows arvutite keskhaldus süsteem, siis osutus kõige efektiivsemaks meetodiks keskhalduse kaudu käsu käivitamine, sest see võimaldab korraga kõikidele arvutitele käsku rakendada. Keskhaldus käivitab ettevõttel käske administraatorina cmd.exe (käsuviip) abil. Sobilikuks käsuks osutus aga powershelli käsk. Powershell on käsurida enda programmeerimiskeelega, mis võimaldab ka hallata Windows platvormi erinevaid komponente. (Microsoft, 2021) Käsurea kaudu on võimalik powershell käivitada, et see omakorda käivitaks temale mõeldud käsu. Sobiv powershell käsk oli Enable-NetAdapterBinding, millele sai antud Name ja ComponentID parameetrid. Enable-NetAdapterBinding võimaldab aktiveerida Windowsi konkreetsel võrguliidesel ettenähtud funktsionaalsuse. (Microsoft, n.d.)

```
1.  powershell -command "Enable-NetAdapterBinding -Name * -ComponentID ms_tcpip6"
```

Käsu „Name“ parameeter täpsustab, mis nimega võrguliidesel tuleks aktiveerida teatud funktsionaalsus. (Microsoft, n.d.) Kuna vaja oli kõikidel võrguliidestel IPv6 tugi sisselülitada, siis kasutasime nime asemel metamärki *. Metamärk * annab powershellile juhise, et valida kõik nimetused. (Microsoft, 2021) Selle abil saab ka vältida arvutivahelisi erinevusi, kus võrguliidese nimed on erinevad nende arvu või lausa Windowsi regionaal sätete pärast. Käsu „ComponentID“ parameeter täpsustab, mis funktsionaalsust soovitakse aktiveerida võrguliidesel. (Microsoft, n.d.) Selle jaoks tuleb parameetrile määrata funktsionaalsuse ID. Selleks, et teada saada mis funktsionaalsuse ID tuleks määrata tuleb kasutada Get-NetAdapterBinding käsku. Selle käsu abil saab kätte kõik võimalikud komponendid adapteril koos nende ID-ga, mis ei sõltu Windowsi regionaal sätetest. (Microsoft, n.d.) Näitena on toodud, kuidas kasutada käsku „Ethernet“ nimelise võrguliidese puhul:

```
1.  Get-NetAdapterBinding -Name "Ethernet" -AllBindings
```

MacOS seadmed

Kõige väiksema osakaalu arvutitest on ettevõttes MacOS seadmed. Kuna neid on kõige vähem (alla kümne) siis oli kõige ajasäästlikum käsitsi seadmetele IPv6 sisselülitada.

Nutitelefonid

Nutitelefonide puhul selgus võrgu täpsema kaardistuse ajal, et tegelikult pole nutitelefonidel endil IPv6 võimekus juhtmevaba ühendusel väljalülitatud. Põhjuseks on asjaolu, et nii

töötajad kui ka külalised kastavad isiklike seadmeid. See teeb aga antud poliitika kasutamise kontrollimise keeruliseks. Selle asemel on kasutusel juhtmevabavõrgu poliitika, kus seadmed ise asuvad nö. liivakastis. Liivakastis olles saab seade ainult interneti, ei pääse ligi teistele arvutivõrkudele ja ei näe samas võrgus olevaid teisi seadmeid. Seega IPv6 tugi on juba nutitelefonidel olemas ja seadmeid ise konfigureerima ei pea.

Antud peatükk andis ülevaate ettevõtte infrastruktuuri seisust ennem üleminekut ja kuidas võttis ettevõtte kasutusele IPv6-e. Järgmine peatükk võtab kokku kogu töö teema.

4. Kokkuvõtte

Käesoleva bakalaureuse töö raames sai Astro Balticsi infrastruktuuri juurutatud IPv6 protokoll.

Töö tulemusena võttis ettevõtte kasutusele Dual Stack ülemineku meetodi. Täielikult ei saadud ainult IPv6 peale üle mindud, sest töö käigus selgunud Mikrotik võrguseadmete puudujäägid takistasid üleminekut. Lisaks sellele esinesid puudujäägid ettevõtte poolt kasutusel olevates majasiseselt arendatud erilahendusega võrguteenustes, mis ei toetanud täielikult veel IPv6-te. Tulevikus plaanib ettevõtte puudujääkidega lahendused viia ka üle IPv6 peale. Hetkeseisuga jääb ettevõtte infrastruktuur Dual Stacki peale.

Töö käigus suuri katkestusi ei esinenud, sest korralik eeltöö ja testimine tõid välja puudujäägid enne kasutuselevõttu. Kõige rohkem esines teenuse katkestusi IPv6 tuge lisavate pakettide paigaldusel, sest osasid teenuseid pidi taaskäivitama, et rakendada muudatusi.

Antud tööd on võimalik tulevikus kasutada kui näidismaterjalina head tava tutvustavas ülevaates. Kindlasti on võimalik antud teemat edasi arendada ka teistes keskkondades, kus näiteks võrguseadmeteks ei ole Mikrotik vaid nt. Cisco.

Viidatud kirjandus

Bagnulo, M., García-Martínez, A., & Beijnum, I. v. (2012, Juuli). The NAT64/DNS64 Tool Suite for IPv6 Transition. *IEEE Communications Magazine, 50*(7), 179-180.

Cisco 6lab veebilehekülg. (17. Jaanuar 2021. a.). Kasutamise kuupäev: 17. Jaanuar 2021. a., allikas https://6lab.cisco.com/stats/index.php?option=all

FirebirdSQL Docs. (4. Mai 2016. a.). Kasutamise kuupäev: 4. Mai 2021. a., allikas https://github.com/FirebirdSQL/firebird/blob/master/doc/README.IPv6

Frankel, S., Graveman, R., Pearce, J., & Rooks, M. (December 2010. a.). *Guidelines for the Secure Deployment of IPv6.* Gaithersburg, Ameerika Ühendriigid: National Institute of Standards and Technology. Allikas: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-119.pdf

Isa, A., & Abdulmumin, I. (2019). Design and Comparison Migration between Ipv4 and Ipv6 Transition Techniques. *In Proceedings: 3rd Annual International Conference of the Faculty of Science.* Yusuf Maitama Sule University, Kano.

Laaneoks, E. (2010). *Sissejuhatus võrgutehnoloogiasse.* Tartu Ülikooli Kirjastus.

Malay, S., & Kuncha, S. (2018). Q-DNS: Optimized Network Lookup for Dual Stack. *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)* (pp. 992-997). Exeter, United Kingdom: IEEE.

Microsoft. (2. Veebruar 2021. a.). *About Wildcards*. Kasutamise kuupäev: 5. Mai 2021. a., allikas Microsoft Docs: https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_wildcards

Microsoft. (22. Märts 2021. a.). *What is powershell?* Kasutamise kuupäev: 2. Mai 2021. a., allikas Microsoft Docs: https://docs.microsoft.com/en-us/powershell/scripting/overview

Microsoft. (kuupäev puudub). *Enable-NetAdapterBinding*. Kasutamise kuupäev: 5. Mai 2021. a., allikas Microsoft Docs: https://docs.microsoft.com/en-us/powershell/module/netadapter/enable-netadapterbinding

Microsoft. (kuupäev puudub). *Get-NetAdapterBinding*. Kasutamise kuupäev: 5. Mai 2021. a., allikas Microsoft Docs: https://docs.microsoft.com/en-us/powershell/module/netadapter/get-netadapterbinding

Mikrotik wiki - IPv6 overview. (8. Aprill 2021. a.). Allikas: https://wiki.mikrotik.com/wiki/Manual:IPv6_Overview

Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M. C., Jiang, S., . . . Winters, T. (2018). RFC 8415 - Dynamic Host Configuration Protocol for IPv6 (DHCPv6). Internet Engineering Task Force. Allikas: https://tools.ietf.org/pdf/rfc8415.pdf

Narten, T., Nordmark, E., Simpson, W. A., & Soliman, H. (September 2007. a.). RFC 4861 - Neighbor Discovery for IP version 6 (IPv6). Internet Engineering Task Force. Allikas: https://tools.ietf.org/html/rfc4861

Thomson, S., Narten, T., & Jinmei, T. (September 2007. a.). RFC 4862 - IPv6 Stateless Address Autoconfiguration. Internet Engineering Task Force. Allikas: https://tools.ietf.org/html/rfc4862

Lisa 1 – Mikrotiki IPv6 tulemüüri vaikekonfiguratsioon

```
1.  /ipv6 firewall address-list
2.  add address=::/128 comment="defconf: unspecified address" list=bad_ipv6
3.  add address=::1/128 comment="defconf: lo" list=bad_ipv6
4.  add address=fec0::/10 comment="defconf: site-local" list=bad_ipv6
5.  add address=::ffff:0.0.0.0/96 comment="defconf: ipv4-mapped" list=bad_ipv6
6.  add address=::/96 comment="defconf: ipv4 compat" list=bad_ipv6
7.  add address=100::/64 comment="defconf: discard only " list=bad_ipv6
8.  add address=2001:db8::/32 comment="defconf: documentation" list=bad_ipv6
9.  add address=2001:10::/28 comment="defconf: ORCHID" list=bad_ipv6
10. add address=3ffe::/16 comment="defconf: 6bone" list=bad_ipv6
11. add address=::224.0.0.0/100 comment="defconf: other" list=bad_ipv6
12. add address=::127.0.0.0/104 comment="defconf: other" list=bad_ipv6
13. add address=::/104 comment="defconf: other" list=bad_ipv6
14. add address=::255.0.0.0/104 comment="defconf: other" list=bad_ipv6
15. /ipv6 firewall filter
16. add action=accept chain=input connection-state=established,related,untracked
17. add action=drop chain=input comment="drop invalid" connection-state=invalid
18. add action=accept chain=input comment="accept ICMPv6" protocol=icmpv6
19. add action=accept chain=input port=33434-33534 protocol=udp
20. add action=accept chain=input dst-port=546 protocol=udp src-address=fe80::/10
21. add action=accept chain=input comment="accept IKE" dst-port=500,4500 protocol=udp
22. add action=accept chain=input comment="accept ipsec AH" protocol=ipsec-ah
23. add action=accept chain=input comment="accept ipsec ESP" protocol=ipsec-esp
24. add action=accept chain=input comment="accept ipsec policy" ipsec-policy=in,ipsec
25. add action=drop chain=input comment="drop everything not LAN" in-interface-list=!LAN
26. add action=accept chain=forward connection-state=established,related,untracked
27. add action=drop chain=forward comment="drop invalid" connection-state=invalid
28. add action=drop chain=forward comment="drop bad src ipv6" src-address-list=bad_ipv6
29. add action=drop chain=forward comment="drop bad dst ipv6" dst-address-list=bad_ipv6
30. add action=drop chain=forward hop-limit=equal:1 protocol=icmpv6
31. add action=accept chain=forward comment="defconf: accept ICMPv6" protocol=icmpv6
32. add action=accept chain=forward comment="accept HIP" protocol=139
33. add action=accept chain=forward comment="IKE" dst-port=500,4500 protocol=udp
34. add action=accept chain=forward comment="accept ipsec AH" protocol=ipsec-ah
35. add action=accept chain=forward comment="accept ipsec ESP" protocol=ipsec-esp
36. add action=accept chain=forward comment="accept ipsec policy" ipsec-policy=in,ipsec
37. add action=drop chain=forward comment="drop not LAN" in-interface-list=!LAN
```

Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, **Martin Mihkelsaar**

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose „**IPv6 juurutamine Astro Baltics näitel“,** mille juhendajad on **Toomas Lepik ja Artjom Lind** reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commonsi litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.

3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.

4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Martin Mihkelsaar

07.05.2021