TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Indrek Pihl 206691IAIB

AUTOMATIC AUDIT OF THE IMPLEMENTATION OF E-ITS SYS MEASURES IN THE ENVIRONMENT OF A FAMILY DOCTOR'S CENTER

Bachelor's Thesis

Supervisor: Toomas Lepik
Master's Degree

Tallinn 2024

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Indrek Pihl 206691IAIB

E-ITSI SYS MEETMETE RAKENDAMISE AUTOMAATSEIRE PEREARSTIKESKUSE KESKKONNAS

Bakalaureusetöö

Juhendaja: Toomas Lepik
Magistrikraad

Tallinn 2024

Autorideklaratsioon

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

Autor: Indrek Pihl

27.05.2024

Abstract

Automatic audit of the implementation of E-ITS SYS measures in the environment of a family doctor's center

The purpose of this work is to create an application that automates and accelerates self-auditing of E-ITS SYS measures and requires few resources from those who E-ITS subjects. The E-ITS profile for family doctor centers and Windows 11 are used as the basis for SYS measures to narrow the scope of the application.[1]

Process Monitor and Registry Editor is used to to determine which Windows 11 registry values need to be checked. [2, 3]

The thesis resulted with a back-end layer written in Java using the Spring Boot framework, and a front-end layer written with Typescript, Vue 3 and Vuetify framework. [4, 5, 6, 7, 8]

The thesis is written in Estonian and is 22 pages long, including 5 chapters, 7 figures and 2 tables.

Annotatsioon

Käesoleva töö eesmärk on luua rakendus, mis automatiseeriks ja kiirendaks eneseauditeerimist E-ITS SYS meetmete osas ning eeldaks väheseid vahendeid E-ITSi kohustuslaselt. Töö kitsenduseks on võetud Windows 11 ja perearstikeskuse keskkond, mistõttu võetakse aluseks E-ITSi profiil perearstikeskustele. [1]

Töö käigus kasutatakse Process Monitori ja Registry Editori, et kindlaks teha, milliseid Windows 11 registriväärtusi kontrollida tuleb. [2, 3]

Rakenduse teenuskiht kirjutatakse Javas, kasutades Spring Boot raamistikku. Esitluskiht kirjutatakse TypeScriptis, kasutades Vue raamistikku koos Vuetify kasutajaliidese teegiga. [4, 5, 6, 7, 8]

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 22 leheküljel, 5 peatükki, 7 joonist ja 2 tabelit.

Lühendite ja mõistete sõnastik

API	Application Programming Interface; rakendusliides
BSI IT-Grundschutz	Saksamaa Föderaalne infoturbeameti infoturbestandard
E-ITS	Eesti Infoturbestandard
ISO 27000	Rahvusvaheline standardite üldnimetus, mis käsitleb infoturbe juhtimissüsteeme (ISMS); Selle alla kuulub ka ISO 27001, mis on mõeldud neile, keda auditeeritakse, ja ISO 27002, mis on mõeldud audiitorile
Java	Platvormist sõltumatu keel
JavaScript	Esitluskihi objektorienteeritud programmeerimiskeel
OOP	Objektorienteeritud programmeerimine
RIA	Riigi Infosüsteemide Amet
Infoturbepoliitka	Infoturvapoliitika; neid termineid kasutatakse samaväärselt töös
ISMS	Information Security Management System ehk infoturbe halduse süsteem; Raamistik, mis hõlmab poliitikaid, protseduure ja ressursse, mis on vajalikud teabega seotud riskide tuvastamiseks, hindamiseks ja juhtimiseks.
Spring	Java platvormi raamistik
Spring Boot	Spring raamistiku osa, mis on eelseadistatud parimate tavade järgi
TypeScript	JavaScript keele ülemhulk, kus on lubatud staatilised tüübid
URI	Uniform resource identifier ehk ühtne ressursi-identifikaator; süntaksiga märgijärjend, mis identifitseerib mingi abstraktse või füüsilise ressursi

Sisukord

1 Sissejuhatus **9**
1.1 E-ITSi tutvustus 9
1.1.1 Infoturbe protsess 10
1.1.2 Infoturvapoliitika ja selle rakendamine 11
1.1.3 Infoturbeprotsessi riskihaldus 11
1.1.4 Infoturvameetmete rakendamine ja käigushoid 11
1.2 Miks tagada küberturvalisus perearstikeskuses? 12
1.3 Mida peaksid perearstikeskused väljaspool SYS meetmete rakendamist tegema, et tagada küberturvalisus? 12

2 Meetmete kaardistamine **15**
2.1 SYS meetmete läbivaatamine 15
2.1.1 Osalise kontrollitavusega meetmete esiletõstmine 15
2.2 Meetmete kaardistamise lõpptulemus 16

3 Rakenduse teenuskihi ja esitluskihi arendamine **17**
3.1 Java Spring Boot raamistiku valik 17
3.1.1 Kiirus ja tõhusus 17
3.1.2 Lai platvormi tugi 17
3.1.3 Aktiivne arendajate kogukond 17
3.2 Rakenduse teenuskihi disain ja arendusprotsess 17
3.3 Teenuskihi lõpptulemus 18
3.4 Vue.js ja Vuetify valik 20
3.4.1 Vuetify kasutajaliidese teek 20
3.4.2 Kogukonna toetus ja dokumentatsioon 20
3.4.3 Arendusjärjekord 20
3.5 Näide rakenduse teenuskihi ja esitluskihi kasutamisest 24

4 Tulevikusuunad ja soovitused **26**

5 Kokkuvõte **27**

Kasutatud kirjandus **28**

Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks **30**

Lisa 2 - Rakenduse arhitektuuri skeem **31**

Lisa 3 - Uue meetme lisamine rakendusse **32**

0.1 Täpsustamine, mida tuleb kontrollida 32

0.2 Teenuskihti implementeerimine 33

0.3 Vastava info lisamine esitluskihti 34

Lisa 4 - Valitud meetmed **35**

Joonistе loetelu

1 Algne kahe meetme vaade ning neist üks on avatud täpsema info saamiseks 21
2 IP-aadresside sisestuseks mõeldud lahter 22
3 Meetmete valimise jaoks loodud nupud, kus hetkel on valitud SYS21M1 ja SYS21M6 22
4 IPv4- ja IPv6-aadresside generaatorid, kus on just genereeritud IPv4-aadresse olemasolevate aadresside otsa 23
5 IP-aadressi 127.0.0.2 meetmetele vastavuse vaade 24
6 Näide arvutivõrgust, kus saaks rakendust kasutada 25
7 Rakenduse arhitektuuri skeem 31

Tabelite loetelu

1 Otspunktid ja nende täpsustus 19

2 Meetmed ja nende alampunktid 36

1. Sissejuhatus

Käesoleva bakalaureusetöö fookuses on küberturvalisuse tagamine perearstikeskuste keskkonnas, kus delikaatsete patsiendiandmete turvalisus on ülioluline. Üha kasvavad küberrünnakud ja andmete väärkasutamise ohud nõuavad tõhusaid meetmeid, et kaitsta tervishoiuasutuste süsteeme ja tagada patsientide konfidentsiaalsus. [9]

Bakalaureusetöö eesmärk on aidata kaasa perearstikeskuste küberturvalisuse parandamisele, pakkudes praktilist lahendust E-ITSi standardi jälgimiseks ning süsteemi turvalisuse tagamiseks. Autori osa seisneb uurimistöö läbiviimises, lahenduste väljatöötamises ja tarkvararakenduse arendamises vastavalt E-ITSi nõuetele. Oodatavaks tulemuseks on rakendus, mis aitab kaasa tervishoiuasutuste süsteemide turvalisuse suurendamisse.

Töö keskendub Eesti Infoturbe Standardi (edaspidi E-ITS) meetmete rakendamisele perearstikeskuste kontekstis. Rakenduse loomisel kasutatakse inseneeria meetodit. Selles töös kasutatakse sellest meetodist samme: uurimistöö tegemine, rakenduse nõuete seadmine, prototüübi loomine ja selle prototüübi täiendamine ja testimine, kuni on valminud lõplik versioon rakendusest. Thomas Lepik toob oma magistritöö "Eesti Infoturbestandardi turvameetmete rakendatuse automaatkontrolli põhimõtted" 2023 kokkuvõttes välja järgneva: "E-ITSi meetmete automaatkontrollimiseks sobilike turbemeetmete hulka hinnatakse ligikaudu üheks neljandikuks kõigist E-ITSi tingimustest. See tähendab, et ligikaudu 1650 tingimuse kontrollimise automatiseerimine võimaldab E-ITS rakendajatel märkimisväärset ressurssi kokkuhoidu." Sellest magistritööst sai autor inspiratsiooni rakenduse loomiseks. [10]

Töö käigus esmalt analüüsitakse E-ITSi SYS meetmeid perearstikeskuse töökeskkonnast lähtuvalt, seejärel uuritakse nende sobivust automaatkontrolliks ja lõpuks realiseeritakse rakendus sobivate turvameetmete kontrollimiseks. Lisaks töötatakse välja tarkvararakendus, mis kontrollib Windows 11 registreid vastavalt E-ITSi standardile ja pakub kasutajaliidest süsteemi turvalisuse jälgimiseks.

1.1 E-ITSi tutvustus

E-ITS on eestikeelne ja Eesti õigusruumile vastav alus infoturbe käsitlemiseks. E-ITSi aluseks on Saksa päritolu BSI IT-Grundschutz etalonturbe meetod. E-ITSi koostamisel on arvestatud vajadusega saavutada vastavus standardi EVS-EN ISO/IEC 27001 nõuetega.

Toimingud jagunevad kaheks: esiteks, äriprotsesside tüüpsete sihtobjektide vastavusse seadmine etalonmoodulitega ja teiseks, ebatüüpsete sihtobjektide puhul täiemahuline riskianalüüs ning vastavate infoturbe meetmete rakendamine. Perearstikeskused on Küberturbe seaduse alusel kohustatud järgima E-ITSi meetmeid. [11, 12]

E-ITSi eesmärk on tagada äriprotsesside ja infosüsteemide kõikehõlmav kaitse avalike ülesannete täitmiseks ning säilitada ühtlane infoturbe tase nende kõigi osade elutsükli vältel. E-ITSi haldamise eest vastutab RIA. Igal aastal vaadatakse E-ITS läbi ja uuendatakse, samal ajal jälgides, et see oleks kooskõlas teiste Eesti õigusaktidega. [13]

E-ITSi rakendatust hindab välisaudiitor ja standardturbe ulatuses vastab see rahvusvahelisele standardile ISO 27000. Siiski on E-ITSi rakendamine vajalik ISO 27000 sertifikaadi saamiseks. Eesti Küberturbe seaduse alusel on vastavad asutused kohustatud vastama kas E-ITSi meetmetele või ISO 27001 nõuetele. Näiteks nende kahe vahel saab tuua meetme ja nõude, mis on sarnase sisuga, kuid erinevalt sõnastatud. Kui E-ITSi meede SYS.2.2.3.M5 nõuab: "Kui IT-süsteemi kaitsmiseks kahjurvaraga nakatumise eest ei ole kasutusele võetud samaväärseid või rangemaid meetmeid, on Windows klientarvutis aktiveeritud Microsofti kahjurvaratõrje (nt Windows Defender) komponendid", siis ISO 27001 peatükis 8.7 on kirjas: "Protection against malware should be implemented and supported by appropriate user awareness" ehk eesti keeles: "Kaitse pahavara vastu tuleks rakendada ja toetada sobiva kasutajate teadlikkusega". ISMS on osa organisatsiooni juhtimisest, mis tegeleb infoturbe rajamise, edendamise, käigushoiu ja pideva täiustamisega, hõlmates struktuuri, protsesse, poliitikaid ja ressursse [14]

1.1.1 Infoturbe protsess

Organisatsiooni infoturve lähtub selle eesmärkidest ja tegutsemisviisidest, mida juhib infoturbe eestvedaja ehk tippjuhtkond. Tippjuhtkond algatab infoturbeprotsessi, nimetades infoturbejuhi ning vastutades infoturbetegevuse korraldamise eest. Infoturve keskendub äriprotsesside ja neist sõltuvate teabevarade kaitsmisele ning enne kaitsemeetmete rakendamist määratakse paika äriprotsessid ja nendega seotud varad. [13]

Tippjuhtkond vastutab äriprotsesside toimimise ja infoturbe eest ning peab saama regulaarselt teavet infoturberiskide, turvaintsidentide ja seaduslike nõuete kohta. Infoturbejuht plaanib ja koordineerib turbealaseid tegevusi, protsessijuht vastutab aga äriprotsessi infoturbe meetmete rakendamise eest. Infoturbeprotsessi kaasatakse kõik töötajad, kes saavad ISMSi rakendamiseks vajalikku motivatsiooni ja koolitust. [13]

1.1.2 Infoturvapoliitika ja selle rakendamine

Infoturvapoliitika on organisatsiooni juhtdokument, milles määratakse kindlaks infoturbe kohustused, eesmärgid ja kaitstavad väärtused. Seda kasutatakse organisatsiooni väärtuste kaitsmiseks ning see koostatakse, võttes arvesse organisatsiooni eesmärke, struktuuri, seadusandlust ning huvipoolte nõudeid. Infoturvapoliitika kinnitab juhtkond ja seda tuleb perioodiliselt üle kontrollida ning vajadusel ajakohastada. Samuti peab organisatsioon tagama töötajatele infoturvapoliitika teadlikkuse ning neid sellest dokumendist teavitama. [13]

Organisatsioon peab tagama, et kõik töötajad omavad vajalikku infoturbealast pädevust ning et seda uuendatakse regulaarselt vastavalt infoturvapoliitikale ja eesmärkidele. Koolituste eesmärk on motiveerida töötajaid infoturbenõudeid järgima ja teavet ning töövahendeid korrektselt käsitlema. Juhtkond eraldab ressursse infoturbeprotsessi elluviimiseks ja kinnitab infoturvapoliitika ning ressursside eraldamise. Infoturbeprotsessi jälgitavuse tagamiseks protseduurid ja juhendid dokumenteeritakse ning organisatsioon kehtestab dokumendihalduse reeglid, mis hõlmavad ka dokumentide ajakohastamist ja ülevaatamist. [13]

1.1.3 Infoturbeprotsessi riskihaldus

Infoturbeprotsessi riskihalduse raames lähtutakse etalonturbe kontseptsioonist, mille kohaselt on etalonturve standardsete ohuolukordade ja kaitstavate varade puhul efektiivne. Organisatsioon määrab kaitsetarbe põhimõtted ja tuvastab kaitset vajavad objektid, sh tellitud teenused. Riskihaldusprotsessi käiku ja meetmete rakendamist dokumenteeritakse, et tagada jälgitavus ja võrreldavus. Etalonturbe modelleerimise käigus ühendatakse organisatsiooni vajadused etalonturbe kataloogi moodulitega, võttes arvesse turvameetmete olulisust ja seost protsessi ja elutsükliga. Etalonturbe välises riskihalduses keskendutakse objektidele, mis ei sobi etalonturbe moodulitega või mille kaitsetarve on suur või ebaselge. Selle protsessi määrab organisatsioon, dokumenteerides riskihalduse kulgu ja tulemusi. Lisaturvameetmeid rakendatakse vastavalt kaitsetarbele ja infoturbe eesmärkidele. [13]

1.1.4 Infoturvameetmete rakendamine ja käigushoid

Infoturvameetmete rakendamine hõlmab nii tehniliste kui ka organisatsiooniliste meetmete kohandamist kõikidele kaitseala sihtobjektidele. Rakendamise eest vastutab vastava protsessi juht või etalonturbe kataloogis nimetatud vastutaja, keda nõustab infoturbejuht. Organisatsioon hindab meetmete kulutusi ja sobivust ning määrab infoturbe meetmete

rakendusplaani, juhindudes infoturbe eesmärkidest. Juhtkond kinnitab rakendusplaani ja tagab selle järgimise. Meetmete rakendamisel jälgitakse ja tagatakse nende tõendatavus, võrreldavus ning õigeaegsus vastavalt dokumenteeritud korduvatele ja regulaarsetele tegevustele. Infoturbe käigushoid hõlmab protsesside ja süsteemide jälgimist ning reageerimist muutustele infoturbeolukorras, regulatsioonides ja lepingulistes kohustustes. Samuti dokumenteeritakse seire, ülevaatuste ja kontrollide tulemused: neid kasutatakse ettepanekuteks infoturbeprotsessi ja -meetmete täiustamiseks. Lisaks teavitatakse olulistest infoturbeprotsessi käigus ilmnenud asjaoludest vajalike rollide täitjaid, organisatsiooni töötajaid ning teisi huvipooli. [13]

1.2 Miks tagada küberturvalisus perearstikeskuses?

Perearstikeskused on oluline osa tervishoiusüsteemist. Nende küberturvalisus on äärmiselt vajalik, kuna seal hoitakse ja töödeldakse delikaatseid meditsiinilisi andmeid ning tehakse elukvaliteeti mõjutavaid otsuseid. E-ITSi rakendamine perearstikeskustes aitab tagada, et kõik süsteemid ja protsessid vastavad kõrgetele turvastandarditele. See omakorda tagab patsientide andmete konfidentsiaalsuse, tervikluse ja kättesaadavuse ning vähendab küberohtude riski. Eeldusest, et enamus perearstikeskusi ei oma püsivat infoturbe spetsialisti ja ei talleta patsiendi andmeid ega diagnoose enda juures olevatesse serveritesse, vaid tellivad neid rakendusi ja salvestusvõimalusi sisse eraettevõtjatelt, tuleneb perearstikeskuste tähtsus küberturvalisuse raames. Ühe kehva küberturvalisusega perearstikeskuse pihta tehtud eduka küberrünnaku tagajärjel võib too perearstikeskus saada nii öelda hüppelauaks, et rünnata edasi neid eraettevõtteid, kellel on enamuse Eesti elanike meditsiinilised andmed. Juhul, kui perearstikeskus ei järgi E-ITS meetmeid, on RIA-l õigus Küberturbe seaduse alusel trahvi teha sellele perearstikeskusele. [9]

1.3 Mida peaksid perearstikeskused väljaspool SYS meetmete rakendamist tegema, et tagada küberturvalisus?

Lisaks SYS meetmetele, mille järgimist saan töö käigus loodud rakendusega testida, on oluline, et perearstikeskused võtaksid küberturvalisuse tagamiseks ette ka muid E-ITSi poolt neile ette nähtud meetmed. Järgnevalt välja toodud tegevusi ei peaks läbi viima perearst või pereõde ise, vaid need tuleks tellida vatavatelt asjatundjatelt.

Regulaarsed andmekaitsealased koolitused töötajatele on kriitilise tähtsusega perearstikeskuse küberturvalisuse tagamisel. Need koolitused aitavad suurendada töötajate teadlikkust küberturvalisuse parimatest tavadest ja võimalikest ohtudest, mis omakorda vähendab inimeksimustest tulenevaid turvariske. Koolituste käigus tuleks käsitleda mitmeid olulisi

teemasid, sealhulgas isikuandmete kaitse põhimõtteid, turvalise parooli loomist, paroolihalduse praktikaid, tundliku teabe töötlemise ja edastamise turvalisust, sotsiaalse manipuleerimise ohtude tuvastamist ja nendega tegelemist ning kiiret ja korrektset reageerimist turvaintsidentide korral. Samuti peaks koolitustel tutvustama kehtivaid seadusi ja regulatsioone, mis puudutavad tervishoiuvaldkonna andmekaitset. Läbimõeldud ja regulaarsed koolitused aitavad töötajatel mõista oma rolli ja vastutust küberturvalisuse tagamisel, luues sellega tugeva esimese kaitseliini võimalike küberohtude vastu. [12]

Samuti tuleb kasuks ka selgete protseduuride loomine andmete varundamiseks ja hädaolukorrast taastumiseks. Need protseduurid tagavad, et rünnaku või andmelekke korral saab süsteemi kiiresti ja tõhusalt taastada, vähendades seeläbi tegevuse katkestusi ja võimalikke andmekadusid. Varundamisprotseduurid peavad hõlmama regulaarset andmete varundamist, varukoopiate turvalist hoiustamist nii kohapeal kui ka väljaspool asutust ning varunduste regulaarset testimist taastatavuse tagamiseks. Hädaolukorrast taastumise protseduurid peavad sisaldama selget tegevusplaani, mis kirjeldab samm-sammult süsteemide taastamise protsessi, määrab vastutavad isikud ja tagab kommunikatsiooni kõikide asjaosaliste vahel. Nende protseduuride dokumenteerimine ja regulaarne läbivaatamine aitab tagada, et kogu personal on teadlik oma rollist ja oskab hädaolukorras kiiresti ja tõhusalt tegutseda. Selline tegutsemine tagab küll kohalike tööjaamade taaste, ent terviklikus mudelis peab perearst nõudma varunduse ja taastamise võimekust ka oma IT teenusepakkujatelt. Sellised kohustused ja ka kohalike tööjaamadega seotud varundus peab olema lepinguliselt reguleeriutud, kui tööjaamade haldus on sisse ostetud. [12]

Unarusse ei tohiks jätta ka regulaarseid haavatavuste skaneeringud. Need meetmed aitavad tuvastada ja kõrvaldada võimalikud turvavead enne, kui need saavad muutuda tõsisteks ohtudeks. Haavatavuste skaneeringud kasutavad spetsiaalseid tööriistu ja tehnikaid, et avastada tarkvaras ja riistvaras peituvad nõrkused. Neid skaneeringuid tuleks läbi viia regulaarselt, vähemalt kord kvartalis, ning iga kord, kui tehakse süsteemimuudatusi. Avastatud haavatavustele tuleb kiiresti reageerida, rakendades asjakohaseid turvapaikasid ja -parandusi. Regulaarsete kontrollide ja skaneeringute abil saavad perearstikeskused ennetada turvaintsidente ja tagada oma süsteemide püsiva kaitstuse. [12]

Kokkuvõttes on perearstikeskuste küberturvalisuse tagamine mitmetahuline protsess, mis nõuab põhjalikku ja järjepidevat lähenemist. Regulaarne andmekaitsealane koolitus aitab töötajatel omandada vajalikke teadmisi ja oskusi, et ära tunda ja vältida võimalikke küberohtusid. Selgete protseduuride loomine andmete varundamiseks ja hädaolukorrast taastumiseks tagab, et süsteemid saavad kiirelt taastatud ja töö jätkub sujuvalt ka pärast küberintsidenti. Regulaarsete süsteemikontrollide ja haavatavuste skaneeringute abil on võimalik ennetada turvaauke ja hoida süsteemid turvalisena. Terviklik lähenemine, mis

hõlmab nii tehnilisi kui ka inimfaktoreid, on võtmetähtsusega, et tagada patsiendiandmete kaitse ja süsteemi terviklikkus pidevalt muutuvas küberturvalisuse maastikus ja kindlasti ei arva töö autor, et seda peab perearst ise tegema, ent selle sisseostmise kohustus perearstil on.

2. Meetmete kaardistamine

RIA E-ITSi veebilehel on kättesaadav spetsiaalselt perearstikeskuste jaoks loodud profiil, mis sisaldab meetmeid ja nõudeid, mis on olulised tervishoiuasutuste infoturbe tagamiseks. Profiili peamine eesmärk on keskenduda patsientide andmete turvalisusele ning see hõlmab erinevaid meetmeid, mis on kohandatud perearstikeskuste eripäradega. [1, 15]

Selle peatüki raames vaadatakse E-ITSi profiil perearstikeskustele põhjalikult läbi ning tehakse ülevaade kõigist SYS meetmetest, mis on tervishoiuasutuste küberturvalisuse tagamiseks vajalikud. Arvesse võetakse kõik eelnevalt mainitud profiilis esinevad SYS meetmed ja analüüsitakse nende automaatset kontrollitavust perearstikeskuste keskkonnas.

2.1 SYS meetmete läbivaatamine

Profiili läbivaatamise käigus loeti kokku 54 SYS meedet. Neist 20 olid seotud Windows 11 tööarvutitega. Ülejäänud 34 käisid serverite, mobiilseadmete, printerite, kontorikombainide ja irdandmekandjate kohta. Neist kahekümnest meetmest 10 olid automaatselt kontrollitavad ja 10 mitte. Üldjoontes need meetmed, mis ei olnud automaatselt kontrollitavad, nägid ette mingeid tegevusi, mis ei olnud nii selgelt sõnastatud, ja seetõttu võib iga perearstikeskus neid enda tingimustele vastavalt tõlgendada. Näiteks saab tuua SYS.2.1.M16, mis käsitleb tarbetuid mooduleid, programme, teenuseid, liideseid ja kasutajakontosid, aga kuna pole võimalik üldistatult defineerida, millised moodulid, programmid, teenused, liidesed ja kasutajakontod on tarbetud ja millised mitte, siis on raske seda automaatselt kontrollida.

2.1.1 Osalise kontrollitavusega meetmete esiletõstmine

Lisaks märgiti ära need SYS meetmed, mis on osaliselt kontrollitavad, näiteks SYS.2.1.M1 - Kasutajate turvaline autentimine. Antud näites on võimalik kontrollida meetme kõiki alampunkte peale punkti: "e. Pikema eemaloleku puhul logib kasutaja end klientarvutist välja või sulgeb arvuti.", kuna see nõue hõlmab kasutaja käitumist, mitte arvuti seadistust.

Samuti tuleb märkida, et mõnest meetme alampunktist, mis oli mitmeosaline, oli võimalik kontrollida vaid mõnda lauset alampunktist ning seetõttu märgistati need lause järjekorra numbriga. Näiteks meetme SYS.2.1.M1 puhul d) alaosas:

d.2 Kasutaja eemaloleku ajaks lukustab kasutaja arvuti juurest lahkudes ekraani.

d.4 Ekraaniluku avamine on võimalik vaid kasutaja autentimisega.

Selline nummerdamine aitab selgitada, milliseid konkreetseid osi meetme alampunktist on mõeldud, lisaks võimaldab see täpsemat jälgimist ja dokumenteerimist vastavalt E-ITSi nõuetele.

2.2 Meetmete kaardistamise lõpptulemus

Kokkuvõttes pakub E-ITS-i profiil perearstikeskustele raamistikku ja juhiseid küberturvalisuse tagamiseks vastavalt kehtivatele standarditele. Siiski võib mõnda meedet olla IT-hariduseta inimesel keeruline hinnata. Näiteks meede SYS.2.2.3.M9 sätestab: "Kesksekskes autentimiseks kasutatakse Kerberost. Kui seda ei tehta, võib alternatiivina kasutada autentimisprotokolli NTLMv2." Sellist tingimust ei ole Windowsi seadete kaudu intuitiivne kontrollida ning selle meetme kontrollimine ja süsteemi vastavalt seadistamine nõuaks registrite muutmist. Seetõttu oleks kasulik kasutada rakendust, mis automaatselt kontrollib neid registriväärtusi. Meetmete regulaarne läbivaatamine ja kontrollimine on oluline patsientide andmete turvalisuse tagamiseks ning aitab suurendada tervishoiuasutuste süsteemide vastupidavust küberohtudele. Kõigi valitud meetmete tabel on peatükis Lisa 4.

3. Rakenduse teenuskihi ja esitluskihi arendamine

3.1 Java Spring Boot raamistiku valik

Teenuskihi arendusprotsessi alguses olid kaalumisel Spring Boot ja Python Flask, aga otsustati Spring Booti kasuks, lähtudes autori eelnevast kogemusest, ülikooli akadeemilisest suunamisest ja mõlema raamistiku tugevustest ja nõrkustest. [5, 16]

3.1.1 Kiirus ja tõhusus

Spring Boot pakub kiiret ja tõhusat rakenduse arendust tänu oma integreeritud abstraktsioonikihtidele ja väljakujunenud konventsioonidele. See võimaldab arendajatel keskenduda rakenduse loogikale ja äriprotsessidele, vähendades samal ajal arendusaega ja -pingutusi. Python Flaskil selline sisseehitatud tugi ja struktuur puudub, see aga võib põhjustada arendusprotsessis rohkem käsitsi tööd ja seetõttu pikendada arendusaega.

3.1.2 Lai platvormi tugi

Spring Boot ja Python Flask pakuvad mõlemad laia platvormi tuge, mis võimaldab rakenduse arendamist mitmesugustele operatsioonisüsteemidele ja keskkondadele. See tuleb kasuks töö laiendamisel teistele operatsioonisüsteemidele. Samuti tagab see platvormidevahelise ühilduvuse ja rakenduse laialdase kasutuselevõtu erinevates töökeskkondades.

3.1.3 Aktiivne arendajate kogukond

Leiti, et Spring Booti taga on aktiivne ja toetav kogukond, mis pakub palju ressursse, dokumentatsiooni ja tuge rakenduse arendajatele. See võimaldab kiiret probleemide lahendamist ja vajaliku toe saamist arendusprotsessi igas etapis. Python Flaskil esines samaväärne aktiivsete arendajate kogukond, kuid Spring Bootiga võrreldes oli see väiksem ning pakkus vähem ressursse.

3.2 Rakenduse teenuskihi disain ja arendusprotsess

Rakenduse teenuskihi disaini poolest oli selge, et teenuskiht peab töötama vastavas arvutis, mida kontrollitakse. Teisel juhul kui panna tööle see mingis virtuaalmasinas, mis on

samas arvutis, mida tuleks kontrollida, siis oleks rohkem tegevust, et virtuaalmasinast teha päringud emuleeritud arvuti pihta, mille tõttu aeglustuks ka teenuskihi vastamise kiirus.

Rakenduse teenuskihi arendusprotsessi esimene samm oli põhjalik meetmete kogumine ja analüüs, kus määratleti rakenduse põhifunktsioonid ja kasutusstsenaariumid. Seejärel loodi üldine arhitektuur ja tehniline kavand, valides Java Spring Boot raamistiku. See vastas nõudmistele kiire arenduse, kerge edasiarenduse ja laia platvormitoe osas.

Järgnevalt keskenduti rakenduse põhifunktsionaalsuse loomisele. Meetmete otspunkte lisati ükshaaval ehk kõigepealt täpsustati, milliseid registreid tuleb kontrollida ja mis kujul sealt vastavad väärtused tulevad ning siis alles hakati koodi kirjutama selle otspunkti jaoks. Üheks suuremaks väljakutseks oli õigete Windowsi registrite leidmine ja kontrollimine, mis võttis ootamatult palju aega. Selleks kasutati mitmeid tööriistu, sealhulgas Process Monitori ja Registry Editori, et leida vajalikku teavet ja ressursse. [2, 3]

Pärast registrite kindlaksmääramist jätkati Powershelli käskude ja Osquery päringute integreerimise ning andmete töötlemise ja salvestamisega. Arendusprotsessi käigus jälgiti pidevalt rakenduse stabiilsust ja vastavust nõuetele, katsetades ja testides iga uut otspunkti. [17]

Esitluskihi ja teenuskihi arendus käis paralleelselt rakenduse põhifunktsionaalsusega, võimaldades lõppkasutajatel süsteemi andmetele ja funktsioonidele lihtsat juurdepääsu. Rakendus viidi kasutatavasse olekusse pärast põhjalikku arendus- ja testimisprotsessi ja tagati rakenduse vastavus nõudmistele ja kvaliteedistandarditele.

Kokkuvõttes oli rakenduse arendusprotsessi peamine fookus pideval enesearendusel ja standarditele vastavusel. See tagas rakenduse kõrge kvaliteedi ja vastavuse E-ITS SYS meetmetele.

3.3 Teenuskihi lõpptulemus

Teenuskihi arenduse lõpuks valmis 11 otspunkti, mis on väljatoodud Tabelis 1. Neist kümme vastab igaüks ühele E-ITS SYS meetmele kas osaliselt või täielikult ning üks otspunkt on ühenduse kontrollimiseks esitluskihiga. REST API otspunktide loomisel kasutati meetmete koode. Kõik päringud on GET tüüpi ehk need tagastavad lihtsalt informatsiooni esitluskihile ilma, et esitluskiht peaks mingit infot kaasa andma päringule. Iga otspunkti URI hakkab `/api` levinud tava kohaselt ehk esimene on täispikkuses `/api/`.

Tabel 1. Otspunktid ja nende täpsustus

URI	Päringu kirjeldus
/	Tagastab lihtsalt "Hello" testimaks, et ühendus on olemas
/SYS21M1	Tagastab andmesaateobjekti muutujatega: screenSaverIsEnabled, screenSaverPasswordProtected, needAuthToChangePassword, autoLogonIsDisabled, baseObjectsAreAudited
/SYS21M3	Tagastab andmesaateobjekti muutujatega: automaticUpdatingEnabled, checkForUpdatesDailyEnabled, controlUpdateServerAuthenticity, checkUpdatePackagesIntegrity, usesWSUS, previousStateIsRestorable
/SYS21M6	Tagastab andmesaateobjekti muutujatega: antiMalwareEnabled, antiMalwareUpToDate
/SYS223M4	Tagastab andmesaateobjekti muutujatega: telemetrySendingDisabled, telemetrySendingDisabledByFirewall
/SYS223M5	Tagastab andmesaateobjekti muutujatega: firewallEnabled, antivirusEnabled, firewallUpToDate, antivirusUpToDate
/SYS223M9	Tagastab andmesaateobjekti muutujaga: KerberosOr NTLMv2Enabled
/SYS223M13	Tagastab andmesaateobjekti muutujatega: smartScreenEdgeDisabled, smartScreenPuaDisabled
/SYS223M14	Tagastab andmesaateobjekti muutujaga: cortanaDisabled
/SYS223M18	Tagastab andmesaateobjekti muutujatega: allRemoteAssistanceRulesAreAllowed, remoteAssistanceDCOMInTCPNoScopeActive, remoteAssistanceRAServerInTCPNoScopeActive, remoteAssistancePnrpSvcUDPInEdgeScope, remoteAssistancePnrpSvcUDPInEdgeScopeActive, remoteAssistanceSSDPSrvInUDPActive, remoteAssistanceInTCPEdgeScope, remoteAssistanceSSDPSrvInTCPActive, remoteAssistanceInTCPEdgeScopeActive
/SYS223M19	Tagastab andmesaateobjekti muutujatega: allRDPRulesAreAllowed, remoteDesktopShadowInTCP, remoteDesktopUserModeInTCP, remoteDesktopUserModeInUDP

3.4 Vue.js ja Vuetify valik

Esitluskihi arendamisel valiti Vue.js raamistik peamiselt selle kasutusmugavuse, jõudluse ja fakti põhjal, et töö autor on selle raamistikuga kõige rohkem tuttav. Vue.js on võrdlemisi uus raamistik, mis pakub arendajatele dünaamiliste kasutajaliideste loomiseks võimsat tööriista. Selle lihtne ja intuitiivne süntaks võimaldab kiiret arusaamist olemasolevast koodist ja kerget juurdearendust. [7]

3.4.1 Vuetify kasutajaliidese teek

Lisaks Vue.js-le valiti ka Vuetify, mis on Vue.js-le spetsialiseerunud kasutajaliidese teek. Vuetify pakub laia valikut valmis komponente ja kujundusmalle, mis kiirendavad oluliselt arendusprotsessi. Need komponendid on mitmekülgsed, esteetilised ja täielikult kohandatavad, võimaldades luua kasutajaliideseid, mis vastavad täpselt rakenduse nõudmistele ja disainieelistustele. [8]

3.4.2 Kogukonna toetus ja dokumentatsioon

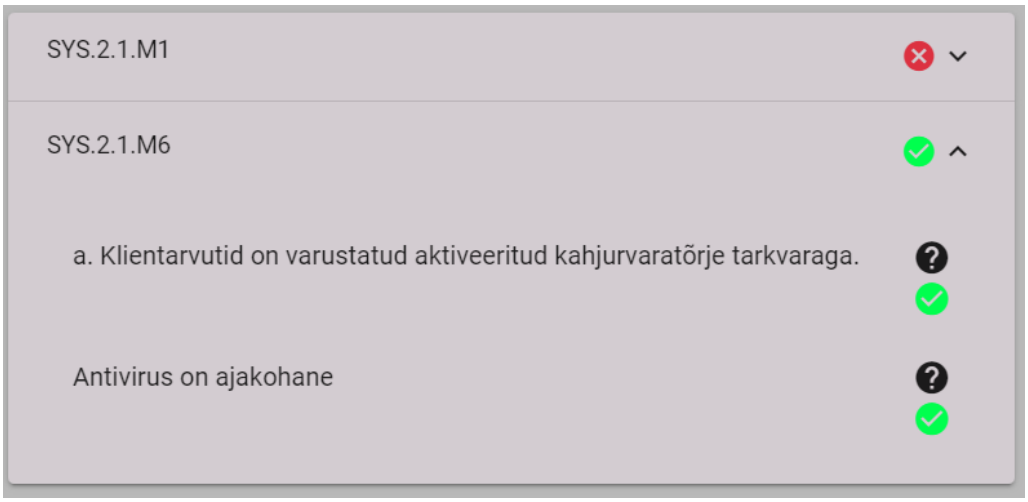
Vuetify valimisel arvestati ka selle kogukonna toetuse ja dokumentatsiooni kättesaadavusega. Laialdane kogukond ja hästi dokumenteeritud ressursid tagavad, et arendajatel on alati juurdepääs vajalikele juhistele ja tugimaterjalidele. See omakorda kiirendab arendusprotsessi ja vähendab võimalikke takistusi.

Kokkuvõttes võimaldavad Vue.js ja Vuetify koos luua kaasaegseid ja kasutajasõbralikke esitluskihte ning pakuvad suurepärast kasutajakogemust.

3.4.3 Arendusjärjekord

Arendusjärjekord oli põhjalik ja hoolikalt planeeritud, alustades lihtsama funktsionaalsusega ja liikudes järk-järgult keerukamate lahenduste poole.

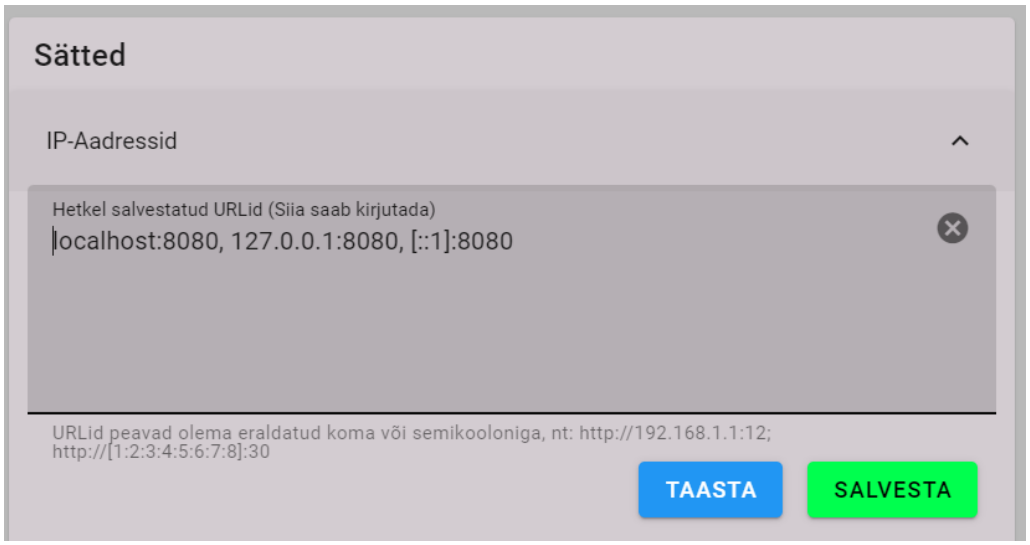
Esmalt loodi meetmete kuvamise vaade (vt Joonis 1), kus kasutati staatilist IP-aadressi. See võimaldas rakenduse esialgset testimist ja veendumist, et meetmete kuvamine toimib ootuspäraselt.



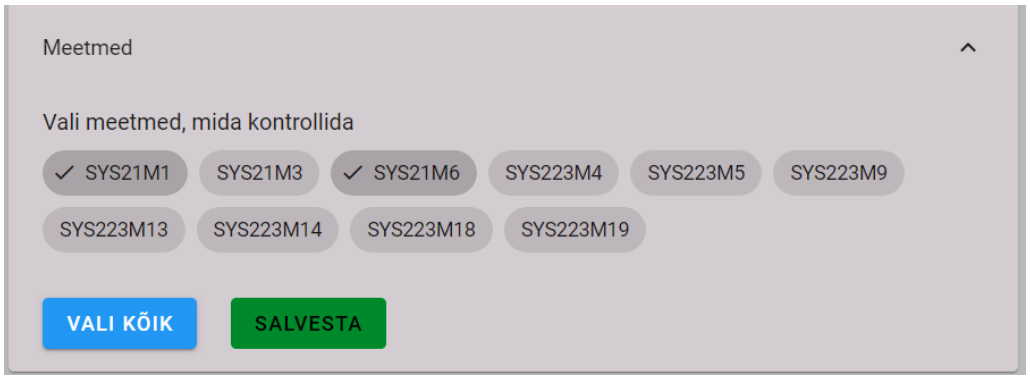
Joonis 1. Algne kahe meetme vaade ning neist üks on avatud täpsema info saamiseks

Järgmisena lisati võimalus kasutajatel sisestada konkreetseid IP-aadresse (vt Joonis 2) ja valida meetmeid (vt Joonis 3), mida kontrollida soovitakse. See laiendas rakenduse funktsionaalsust, võimaldades kasutajatel interaktiivselt määratleda, milliste SYS meetmete kontrolli soovitakse rakendada konkreetsetele IP-aadressidele. Komponentide algset paigutust, mis oli seadistatud autori nägemuse järgi, muudeti arenduse käigus pärast seda, kui mõned potentsiaalsed kasutajad testisid rakendust ja andsid tagasisidet. Peale IP-aadresside sisestamist ja koma või semikooloniga eraldamist tuleks vajutada nupule `Salvesta`. Sellele nupule vajutades teeb esitluskiht enne IP-aadresside salvestamist kolm tegevust. Esiteks eemaldab see kõik duplikaat IP-aadressid, mis näevad teksti kujul identsed välja. Teiseks lisab IP-aadresside algusesse `http://`, kui seda seal juba pole, ning kolmandaks testib iga IP-aadressi päringuga URI-le `/api/`. Kõik IP-aadressid, mis ei vasta ootuspäraselt sellele päringule, eemaldatakse nimekirjast. Selline funktsionaalsus on mõeldud trükivigade ja liigse internetiliikluse vältimiseks.

Samuti on nupp `Taasta`, mis taastab teksti seisundi IP-aadresside lahtris, mis oli enne nuppu `Salvesta` vajutamist. Antud nupp tuleb kasuks, kui mõni IP-aadress kadus kirjavea tõttu ning tekib soov seda parandada.

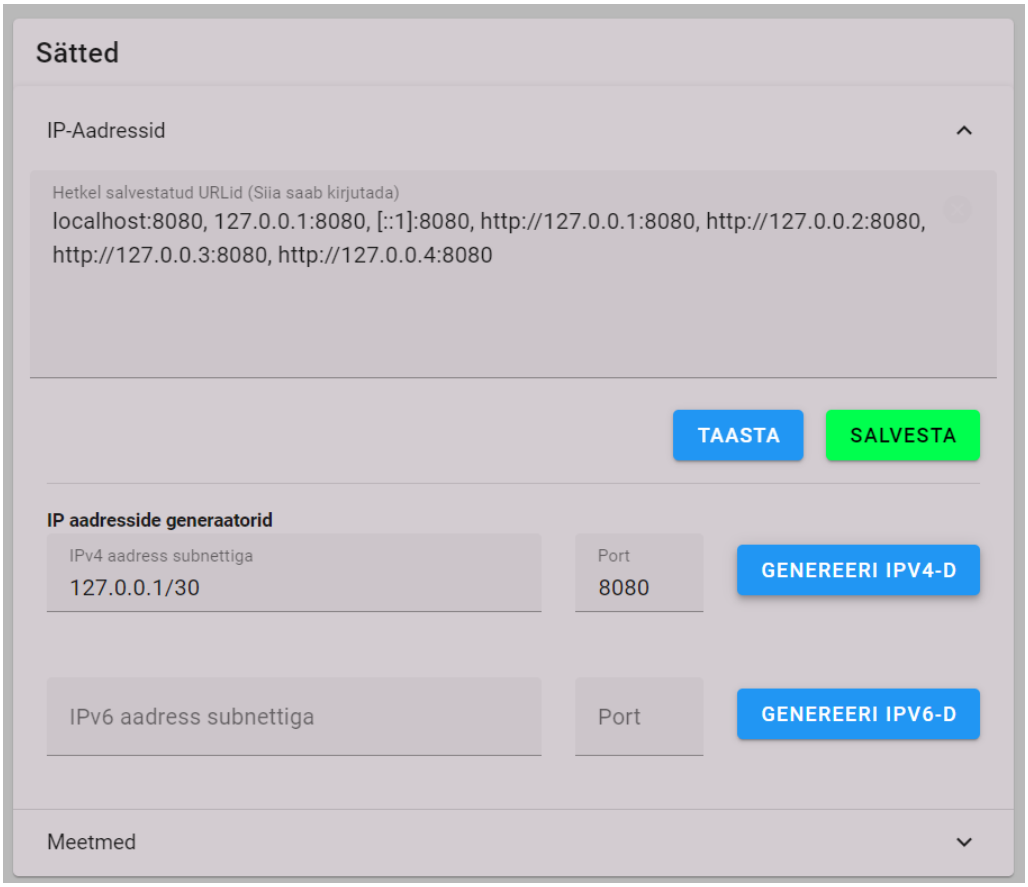


Joonis 2. IP-aadresside sisestuseks mõeldud lahter



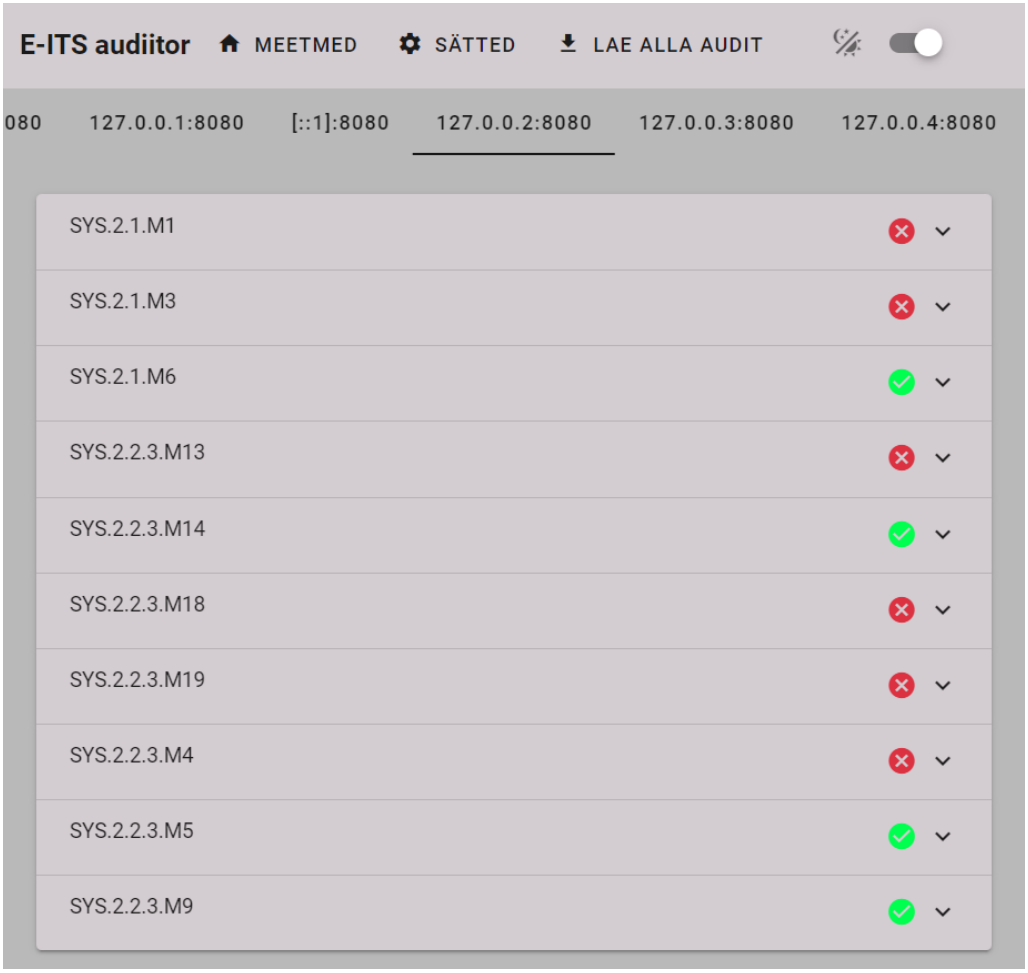
Joonis 3. Meetmete valimise jaoks loodud nupud, kus hetkel on valitud SYS21M1 ja SYS21M6

Edasi implementeeriti IPv4- ja IPv6-aadresside generaatorid (vt Joonis 4), mis võimaldavad aadresse genereerida alamvõrgumaski alusel. Otsus nende generaatorite lisamiseks tuli rakenduse testimisest, kus rakendust katsetasid potentsiaalsed kasutajad ning andsid tagasisidet. See funktsionaalsus on mõeldud suurema hulga IP-aadresside genereerimiseks ja kontrollimiseks ilma, et neid peaks käsitsi sisestama.



Joonis 4. IPv4- ja IPv6-aadresside generaatorid, kus on just genereeritud IPv4-aadresse olemasolevate aadresside otsa

Salvestatud IP-aadresside tulemuste nägemiseks ja nende vaadete eraldamiseks loodi brauserilaadne akende lahendus, kus iga aken kuvas ühe IP-aadressi tulemusi (vt Joonis 5). Lisati ka nupp `Lae alla audit`, mis võimaldab talletada kontrollitud arvutite tolle hetke vastavust meetmetele tekstifaili kujul.

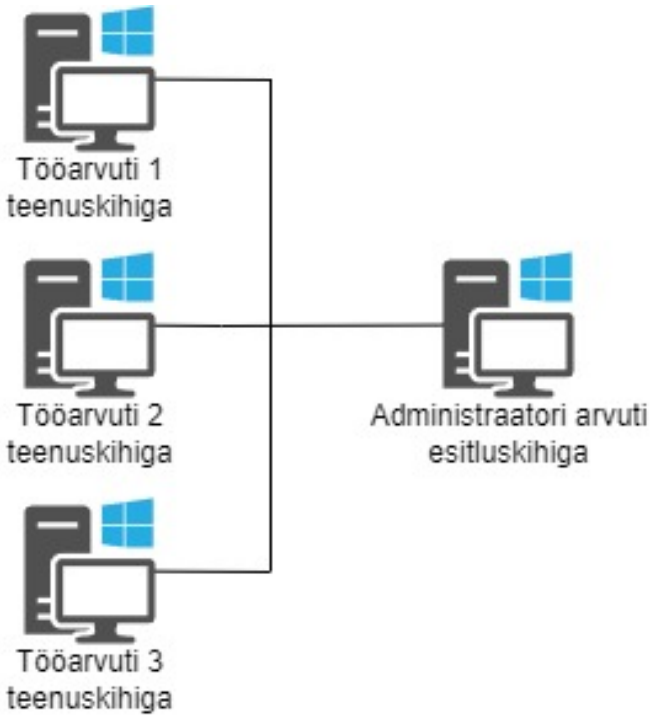


Joonis 5. IP-aadressi 127.0.0.2 meetmetele vastavuse vaade

Kogu arenduse vältel testiti põhjalikult iga uut funktsionaalsust. Selline lähenemine võimaldas rakenduse töökindluse ja arengu vastavalt kasutajate vajadustele ja tagasisidele. Rakenduse arhitektuur on kujutatud peaktükis Lisa 2.

3.5 Näide rakenduse teenuskihi ja esitluskihi kasutamisest

Kui arvutivõrgu administraator on teenuskihi edukalt tööarvutitel konfigureerinud, kasutades selleks teenuskihi lähtekoodi juures olevaid juhiseid, ning taganud vajalikud õigused, et võimaldada administraatori arvuti ning kontrollitavate arvutite vaheline võrguliiklus, on tal võimalus esitluskihti edukalt kasutada. Olles paigaldanud esitluskihi lähtekoodi juures olevate juhendite järgi, saab administraator esitluskihi kaudu sisestada tööarvutite IP-aadressid koos vastavate portidega ning pärida auditi raportit nende arvutite meetmete täitmise kohta. Lihtne versioon arvutivõrgust on välja toodud Joonisel 6. [18, 19]



Joonis 6. Näide arvutivõrgust, kus saaks rakendust kasutada

4. Tulevikusuunad ja soovitused

Tulevikus võiks kaaluda taustrakenduste arendamist mitte ainult Windows 11 keskkonnale, vaid ka teistele populaarsetele platvormidele nagu macOS, Linux ja Unix. See laiendaks rakenduse kättesaadavust ja kasutatavust erinevate klientide ja serverite jaoks ning suurendaks üldist turvalisust. Edasiarenduses on samuti võimalik Osqueryt kasutada, sest see toetab ka macOS-i ja Linuxit.

macOS on populaarne operatsioonisüsteem, mida kasutatakse laialdaselt nii isiklikuskui ka töökeskkonnas. Seetõttu oleks oluline arendada taustrakendusi, mis ühilduvad macOS-iga ja pakuvad sarnast turvalisust nagu Windows 11 platvormil. See võib hõlmata spetsiaalset tarkvaraarenduskomplekti (SDK) loomist macOS-i jaoks.

Linux ja Unix on serverikeskkonnas levinud operatsioonisüsteemid. Taustrakenduste arendamine nendele platvormidele võimaldaks suurendada süsteemide turvalisust ja tagada ühilduvus erinevate klientide ja serveritega. See võib hõlmata spetsiaalsete tööriistade ja API-de loomist Linuxi ja Unixi keskkondade jaoks ning sügavat integreerimist süsteemi turvameetmetega. Selle edasiarenduse juures aga osutuks suureks probleemiks mitmete erinevate Linuxi distributsioonide kontrollimine, mis viiks selle töö mahtu suuremaks kui bakalaureuse töös.

Samuti on oluline kaaluda rakenduste arendamist mobiilplatvormidele, nagu iOS ja Android, kuna tahvelarvutite ja nutitelefonide kasutamine on oluline osa perearsti töökeskkonnast. iOS ja Android on laialdaselt kasutatavad mobiilsed operatsioonisüsteemid, mis pakuvad erinevaid turvalahendusi ja funktsioone. Rakenduste arendamine nendele platvormidele tagaks, et perearstid saavad kasutada mobiilseid seadmeid turvaliselt ja tõhusalt.

Perearstikeskustes on tõenäosus kohata seadmeid, millel on integreeritud ja uuendamata operatsioonisüsteeme, mille E-ITSi nõuetele vastavust võib olla võimatu saavutada või võimatu saavutada nii, et säiliks seadme põhifunktsionaalsus. See suurendab turvariske, mistõttu on oluline, et selliseid seadmeid kas ei liideta võrku või paigutatakse need spetsiaalselt seda seadet kaitsma konfigureeritud tulemüüri taha.

RIA vaatab E-ITSi meetmed igal aastal üle, mille tagajärjel võivad mõned juba implementeeritud meetmed muutuda või täiesti uued lisanduda. Seega on oluline hoida ka Windowsi toetavat teenuskihti ajakohasena.

5. Kokkuvõte

Antud bakalaureusetöö eesmärgiks oli aidata kaasa perearstikeskuste küberturvalisuse parandamisele, pakkudes praktilist lahendust E-ITSi standardi jälgimiseks ning süsteemi turvalisuse tagamiseks. Selle eesmärgi täitmiseks tutvustati E-ITSi põhimõtteid ja süsteemi, kaardistati SYS meetmed ning arendati rakenduse teenuskiht ja esitluskiht vastavalt valitud raamistikele.

Töö käigus saavutati mitmeid olulisi tulemusi. Esiteks süvendati arusaamist E-ITSi põhimõtetest ja meetmetest ning nende rakendamise olulisusest organisatsiooni infoturbe tagamisel. Teiseks kaardistati ja analüüsiti SYS meetmed, tuues esile nende kontrollitavuse ja rakendatavuse perearstikeskuse kontekstis.

Rakenduse arendusprotsessis valiti teenuskihi jaoks Java Spring Boot raamistik ning esitluskihi jaoks Vue.js koos Vuetify kasutajaliidese teegiga. Arendusprotsessi käigus demonstreeriti tehnilisi oskusi ning võimet rakendada valitud tehnoloogiaid vastavalt töö nõuetele.

Kuigi töö saavutas suures osas seatud eesmärgi, tuli ette ka puuduseid ja väljakutseid. Üheks oluliseks väljakutseks oli osalise kontrollitavusega meetmete täieliku mõistmise ja rakendamise keerukus. Lisaks võisid rakendusprotsessis ilmnenud tehnilised raskused mõjutada rakenduse lõpptulemust.

Kasutatud kirjandus

[1] Riigi Infosüsteemide Amet. *E-ITS profiili näidis perearstidele v.2023*$_1$. [Vaadatud: 10-04-2024]. URL: https://eits.ria.ee/api/2/main_menu/asset/E-ITS%20profiil%20perearstidele%20v.2023_1.pdf.

[2] *Process monitori koduleht*. [Vaadatud: 29-04-2024]. URL: https://learn.microsoft.com/en-us/sysinternals/downloads/procmon.

[3] Kinza Yasar. *Mis on Registry Editor?* [Vaadatud: 29-04-2024]. URL: https://www.techtarget.com/searchenterprisedesktop/definition/Windows-Registry-Editor.

[4] *Java 17*. [Vaadatud: 07-05-2024]. URL: https://www.oracle.com/java/technologies/downloads/#java17.

[5] *Spring Booti koduleht*. [Vaadatud: 29-04-2024]. URL: https://spring.io/projects/spring-boot.

[6] *Typescript*. [Vaadatud: 07-05-2024]. URL: https://www.typescriptlang.org.

[7] *Vue.js koduleht*. [Vaadatud: 29-04-2024]. URL: https://vuejs.org/.

[8] *Vuetify koduleht*. [Vaadatud: 29-04-2024]. URL: https://vuetifyjs.com.

[9] Mai-Brit Jürman. *RIA: perearstikeskused on küberturbe nõuete täitmisega hädas*. [Vaadatud: 10-04-2024]. URL: https://www.postimees.ee/7961144/ria-perearstikeskused-on-kuberturbe-nouete-taitmisega-hadas.

[10] Thomas Lepik. "Eesti Infoturbestandardi turvameetmete rakendatuse automaatkontrolli põhimõtted". MA thesis. Ehitajate tee 5,19086, Tallinn, Eesti: Taltech, 2023.

[11] *BSI-IT-Grundschutz koduleht*. [Vaadatud: 30-04-2024]. URL: https://www.bsi.bund.de/EN/Themen/Unternehmen-und-Organisationen/Standards-und-Zertifizierung/IT-Grundschutz/it-grundschutz_node.html.

[12] Riigi Infosüsteemide Amet. *E-ITS küberturvalisuse nõuanded*. [Vaadatud: 16-04-2024]. URL: https://eits.ria.ee/et/versioon/2023/eits-poohidokumendid/eits-nooudedinfoturbehalduse-suesteemile.

[13] Riigi Infosüsteemide Amet. *E-ITSi tutvustus*. [Vaadatud: 10-04-2024]. URL: https://eits.ria.ee/et/avalehe-menueue/tutvustus.

[14] *ISO/IEC-27001 koduleht*. [Vaadatud: 30-04-2024]. URL: https://www.evs.ee/en/iso-iec-27001-2022.

[15] Riigi Infosüsteemide Amet. *E-ITS juhendid ja profiilid*. [Vaadatud: 10-04-2024]. URL: https://eits.ria.ee/et/avalehe-menueue/juhendid.

[16] *Flask koduleht*. [Vaadatud: 30-04-2024]. URL: https://flask.palletsprojects.com/en/3.0.x/.

[17] *OSQuery koduleht*. [Vaadatud: 29-04-2024]. URL: https://osquery.io/.

[18] *Esitluskihi lähtekood*. [Vaadatud: 07-05-2024]. URL: https://github.com/Antspihl/EITS_auditor_front.

[19] *Teenuskihi lähtekood*. [Vaadatud: 07-05-2024]. URL: https://github.com/Antspihl/EITS_auditor_back.

Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks[1]

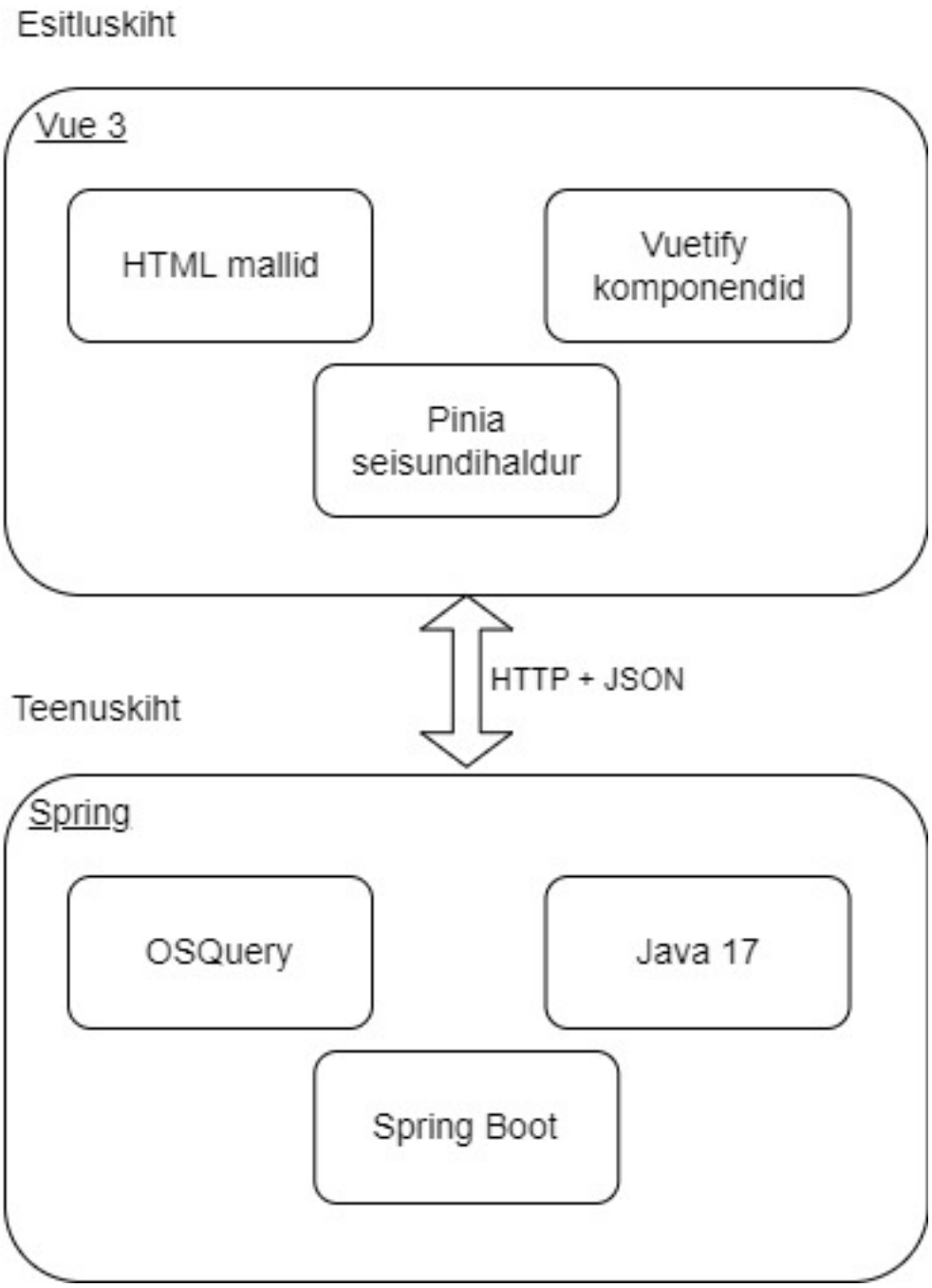
Mina, Indrek Pihl

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose "E-ITSi SYS meetmete rakendamise automaatseire perearstikeskuse keskkonnas", mille juhendaja on Toomas Lepik
   1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
   1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

27.05.2024

[1] Lihtlitsents ei kehti juurdepääsupiirangu kehtivuse ajal vastavalt üliõpilase taotlusele lõputööle juurdepääsupiirangu kehtestamiseks, mis on allkirjastatud teaduskonna dekaani poolt, välja arvatud ülikooli õigus lõputööd reprodutseerida üksnes säilitamise eesmärgil. Kui lõputöö on loonud kaks või enam isikut oma ühise loomingulise tegevusega ning lõputöö kaas- või ühisautor(id) ei ole andnud lõputööd kaitsvale üliõpilasele kindlaksmääratud tähtajaks nõusolekut lõputöö reprodutseerimiseks ja avalikustamiseks vastavalt lihtlitsentsi punktidele 1.1. ja 1.2, siis lihtlitsents nimetatud tähtaja jooksul ei kehti.

Lisa 2 - Rakenduse arhitektuuri skeem



Joonis 7. Rakenduse arhitektuuri skeem

Lisa 3 - Uue meetme lisamine rakendusse

Koodi lingid:

- Esitluskiht: github.com/Antspihl/EITS_auditor_front
- Teenuskiht: github.com/Antspihl/EITS_auditor_back

Uue meetme lisamise järjekord:

1. Täpsustada, mida tuleb kontrollida
2. Implementeerida see teenuskihis
3. Lisada vastav info esitluskihti

0.1 Täpsustamine, mida tuleb kontrollida

Teen näitena läbi väljamõeldud meetme. Väljamõeldud meede on järgnev: "Arvutil olev kahjurvaratõrje tarkvara on uuendatud".

Seda nõuet saab kergelt kontrollida Osqueryga. Tehes päringu:

```
SELECT type, name, state, signatures_up_to_date
AS up_to_date
FROM windows_security_products;
```

, mis annab tulemuseks:

```
[
{"name":"Windows Firewall","state":"On",
"type":"Firewall","up_to_date":"1"},
{"name":"Microsoft Defender Antivirus","state":"On",
"type":"Antivirus","up_to_date":"1"}
]
```

Sellest saab välja lugeda, et kõik on uuendatud, sest muutuja `up_to_date` on mõlemal võrdne 1ga.

0.2 Teenuskihti implementeerimine

Faili `src/main/java/ee/taltech/EITS_auditor_back/controller/CheckController.java` klass `Checkcontroller` tuleb lisada otspunkt meetodi kujul. Eeldusel, et väljamõeldud meetme kood on SYS.2.2.9.M99, siis kood oleks järgnev:

```
@GetMapping("/SYS229M99")
    public Sys229M99DTO checkIFAntiMalwareIsUpdated() {
        return checkService.getAntiMalwareStatus();
    }
```

Samuti tuleb teha andmesaateobjekt kausta `src/main/java/ee/taltech/EITS_auditor_back/dto/response` nimega `Sys229M99DTO`, mida läheb vaja esitluskihile info saatmiseks. Tavaliselt tuleks teha ka iga Osquery tabeli vastu tehtud päringu vastuse tõlgendamiseks Java keelde andmesaateobjekt, mis vastab päringus kirjutatud tulpade nimedele. Kuna rakenduses on juba tehtud päringuid tabeli `windows_security_products` vastu, siis on vastav andmesaateobjekt `SecurityDTO` kausta `src/main/java/ee/taltech/EITS_auditor_back/dto/osquery` juba loodud. `Sys229M99DTO` ja `SecurityDTO` näeksid välja sellised.

```
package ee.taltech.EITS_auditor_back.dto.response;

public record Sys229M99DTO(
        boolean antiMalwareIsUpdated
) {
}
-----------------------------------------------
package ee.taltech.EITS_auditor_back.dto.osquery;

public record SecurityDTO(
        String name,
        String state,
        String type,
        String up_to_date
) {
}
```

Sisukamate meetmete puhul võib lisada loodavasse andmesaateobjekti rohkem kui ühe välja, aga need peavad kõik olema Boole'i muutuja väljad.

Järgmisena tuleb lisada loogika, mis kontrollib vastava päringu tulemust faili `src/main/java/ee/taltech/EITS_auditor_back/service/CheckService.java` klassi `CheckService`:

```
public Sys229M99DTO getAntiMalwareStatus() {
AtomicBoolean antiMalwareIsUpdated = new AtomicBoolean(false);

String response = OSQuery.executeOSQueryCommand(
"SELECT type, name, state, signatures_up_to_date AS up_to_date
FROM windows_security_products"
);

List<SecurityDTO> securityProducts = objectMapper.readValue(
response, new TypeReference<>() {}
);

securityProducts.forEach(securityDTO -> {
    if (securityDTO.name().toLowerCase().contains("antivirus")
        && (securityDTO.state().equalsIgnoreCase("on"))
        && securityDTO.up_to_date().equalsIgnoreCase("1")) {
        antiMalwareIsUpdated.set(true);
    }
});

return new Sys229M99DTO(antiMalwareIsUpdated.get());
}
```

0.3 Vastava info lisamine esitluskihti

Esmalt tuleb lisada meetme kood failis `src/api/MainStore.ts` massiivi nimega `allMeasures` ehk see massiv näeks välja selline pärast lisamist.

```
allMeasures: ["SYS21M1", "SYS21M3", "SYS21M6",
  "SYS223M4", "SYS223M5", "SYS223M9", "SYS223M13",
  "SYS223M14", "SYS223M18", "SYS223M19", "SYS229M99"
] as string[],
```

Teiseks tuleks lisada teenuskihi poolt päritava andmesaateobjekti nii öelda tõlkimiseks paar rida faili `src/molecules/translation.ts`. Selleks, et Boole'i muutuja nime, hetkel siis `antiMalwareIsUpdated`, asemel kuvataks tekst `Arvutil olev`

`kahjurvaratõrje tarkvara on uuendatud` tuleb lisada üks rida loendtüüpi `TranslationEnum`.

```
export enum TranslationEnum {
    // Alumised kaks rida on lisatud read
    antiMalwareIsUpdated = "
    Arvutil olev kahjurvaratorje tarkvara on uuendatud"
}
```

Viimasena tuleb lisada rida samas failis olevasse loendtüüpi `TranslateMeasure` , et kuvataks andmesaateobjekti nime `SYS229M99DTO` asemel korrektne nimi `SYS.2.2.9.M99`.

```
    export enum TranslateMeasure {
    // Alumine rida on lisatud rida
    antiMalwareIsUpdated = "SYS.2.2.9.M99"
    }
```

Lisa 4 - Valitud meetmed

Tabel 2. Meetmed ja nende alampunktid

Meetme kood	**Kirjeldus**
SYS.2.1.M1	Kasutajate turvaline autentimine [kasutaja]
a)	Klientarvutit on võimalik kasutada ainult end nõuetekohaselt autentinud kasutajal.
b)	Mistahes tegevuse puhul klientarvutis on võimalik tuvastada tegevuse sooritaja.
c)	Autentimisandmeid (nt parooli) muuta saab üksnes selleks volitatud kasutaja.
d.2)	Ekraanilukk käivitub kasutaja poolt käsitsi aktiveerituna või automaatselt pärast ettemääratud ajavahemikku.
d.4)	Ekraaniluku avamine on võimalik vaid kasutaja autentimisega.
SYS.2.1.M3	Uuendite automaatpaigaldus
a)	Kui IT-süsteemi kaitsmiseks kahjurvaraga nakatumise eest ei ole kasutusele võetud samaväärseid või rangemaid meetmeid, on Windows klientarvutis aktiveeritud Microsofti kahjurvaratõrje (nt Windows Defender) komponendid.
SYS.2.1.M6	Kahjurvaratõrje tarkvara
a)	Klientarvutid on varustatud aktiveeritud kahjurvaratõrje tarkvaraga.
SYS.2.2.3.M4	Telemeetria andmekaitseseaded
a)	Telemeetriateenuste andmete edastamine operatsioonisüsteemi tootjale on seadistuses piiratud. Windows 10 või Windows 11 Enterprise versiooni kasutamisel on telemeetria tase seadistatud valikväärtusele 0 (Security).
b)	Kui klientarvuti telemeetriaseadistusi ei ole võimalik piirata, on andmete edastamine operatsioonisüsteemi tootjale blokeeritud võrgutaseme meetmetega (nt tulemüürireeglitega).
SYS.2.2.3.M5	Windows klientarvuti kahjurvara tõrje
a)	Kui IT-süsteemi kaitsmiseks kahjurvaraga nakatumise eest ei ole kasutusele võetud samaväärseid või rangemaid meetmeid, on Windows klientarvutis aktiveeritud Microsofti kahjurvaratõrje (nt Windows Defender) komponendid.

SYS.2.2.3.M9	Keskne autentimine
a)	Kekseks autentimiseks kasutatakse Kerberost. Kui seda ei tehta, siis alternatiivina võib kasutada autentimisprotokolli NTLMv2.
SYS.2.2.3.M13	Funktsiooni SmartScreen desaktiveerimine
a)	Microsoft Defenderi funktsioon SmartScreen, mis kontrollib Internetist alla laaditud faile ja veebisisu võimaliku kahjurtarkvara suhtes, kuid võib teatud tingimustel edastada Microsoftile isikuandmeid, on desaktiveeritud.
SYS.2.2.3.M14	Digitaalse assistendi Cortana desaktiveerimine [kasutaja]
a)	Digitaalne assistent Cortana on desaktiveeritud.
SYS.2.2.3.M18	Remote Assistance kaugtoe turvaline rakendamine
a)	Lokaalse tulemüüri konfiguratsioon võimaldab kasutada kaugtoevahendit Remote Assistance.
SYS.2.2.3.M19	Kaughaldusvahendi RDP turvaline rakendamine [kasutaja]
a)	Lokaalse tulemüüri konfiguratsioon võimaldab kasutada RDP-d (ingl Remote Desktop Protocol, RDP).