Niklas Toikka

FYYSISET TUNNISTEET JA ELEMENTIT OSANA SISÄISEN TIETOVERKON MONIVAIHEISTA TUNNISTAUTUMISTA

Opinnäytetyö

Tekniikan ammattikorkeakoulututkinto

Kyberturvallisuuden koulutus

2025

Kaakkois-Suomen ammattikorkeakoulu

Kaakkois-Suomen ammattikorkeakoulu

Tutkintonimike	Insinööri (AMK)
Tekijä/Tekijät	Niklas Toikka
Työn nimi	Fyysiset tunnisteet ja elementit osana sisäisen tietoverkon monivaiheista tunnistautumista
Toimeksiantaja	Kaakkois-Suomen ammattikorkeakoulu Oy, ICTLAB
Vuosi	2025
Sivut	64 sivua, liitteitä 7 sivua
Työn ohjaaja(t)	Tomi Pahula

TIIVISTELMÄ

Monivaiheinen tunnistautuminen on tärkeä osa digitaalista tietoturvallisuutta. Monivaiheisen tunnistautumisen avulla digitaalisia identiteettejä kyetään suojaamaan varmentamalla identiteetin aitous kahdella tai useammalla autentikaattorilla, mikä torjuu identiteettien väärinkäyttötapauksia tehokkaasti. Tästä huolimatta monivaiheiset autentikointimenetelmät eivät täysin kykene suojaamaan identiteettiä kalasteluhyökkäyksiltä. Autentikaattorit ovat vahvasti kytköksissä kolmansien osapuolten pilvi-infrastruktuuriin, jolloin tunnistautuminen eri palveluihin on mahdotonta häiriötilanteiden sattuessa.

Tässä opinnäytetyössä tutkittiin, miten identiteetin voisi autentikoida monivaiheisesti hyödyntämällä olemassa olevia fyysisiä tunnisteita ja elementtejä. Fyysisten tunnisteiden ja elementtien avulla identiteetti pyritään varmentamaan implisiittisesti, jolloin inhimilliset virheet saadaan poistettua autentikointimetodista. Ongelmaa on aiemmin tutkittu projektien muodossa, mutta projektien tuloksena syntynyt Kuisti-järjestelmä kykeni hyödyntämään vain kulunvalvontajärjestelmää käyttäjien monivaiheisessa autentikoinnissa. Tämän opinnäytetyön tarkoituksena oli laajentaa tätä konseptia kattamaan myös muut järjestelmät, joita työntekijät päivittäin käyttävät.

Opinnäytetyö toteutettiin kehittämistutkimuksena. Tämä lähestymistapa soveltui hyvin opinnäytetyön suoritukseen, sillä tutkimuksen tarkoituksena oli kehittää aiemmin rakennettua Kuisti-järjestelmää. Kehitettyä Kuisti-järjestelmää testattiin Kaakkois-Suomen ammattikorkeakoulun VirtualLab-alustalla.

Tutkimustulokset osoittivat, että fyysisiä tunnisteita ja elementtejä voidaan hyödyntää monivaiheisessa tunnistautumisessa. Kuisti-järjestelmän avulla testiympäristössä simuloidun kulunvalvonta- ja työajanseurantajärjestelmän dataa kyettiin käyttämään uudessa, implisiittisessä autentikointimetodissa, joka liitettiin osaksi sisäisen tietoverkon autentikointiprosessia. Tutkimustulokset myös osoittivat, että ulkoisten järjestelmien dataa kyetään käyttämään autentikoinnin lisäksi myös verkkotason pääsynhallinnassa.

Tämä opinnäytetyö tuo esille mahdollisuuden käyttää olemassa olevia järjestelmiä paikallisesti suoritettavassa, monivaiheisessa autentikointiprosessissa. Opinnäytetyön lopputuloksena kehitetty järjestelmä ei kuitenkaan täysin kykene estämään inhimillisistä virheistä johtuvia hyökkäyksiä, minkä takia jatkotutkimus ja -kehitys on tarpeellista.

Asiasanat: todentaminen, pääsynvalvonta, kyberturvallisuus, tietoturvallisuus

South-Eastern Finland University of Applied Sciences

Degree title	Bachelor of Engineering
Author	Niklas Toikka
Thesis title	Physical identifiers and elements as part of multi-factor authentication in an internal information network
Commissioned by	South-Eastern Finland University of Applied Sciences, ICTLAB
Time	2025
Pages	64 pages, 7 pages of appendices
Supervisor	Tomi Pahula

ABSTRACT

Multi-factor authentication is an important part of digital information security. With multi-factor authentication, digital identities can be protected by verifying the identity with two or more authenticators, which effectively prevents misuse. However, multi-factor authentication methods cannot completely protect identities from phishing attacks. Authenticators are strongly connected to third-party cloud infrastructure, making it impossible to authenticate to different services if the infrastructure is unavailable.

The objective of this thesis was to research how identities could be authenticated in multifactorial means by utilizing existing physical identifiers and elements that employees use daily in their work environment. With physical identifiers and elements, identities can be authenticated implicitly, thereby removing human errors from the authentication method. The problem has previously been studied in previous projects, but the resulting Kuisti system made during the projects was only able to utilize data from access control systems. The purpose of this thesis was to extend this concept to also cover other systems that employees use daily.

This thesis was carried out as design-based research. This approach was well suited for this thesis, as the purpose of this thesis was to improve the previously built Kuisti system. The improved Kuisti system was tested in a virtual environment on the VirtualLab platform.

The research results showed that physical identifiers and elements can be utilized in multi-factor authentication. With the improved Kuisti system, it was proved that data from simulated access control and work time monitoring systems in the test environment can be used in a new, implicit authentication method, which can be integrated into the authentication process of an internal network. The research results also showed that data from external systems can also be used in network-level access control management.

This thesis presents the possibility of using existing systems in a locally hosted, multi-factor authentication process. However, the improved Kuisti system is not fully capable of preventing attacks which stem from human error, which is why further research and development is necessary.

Keywords: access control, authentication, cyber security, information security

SISÄLLYS

1 JOHDANTO

Käyttäjätunnusten monivaiheiset autentikointimekanismit ovat osoittautuneet salasanapohjaisen autentikoinnin tärkeäksi turvaelementiksi. Jatkuvasti kasvavien kyberhyökkäysten määrästä huolimatta, monivaiheinen tunnistautuminen (MFA) on onnistunut vähentämään käyttäjätunnusten väärinkäyttötapauksia, minkä takia monivaiheista tunnistautumista, tai vähintään kaksivaiheista tunnistautumista (2FA), suositellaan käytettäväksi kaikissa verkkopalveluissa. Microsoftin teettämän tutkimuksen mukaan noin 99 % käyttäjätilien kaappausyrityksistä on estettävissä MFA:n avulla (Meyer ym. 2023, 4).

Vaikka monivaiheista autentikointia pidetäänkin tärkeänä turvaelementtinä, ei se silti kykene estämään kaikkia käyttäjätunnuksiin liittyviä hyökkäyksiä. Varsinkin yksinkertaisimmat MFA-metodit, kuten push-ilmoitukset ja OTP-koodit (One-Time Password), ovat alttiita käyttäjän tekemille virheille ja tietojenkalasteluhyökkäyksille. Perinteisiin MFA-metodeihin liittyvien tietoturvariskien vähentämiseksi organisaatioita on pyydetty käyttämään uudenlaisia, tietojenkalasteluhyökkäysresistenttejä (engl. phishing-resistant) MFA-metodeja (CISA 2022b). Uudet suositellut MFA-metodit poistavat ihmisen toiminnan autentikointiprosessista, jolloin kalasteluhyökkäysten tuomat riskit saadaan poistettua.

Uudenlaisia ja perinteisiä MFA-metodeja tarjoavat ratkaisut ovat kuitenkin riippuvaisia kolmansien osapuolten pilvipalveluista. MFA-ratkaisujen kytkeytyneisyys kolmansien osapuolten pilvi-infrastruktuuriin altistaa MFA-ratkaisut palvelukatkoksille, jos MFA-palveluntarjoaja tai tietoliikenneyhteys MFA-palveluun on tahattoman tai tahallisen häiriön kohteena. Esimerkiksi Microsoftin ja Duon MFA-infrastruktuurien häiriöt ovat aiemmin aiheuttaneet palvelukatkoksia MFA-palveluihin (Gatlan 2025; Cisco 2023).

MFA-metodien tulisi siis autentikoida käyttäjä monivaiheisesti ja paikallisesti ilman käyttäjästä riippuvaisia toimenpiteitä. Tämän opinnäytteen tarkoituksena on selvittää, voisiko olemassa olevia tai helposti käyttöönotettavia, fyysisiä elementtejä ja tunnisteita hyödyntää sisäisen tietoverkon autentikoinnin monivaiheistamisessa käyttäjille huomaamattomalla tavalla. Tässä kontekstissa

fyysisillä elementeillä tarkoitetaan yrityksen tiloissa olevia, työntekijöiden päivittäin käyttämiä fyysiseen tunnisteeseen tai elementtiin liittyviä järjestelmiä ja laitteita, joihin lukeutuvat muun muassa fyysisillä tunnisteilla toimivat kulunvalvontajärjestelmät, työajan seurannan leimasinlaitteet ja henkilökohtaiset mobiililaitteet. Monivaiheinen tunnistautuminen suoritettaisiin paikallisessa verkossa, ja autentikoinnissa hyödynnettäisiin edellä mainituista järjestelmistä saatua dataa. Opinnäytteen tuloksena kehitettyä järjestelmää kyetään hyödyntämään organisaation sisäisen tietoverkon monivaiheisen tunnistautumisen yksinkertaistamisessa käyttäjän näkökulmasta sekä sisäisen tietoverkon autentikoinnin paikallisessa monivaiheistuksessa.

Opinnäytetyön toimeksiantajana toimii Kaakkois-Suomen ammattikorkeakoulu. Ongelmaa on alustavasti tutkittu toimeksiantajan ehdotuksesta projektitöiden muodossa, joiden tuloksena saatiin rakennettua Kuisti-järjestelmä. Kuisti-järjestelmän tarkoituksena oli integroida kuluvalvontajärjestelmä osaksi sisäisen tietoverkon monivaiheista tunnistautumista, mutta myöhemmin tämä osoittautui ongelmalliseksi, sillä kaikilla organisaatioilla ei välttämättä ole pääsyä käyttämänsä kulunvalvonnan dataan. Tämän tutkimuksen tarkoituksena on kehittää Kuisti-järjestelmää yhteensopivammaksi myös muiden vastaavanlaisten, fyysisiä tunnisteita tai elementtejä käyttävien järjestelmien ja laitteiden kanssa monivaiheisen tunnistautumisen turvallisuuden parantamiseksi. Opinnäytetyön toteutuksessa hyödynnetään toimeksiantajan ylläpitämää Virtual-Lab-alustaa, jossa kehitetyn järjestelmän toimivuutta tullaan testaamaan.

2 TUTKIMUSASETELMA

2.1 Tutkimusongelma, -kysymykset ja tavoitteet

Tutkimusongelmana on organisaatioiden sisäisten tietoverkkojen MFA-metodien riippuvuus pilvipalveluihin ja perinteisiin MFA-metodeihin liittyvät tietoturvariskit. Ongelman keskiössä ovat perinteisiin MFA-metodeihin liittyvät sosiaalisen manipuloinnin hyökkäykset, jotka mahdollistavat MFA-metodien ohitukset mahdollisissa hyökkäystilanteissa. Verizone (2022) on raportoinut inhimillisten virheiden vaikuttaneen 82 prosenttiin kaikista raportointihetkellä tapahtuneista tietomurroista. Taku (2023) ja Spitzner (2022) ehdottavatkin ihmiselle näkymättömien autentikointimenetelmien kehitystä ja käyttöönottoa inhimillisten virheiden poistamiseksi.

MFA-ratkaisujen käyttö on myös hyvin kytkeytynyttä kolmansien osapuolten pilvipalveluihin. MFA-ratkaisujen kasvava pilvikeskeisyys voi aiheuttaa palvelukatkoksia MFA-infrastruktuurin häiriötilanteissa, ja pilvi-infrastruktuuriin nojautuminen tekee monivaiheisen tunnistautumisen mahdottomaksi eristetyissä verkoissa. Esimerkiksi Microsoft (2025) ei enää tue paikallisia MFA-ratkaisuja, ja pyytää käyttäjiään siirtymään pilvipalveluihin nojautuvaan infrastruktuuriin. RSA Security (2024) ylläpitää täysin paikallisesti toimivaa MFA-ratkaisua, mutta se pohjautuu OTP-koodien käyttöön, joka vaatii joko sovelluksen tai erillisen laitteiston hankintaa.

Ongelmien ratkaisuksi kyettäisiin hyödyntämään paikallisesti toimivaa järjestelmää, joka poistaisi inhimilliset tietoturvariskit autentikointiprosessista. Yhtenä ongelmaa korjaavana ratkaisuna voidaan käyttää aiemmin kehitettyä Kuisti-järjestelmää. Ongelmasta voidaan johtaa seuraavat tutkimuskysymykset Kuisti-järjestelmän kehitystä varten:

- Mitä tarkoitetaan monivaiheisella tunnistautumisella?
- Minkälaisia ongelmia perinteisiin monivaiheisen tunnistautumisen menetelmiin liittyy?
- Miten fyysisiä elementtejä ja tunnisteita voitaisiin hyödyntää monivaiheisessa tunnistautumisessa ja sisäisen tietoverkon tietoturvallisuuden parantamisessa?

Tutkimuksen tavoitteena on kehittää Kuisti-järjestelmää korjaamaan autenttisissa ympäristöissä havaitut ongelmat tutkimuskysymysten vastausten perusteella. Kehitetyn Kuisti-järjestelmän dokumentaatiota kyetään hyödyntämään tutkimusongelmien korjaamisessa tutkimuksen lähtötilannetta vastaavissa ympäristöissä, ja ratkaisun toimivuutta havainnoimalla ehdottamaan uusia autentikointimenetelmiä monivaiheisen tunnistautumisen turvallisuuden parantamiseksi. Havainnoista muodostettujen teoriaehdokkaiden vakiintuminen Kanasen (2017, 69) mukaan vaatii kuitenkin laajempaa näyttöä uudenlaisten menetelmien toimivuudesta. Dokumentaation avulla mahdollistetaan kehitetyn järjestelmän käyttöönotto ja näin ollen tulosten yleistettävyys, mutta tulosten yleistäminen ja soveltaminen ovat soveltajan vastuulla. Muutokseen tähtäävän tutkimuksen yleistäminen on hankalaa, sillä soveltajan hallinnoiman ympäristön tulee vastata tutkimuksen lähtötilannetta. (Kananen 2017, 67–68; Lukka 2001.) Kehitystutkimuksen aikana dokumentoidut työvaiheet ja testaustulokset

pyritään tekemään tarpeeksi yksityiskohtaisesti, jotta tutkimuksen aikana kehitettyjen ratkaisujen implementointi ja tulosten siirrettävyys olisi mahdollista.

2.2 Tutkimusote ja -menetelmät

Tutkimus suoritetaan kehittämistutkimuksena, sillä tutkimuksen tarkoituksena on parantaa aiemmin rakennetun järjestelmän toiminnallisuutta. Tutkimusta voitaisiin myös lähestyä konstruktiivisella tutkimusotteella, sillä tutkimuksen tavoitteena on luoda ratkaisu tosielämässä havaittuihin ongelmiin. Tämä tutkimus kuitenkin perustuu jo olemassa olevaan konstruktioon (Kuisti-järjestelmä) ja tutkimuksen tavoitteena on olemassa olevan konstruktion kehittämisen lisäksi myös monivaiheiseen tunnistautumiseen liittyvien menetelmien kehitys, jolloin kehittämistutkimus näyttäytyy sopivimmalta lähestymistavalta. Myös Kananen (2017, 18) ja Pernaa (2013) ovat maininneet kehittämistutkimuksen olevan soveltuva menetelmän tai tuotteen tutkimuspohjaiseen kehitykseen toisin kuin konstruktiivinen tutkimus, joka Kanasen (2017, 14) ja Lukan (2001) mukaan yleensä painottuu suuremman kokonaisuuden eli konstruktion rakentamiseen.

Kehittämistutkimuksen valinta on myös perusteltua asetetun tavoitteen kannalta. Tutkimuksen tavoitteena on kehittää Kuisti-järjestelmää korjaamaan käytännön ongelma ja onnistuneen ratkaisun kautta luoda ja ehdottaa uusia autentikointimenetelmiä monivaiheiseen tunnistautumiseen. Kehittämistutkimus mahdollistaa ongelman syklimäisen ratkaisuprosessin, jolloin muutokseen tähtäävän intervention vaikuttavuus voidaan varmentaa intervention jälkeen havainnoimalla. Vaikutuksellisiksi arvioitujen muutosten jälkeen ratkaisut voidaan palauttaa teoriaan. (Kananen 2017, 34–35.) Tässä tapauksessa uudet teoriamaailmaan palautettavat konstruktiot ovat autentikointimetodit, joiden toimivuutta pyritään testaamaan käytännössä Kuisti-järjestelmää kehittämällä.

2.3 Aineistonkeruumenetelmät ja aineiston analyysi

Tutkimuksen aikana käytettävä primääriaineisto koostuu toteutus- ja testausvaiheessa tehdyistä kenttämuistiinpanoista ja havainnoista. Kananen (2017, 46–47) mainitsee, että havainnointi voi olla joko tutkijan tai automatiikan suorittamaa. Tässä tutkimuksessa havainnointi on automatisoitu skriptien avulla,

ja havainnointi kohdistuu pääosin fyysisten tunnisteiden ja elementtien käytöstä johtuvien tapahtumien tarkkailuun. Myös verkkotason pääsynhallinnan toimivuutta havainnoidaan automaattisesti työasemille luotujen skriptien lokimerkintöjen kautta. Ja koska havainnot keskittyvät vain tiettyihin osa-alueisiin, ovat havainnot strukturoituja (Kananen 2017, 47).

Testien aikana primääriaineistoa rikastetaan kenttämuistiinpanoin. Kenttämuistiinpanot kohdistuvat testien aikana tehtyihin, manuaalisiin toimenpiteisiin (simuloidun käyttäjän tekemät toimet). Näiden kenttämuistiinpanojen avulla automatiikan keräämät havainnot voidaan varmentaa, ja ne voidaan korreloida kenttämuistiinpanoihin kirjattujen toimenpiteiden kanssa yhteiselle aikajanalle, jotta intervention vaikutus voidaan todentaa automaattisten havaintojen ja manuaalisten toimenpiteiden turvin. Tämä on tärkeää interventiotutkimuksen kannalta, sillä tehtyjen muutosten vaikutus tutkittavaan ilmiöön on pystyttävä toteamaan (Kananen 2017, 64).

Sekundääriaineistona käytetään Kuisti-järjestelmästä luotua alustavaa dokumentaatiota ja muistiinpanoja sekä kehitykseen tarvittavia ohjeistuksia. Sekundääriaineisto antaa tilannekuvan järjestelmän nykytilasta, ja teoreettisen viitekehyksen kanssa pohjan ratkaisun suunnittelulle. Aineistojen avulla järjestelmää kyetään kehittämään syklisesti ensin suunnittelemalla kehitettävään kohteeseen tehtävä muutos teoriaan pohjautuen, minkä jälkeen muutosten vaikutuksia havainnoimalla ja seuraamalla muutosten vaikuttavuus pystytään arvioimaan. Sykleissä tapahtuva kehitys mahdollistaa myös suuremman muutoksen pilkkomisen useammaksi sykliksi, mikä on Pernaan (2013) mukaan yksi tapa suorittaa järjestelmäkehitykseen liittyvä tutkimus.

Kerätyn aineiston analyysillä pyritään varmentamaan kaksi asiaa: fyysisten tunnisteiden ja elementtien hyödynnettävyys autentikointiprosessissa sekä verkkotason pääsynhallinnan toimivuus. Fyysisten tunnisteiden ja elementtien hyödynnettävyyttä analysoidaan Kuisti-palvelimen lokimerkintöjen ja kenttämuistiinpanoihin kirjattujen toimenpiteiden avulla. Lokimerkintöjen aikaleimat mahdollistavat toimenpiteiden yhdistämisen palvelimien tekemiin toimenpiteisiin ja havaintoihin, jolloin kehitetyn järjestelmän toimivuus voidaan varmentaa.

Verkkotason pääsynhallinnan toimivuutta analysoidaan testiskenaarion työasemille rakennettujen skriptien tulosteiden perusteella. Tulosteihin sisällytettävien aikaleimojen avulla havainnot kyetään yhdistämään Kuisti-palvelimen kirjaamiin toimenpiteisiin, jolloin Kuistin-järjestelmän vaikutus sekä ennen muutosta että sen jälkeen on suoraan yhdistettävissä työasemien tekemiin havaintoihin.

Tutkimuksen luotettavuutta tarkastellaan päivitetyn järjestelmän dokumentaation ja ratkaisun toiminnan kannalta. Kattava työvaiheiden ja testien dokumentointi mahdollistaa tutkimustulosten hallitun toistamisen, joka takaa tutkimustulosten reliabiliteetin (Kananen 2017, 70). Tarkka dokumentaatio takaa myös työn siirrettävyyden ja vahvistettavuuden vastaavanlaisissa ympäristöissä (Pernaa 2013).

3 TEORIA

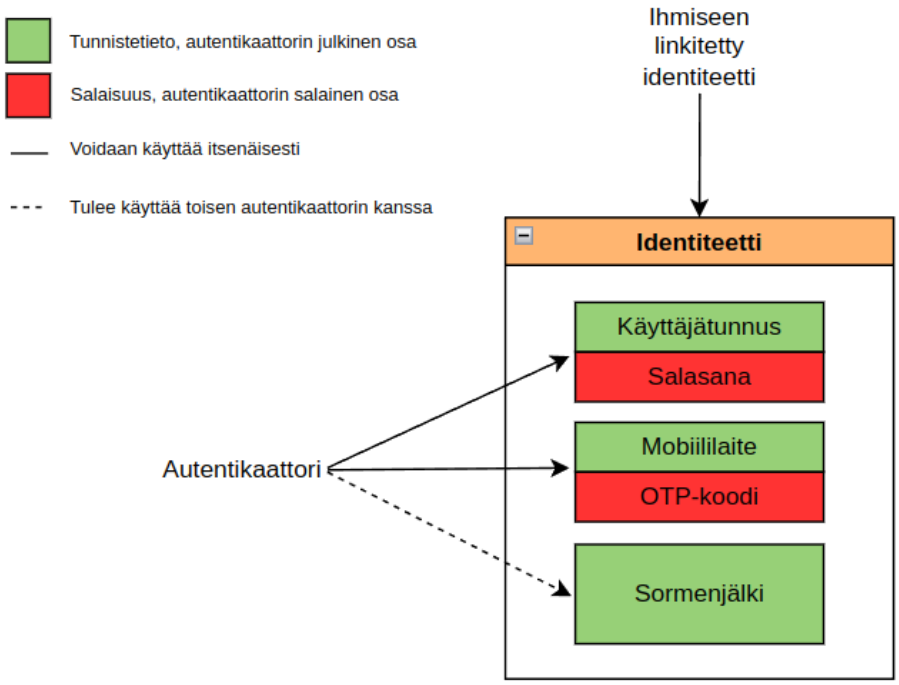
Tässä luvussa käydään läpi mitä monivaiheisella tunnistautumisella käytännössä tarkoitetaan, ja miten identiteetin autentikointi liittyy identiteetin auktorisointiin ja pääsynhallintaan. Tässä luvussa käsitellään myös monivaiheisen tunnistautumisen eri menetelmiä ja menetelmiin liittyviä ongelmia. Teoriaosuus käsittelee myös tapoja, miten perinteisiin MFA-metodeihin liittyviä ongelmia on pyritty ratkaisemaan aiemmin.

3.1 Monivaiheinen tunnistautuminen

Monivaiheisella tunnistautumisella tarkoitetaan identiteetin todentamista kahdella tai useammalla tunnistautumistavalla. Tunnistautumistavat perustuvat identiteetin tietämykseen, omistukseen tai olemukseen. (Grassi, Garcia ym. 2017, 12–14; Traficom s.a.) Brooks (2018, 23) ja OWASP (s.a.) pitävät myös sijantitiedon käyttöä hyväksyttävänä tunnistautumistapana, mutta Grassi, Fenton ym. (2017, 26) ja Awati (s.a.) luokittelevat sijaintitiedon kuuluvan adaptiivisen autentikoinnin piiriin.

Identiteetti on uniikkien todennustekijöiden ryhmä. Todennettava identiteetti voi olla esimerkiksi ihmiseen linkitetty, jolloin identiteetin uniikit todennustekijät, eli autentikaattorit, voivat olla käyttäjätunnuksia, fyysisiä tunnisteita ja hen-

kilötietoja. Identiteetit voivat olla myös laitteisiin ja sovelluksiin linkitettyjä, jolloin autentikaattoreina voidaan käyttää laitteisiin ja sovelluksiin liittyviä uniikkeja arvoja, kuten IP-osoitteita ja ohjelmistoon liittyviä attribuutteja. (Microsoft 2024a; Brooks ym. 2018, 25–26.) Sähköisten identiteettien koostumus on havainnollistettuna kuvassa yksi.

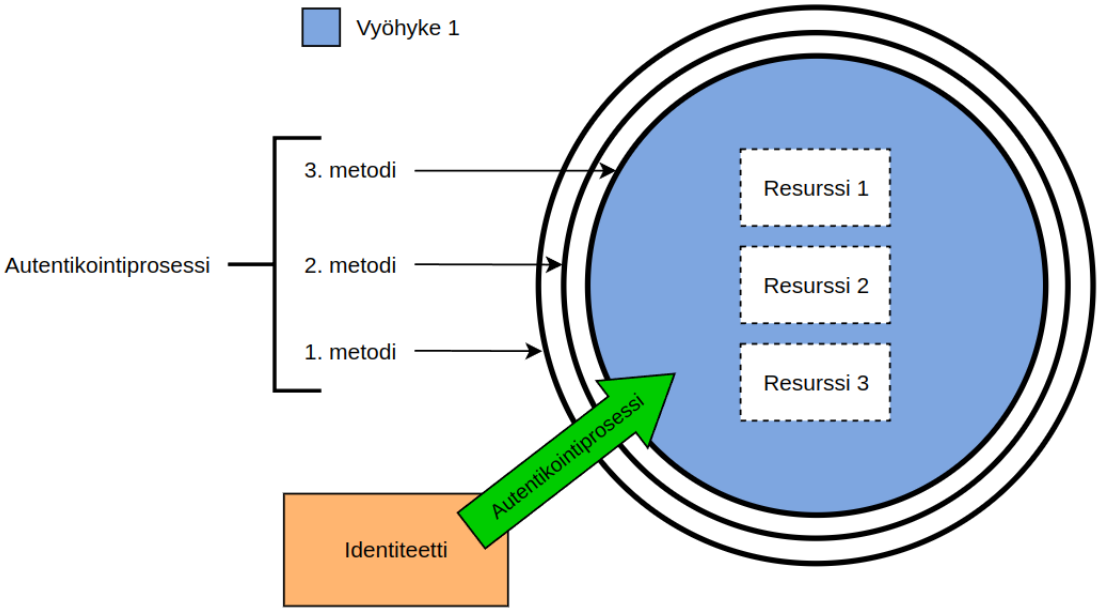


Kuva 1. Identiteetin ja autentikaattoreiden rakenne visualisoituna Microsoftin (2024) ja Brooksin ym. (2018, 25–26) kuvauksiin pohjautuen

Kuvasta yksi voidaan havaita, mistä identiteetti ja autentikaattorit koostuvat. Identiteetti koostuu yhdestä tai useammasta autentikaattorista ja autentikaattorit julkisesta sekä salaisesta osasta. Biometriset tunnisteet, kuten sormenjäljet, eivät kuitenkaan sisällä salaista osaa, minkä takia biometrisiä tunnistetietoja ei voida käyttää autentikoinnissa ilman toista tunnistautumistapaa. Biometrisiä tunnisteita voidaan käyttää autentikaattoreina joko toisten tunnistautumistapojen kanssa tai niillä voi suojata toiseen tunnistautumistapaan kuuluvan autentikaattorin salaisuuden; esimerkiksi mobiililaitteen sisältämä OTP-koodi voidaan suojata sormenjäljellä. (Grassi, Fenton ym. 2017, 26–28; Grassi, Garcia ym. 2017, 12–14.)

Moodley ym. (2014, 464) kuvailevat monivaiheista tunnistautumista monikerroksiseksi autentikoinniksi, sillä autentikointiprosessi voidaan jakaa eri kerroksiin eli metodeihin, missä jokaisen metodin tehtävänä on todentaa identiteetin

esittämän tunnistetiedon paikkansapitävyys. Käytännössä tämä tarkoittaa esimerkiksi identiteetin esittämän käyttäjätunnuksen todentamista salasanan (tietämys) ja biometrisen tunnisteen (olemus) avulla. Yleisesti ottaen todennuksen kokonaisturvallisuus kasvaa metodien määrän kasvaessa, sillä identiteetin on läpäistävä kaikki autentikointikerrokset päästäkseen käsiksi haluamaansa resurssiin (Bayuk 2012, 120). Monivaiheista tunnistautumista kuvaava arkkitehtuuri on esitettynä kuvassa kaksi.



Kuva 2. Monivaiheinen autentikointiprosessi visualisoituna Bayukin (2012, 120) ja Colen ym. (2008, 125) kuvausten perusteella

Kuvan kaksi esimerkissä autentikointiprosessi on jaettu kolmeen eri metodiin, jotka on suoritettava sarjassa, jotta vyöhykkeellä yksi oleviin resursseihin pääsisi käsiksi. Resurssit voivat olla esimerkiksi palvelimia, verkkolaitteita tai tiedostoja, jolloin vyöhyke voisi olla verkkosegmentti. Katkoviivat resurssien ympärillä kuvastavat resurssien omia autentikointiprosesseja, jos sellaisia on käytössä. Kuvan kaksi tapauksessa autentikointiprosessin on todennettava kolme identiteetin esittämää tunnistetietoa, ja jokaisen tunnistetiedon on kuuluttava eri tunnistautumistapaan, jotta autentikointiprosessi voidaan luokitella monivaiheiseksi, kuten Cole ym. (2008, 125) ja OWASP (s.a.) ovat määritelleet.

3.2 Autentikointi, auktorisointi ja pääsynhallinta

Edellä mainittujen autentikaattorien tehtävänä on identiteetin autentikointi. Autentikoinnin jälkeen identiteetillä on implisiittinen pääsy järjestelmään tai ympäristöön, eli identiteetti on auktorisoitu käyttämään ympäristöä tai järjestelmää. Identiteetillä ei kuitenkaan välttämättä ole pääsyä järjestelmän tai ympäristön resursseihin. Tätä ilmiötä on havainnollistettu kuvassa kaksi, missä katkoviivat kuvastavat resurssien omia autentikointimetodeja. Nämä autentikointimetodit hallitsevat pääsyä resursseihin, eli ne ovat resurssien pääsynhallinnan metodeja, jotka erottelevat auktorisoimattomat ja auktorisoidut käyttäjät toisistaan. Autentikoinnin, auktorisoinnin ja pääsynhallinnan termit ovat erittäin tärkeitä ymmärtää autentikointiprosessia käsitellessä, sillä autentikointimetodit ovat suoraan kytköksissä auktorisointiin ja pääsynhallintaan, kuten myös Piscitello (2015) on kertonut. Tässä osiossa perehdytään tarkemmin näiden termien käsittelyyn, jotta niiden eroavaisuudet ja yhteydet toisiinsa tunnistettaisiin.

Autentikoinnin tehtävänä on identiteetin tunnistetietojen todentaminen autentikaattorin avulla. Auktorisoinnilla tarkoitetaan identiteetin valtuuksien tarkastamista, eli onko identiteetillä valtuudet päästä tietyn resurssin luokse tai käsitellä tiettyä resurssia. Valtuudet resurssiin ja resurssin ympäristöön määrittää auktoriteetti, joka on resurssin tai ympäristön omistaja. (Brooks 2018, 13; Cole 2008, 125.)

Pääsynhallinnalla tarkoitetaan metodeja auktorisoitujen ja auktorisoimattomien identiteettien erotteluun. Toisin sanoen pääsynhallinnan tehtävänä on varmistaa, että vain auktorisoidut identiteetit pääsevät resursseihin ja ympäristöihin käsiksi. Pääsynhallinta koostuu tulo-, meno-, ja paluuliikenteen hallinnasta identiteetin valtuuksiin perustuen. Resurssiin tai ympäristöön liittyvän liikenteen rajaus on Brooksin ym. (2018, 12–23) mukaan ensimmäinen ja perusteellisin turvatoimi infrastruktuurin tietoturvallisuuden parantamisessa, sillä infrastruktuuria ei voi vahingoittaa, jos sinne ei ole pääsyä. Pääsyä infrastruktuuriin ja resursseihin turvataan autentikointiprosessein, jotta auktorisoimattomat identiteetit saadaan pidettyä suojattavan infrastruktuurin ulkopuolella. Tällöin identiteetin autentikoinnin voidaan sanoa toimivan auktorisoinnin perustana, kuten Piscitello (2015) ja Brooks ym. (2018, 23) ovat kertoneet.

Kuvassa kaksi esitetty vyöhyke havainnoi autentikoidun identiteetin auktorisoinnin tasoa, joka kertoo identiteetin pääsevän kaikkiin vyöhykkeen rajojen sisäpuolella olevien resurssien luokse, mikä ei kuitenkaan takaa valtuuksia resurssien käsittelyyn. Resurssien käsittelyä estävät resurssien omat autentikointiprosessit, jotka ovat esitettynä kuvassa kaksi katkoviivoin. Tämä autentikointiin perustuva auktorisointi on pääsynhallinnan metodi, jonka tarkoituksena on estää auktorisoimattomien identiteettien pääsy infrastruktuuriin (Piscitello 2015). Kuvasta kaksi myös nähdään, miten identiteetin auktorisointi pohjautuu autentikointiin, kuten Brooks ym. (2018, 23) ja Cole ym. (2008, 119) ovat kertoneet.

3.3 Perinteiset tunnistautumistavat ja niiden ongelmat

3.3.1 Tietämys

Tietämykseen perustuvat autentikaattorit ovat asioita, joita vain autentikoinnin kohteena oleva identiteetti tietää. Yleisimmät tähän luokkaan kuuluvat autentikaattorit ovat käyttäjätunnukset ja PIN-koodit. Tällaisia autentikaattoreita on käytetty perinteisissä verkkopalveluissa niiden helpon implementoitavuuden ja yksinkertaisuuden takia jo pitkään, vaikka ne ovatkin alttiita kalasteluhyökkäyksille. (CISA 2022a; OWASP s.a.) Salasanoja ja niiden tiivistettyjä (engl. hashed) muotoja voidaan kalastelun ohella myös varastaa suoraan tietojärjestelmistä tietomurtojen yhteydessä. Hyökkääjät voivat käyttää varastettuja salasanoja toisiin järjestelmiin murtautumisessa, jos varastettuja salasanoja käytetään myös toisissa tietojärjestelmissä. Google (2019) on raportoinut, että noin 65 % Yhdysvaltain kansalaisista käyttää samoja salasanoja eri järjestelmien välillä, jolloin varastetun salasanan toimivuus toisessa järjestelmässä on todennäköistä. Tämän vuoksi tietämykseen perustuvat autentikaattorit suositellaan suojaamaan hyvin väärinkäyttötapausten estämiseksi (Password Storage Cheat Sheet s.a.).

Turvakysymykset (engl. security questions) kuuluvat myös tähän autentikaattoriluokkaan, mutta Grassi, Fenton ym. (2017, 14) eivät pidä turvakysymyksiä hyväksyttävänä tapana identiteetin autentikointiin, sillä ne ovat salasanojen ja PIN-koodien kanssa alttiita kalasteluhyökkäyksille, kuten myös OWASP (s.a.)

on omissa ohjeistuksissaan maininnut. Turvakysymykset voivat olla jopa salasanoja turvattomampia autentikaattoreita, sillä turvallisten ja muistettavien turvakysymys-vastaus-parien muodostus on osoittautunut erittäin haasteelliseksi. (Bonneau ym. 2015, 141).

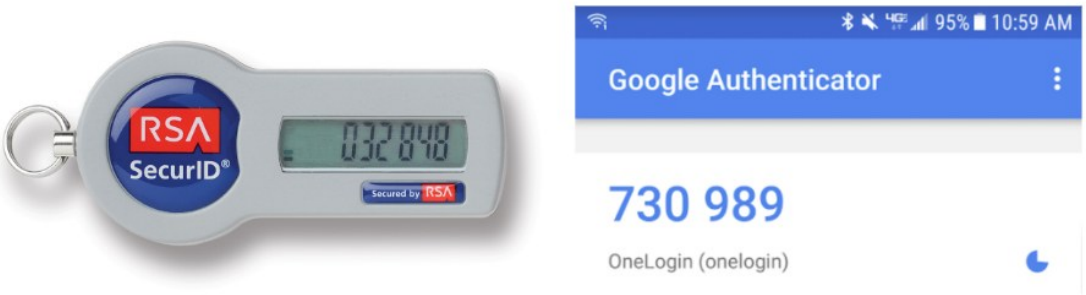
3.3.2 Omistus

Omistukseen pohjautuvilla autentikaattoreilla tarkoitetaan vain autentikoitavan identiteetin omistuksessa olevia asioita, kuten mobiililaitteita, varmennekortteja ja OTP-koodeja. Yleisin omistukseen pohjautuva autentikaattori on tietyn ajan voimassa oleva TOTP-koodi (Time-Based One Time Password), joka generoituu joko kerran autentikointiprosessin aikana, tai vaihtuu tietyin väliajoin identiteetin omistamassa laitteessa tai sovelluksessa. Laitteen tai sovelluksen generoimat TOTP-koodit vaativat kuitenkin koodien syöttämisen autentikointijärjestelmään, minkä takia koodit ovat alttiita sosiaalisen manipuloinnin hyökkäyksille, kuten kalastelulle. (OWASP s.a.; Spitzner 2022.)

Sosiaalisen manipuloinnin hyökkäykset eivät kuitenkaan rajoitu vain käyttäjään. Hyökkääjä kykenee saamaan tekstiviestitse tai puhelimitse lähetettävät OTP-koodit haltuunsa SIM-korttihuijauksella (engl. SIM swapping attack). SIM-korttihuijauksessa hyökkääjä pyytää operaattoria siirtämään uhrin matkapuhelinliittymän uudelle, hyökkääjän hallitsemalle SIM-kortille. Siirtoa varten hyökkääjä kerää kohteestaan henkilökohtaisia tietoja joko sosiaalisen median tai tietovuotojen kautta, ja kerättyjen tietojen avulla hyökkääjä kykenee tunnistautumaan operaattorille liittymän siirtoa varten. Uudelle SIM-kortille siirretty liittymä sisältää uhrin puhelinnumeron, jonka avulla hyökkääjä saa pääsyn uhrille lähetettäviin varmennekoodeihin. (F5 2023; Traficom 2023.)

F5 (2023) mainitsee laitteessa tai sovelluksessa olevien OTP-koodien olevan hyökkääjän saatavilla myös haittaohjelman tai toimitusketjuhyökkäyksen kautta. Haittaohjelman avulla hyökkääjä kykenee lukemaan OTP-koodit suoraan sovelluksesta tai selaimen ja autentikoivan palvelun väliin muodostetun väliintulohyökkäyksen kautta. Väliintulohyökkäys on mahdollista myös toteuttaa huijaamalla käyttäjä oikeaa palvelua imitoivalle huijaussivustolle. Ohjelmis-

topohjaisten OTP-koodien lisäksi myös puhtaasti laitteistopohjaiset OTP-generaattorit ovat haavoittuvaisia. Kuvassa kolme on esitettynä, miltä laitteistopohjainen ja ohjelmistopohjainen OTP-koodigeneraattori näyttävät.



Kuva 3. Laitteisto- ja ohjelmistopohjainen OTP-koodigeneraattori (Grimes 2019, 18)

Kuten kuvasta kolme voidaan havaita, laitteistopohjainen OTP-koodi on näkyvissä suoraan laitteen näytöltä ilman ylimääräisiä suojauksia, jolloin laitteen varastaneen hyökkääjän ei tarvitse selvittää uhrinsa PIN-koodia tai salasanaa laitteen käyttöä varten. Laitteistopohjaisia OTP-koodeja voidaan myös monistaa, jos hyökkääjät tietävät OTP-koodeja generoivan algoritmin ja saavat haltuunsa laitteeseen yhdistetyn siemenluvun (engl. seed). Grimes (2019, 18) on raportoinut kiinalaisten toimijoiden käyttäneen monistettuja siemenlukuja tunkeutuakseen Lockheed Martinin tietoverkkoihin vuonna 2011. Tällaiset hyökkäykset alleviivaavat F5:n (2023) mainitseman, MFA-ratkaisujen tuoman hyökkäyspinta-alan tarkastelun tarpeen.

3.3.3 Olemus

Olemukseen liittyvät autentikaattorit ovat identiteettien biometrisiä tunnistetietoja. Biometriset tunnistetiedot ovat fysiologisia tai käytöksellisiä piirteitä. Fysiologiset piirteet voiva olla sormenjälkiä ja silmän rakenteellisia ominaisuuksia, kun taas käytökselliset piirteet ovat tunnistettavissa olevia tapoja kuten allekirjoituksen tai näppäinpainallusten dynamiikka ja puhe. (Cole ym. 2008, 129; Veerubhotla 2014, 245.) Brooks ym. (2018, 27–29) ovat kertoneet silmien rakenteellisten ominaisuuksien tunnistamisen olevan luotettavin tapa identiteetin biometriseen autentikointiin ja puheeseen perustuvan tunnistautumisen huonoin, sillä henkilön terveydentila voi vaikuttaa puheen äänenpainoon ja laatuun huomattavasti. Biometristen tunnistetietojen skannaukseen tai mittaukseen liittyvä laitteisto on henkilön olotilan aiheuttamien virheiden ohella

altis myös mittauslaitteiston tekemille virheille. Biometristen tunnisteiden käytössä tulisi ottaa huomioon tunnistautumiseen käytettävän laitteiston hygienia, hinta, kerätyn datan talletus ja identiteettien yksityisyys (Cole ym. 2008, 131; OWASP s.a.).

Grassi, Fenton ym. (2017, 7–26) ohjeistavat käyttämään biometrisiä tunnistetietoja jonkin toisen autentikaattorin kanssa, sillä ne eivät muodosta salaisuuksia; sormenjälkiä voidaan kerätä ja kasvojen rakenne kuvata ilman henkilön suostumusta, jolloin kerättyä dataa kyetään käyttämään haitallisissa tarkoituksissa. Tämän vuoksi biometriset tunnistetiedot eivät yksinään voi toimia autentikaattoreina. Veerubhotla (2014, 245) onkin maininnut biometriikan käytön tähtäävän identiteetin tunnistukseen. Tietämykseen ja omistukseen perustuvat autentikaattorit varmennetaan deterministisesti toisin kuin biometriset tunnistetiedot, joiden varmennus perustuu todennäköisyyteen (Grassi, Fenton ym. 2017, 26). Myös F5 (2023) painottaa biometristen tunnisteiden kopiointiin ja varastamiseen liittyviä riskejä, sillä varastettu laite tai salana on vaihdettavissa huomattavasti helpommin biometrisiin tunnisteisiin verrattuna.

Biometrinen tunnistautuminen tuo mukanaan enemmän implementointitapoja kuin omistukseen pohjautuva tunnistautuminen, minkä vuoksi F5 (2023) kertoo jokaisen mahdollisen haavoittuvuuden mallintamisen haastavaksi. Biometrinen tunnistautumistapa on kuitenkin tietämykseen ja omistukseen pohjautuvien autentikaattoreiden tapaan haavoittuvainen välityspalvelinta käyttäville kalasteluhyökkäyksille. Nämä kalasteluhyökkäykset pyrkivät muodostamaan väliintulohyökkäyksen houkuttelemalla käyttäjän kirjautumaan hyökkääjän hallitsemalle valesivustolle. Hyökkääjä kykenee uudelleenohjaamaan käyttäjän syöttämät tunnukset aitoon verkkopalveluun, ja odottamaan kun käyttäjä suorittaa biometrisen tunnistautumisen laitteellaan. Biometrisen tunnistautumisen jälkeen hyökkääjä kykenee kirjautumaan käyttäjän tilille.

3.4 Passkey

Inhimillisten virheiden poistamiseksi on kehitetty tietojenkalasteluhyökkäysresistenttejä autentikointimetodeja, jotka eivät vaadi erillisen koodin tai salasanan syöttämistä kirjautumista vaativaan järjestelmään. Suosituin tähän kategoriaan kuuluva tapa on passkey-avainten käyttö. FIDO Alliance (s.a.) kuvaa

passkey-avainten olevan asymmetriseen salaukseen perustuvia avainpareja, joiden yksityinen avain talletetaan identiteetin omistamalle laitteelle ja julkinen avain autentikointia vaativaan palveluun. Asymmetrisen avainparin avulla palvelu voi autentikoida identiteetin lähettämällä haasteen (engl. challenge), jonka identiteetti allekirjoittaa yksityisellä avaimellaan. Allekirjoitus lähetetään autentikoivalle palvelulle, joka kykenee varmentamaan identiteetin vahvistamalla allekirjoituksen aitouden identiteettiin linkitetyllä julkisella avaimella. Passkey-avainten käyttö täyttää myös monivaiheisen tunnistautumisen kriteerit, sillä yksityisen avaimen käyttö vaatii toisen autentikaattorin, kuten PIN-koodin tai biometrisen tunnisteen, käyttöä. (Machani ym. 2020; Passkey Security s.a.) Grassi, Fenton ym. (2017, 8) kuitenkin huomauttavat, että laitteen lukituksen avausta ei tule käyttää omana tunnistautumistapanaan monivaiheisessa tunnistautumisessa, sillä autentikointia vaativa palvelu ei voi varmistaa laitteen lukituksen tilaa tai lukituksen avaamiseen asetettuja vaatimuksia. Hodges ym. (2021) tarkentavatkin WebAuthn-spesifikaatiossa muiden tunnistautumistapojen käytön tapahtuvan laitteen lukituksen avauksen jälkeen, jolloin MFA:n kriteerit täyttyvät.

Passkey-avaimet luokitellaan kalasteluhyökkäysresistenteiksi, sillä ne ovat sidoksissa palvelun DNS-nimeen, ja haasteiden allekirjoitus tapahtuu automaattisesti laitteen toimesta. Passkey-avainten sidonta DNS-nimiin sallii avainten käytön vain rekisteröidyissä palveluissa, jolloin valheellisen DNS-nimen omaava valesivusto ei kykene pyytämään käyttäjää allekirjoittamaan valesivuston lähettämää haastetta, sillä laite ei tunnista valesivuston DNS-nimeä. Allekirjoitusten automaatio poistaa autentikointiprosessista myös ihmisen toiminnan, jolloin allekirjoitukseen tarvittavaa yksityistä avainta ei kyetä antamaan hyökkääjille inhimillisen virheen kautta. (Passkey Security s.a.)

3.5 Adaptiivinen autentikointi

Adaptiivisessa autentikointiprosessissa tunnistautumismenetelmien määrä vaihtelee identiteetille lasketun riskiarvon perusteella. Riskiarvo lasketaan riskitekijöiden avulla, joita ovat kirjautumiskontekstiin liittyvät muuttujat kuten IP-osoite, sijainti ja kirjautumisessa käytetyn laitteen tai selaimen tiedot. Kirjautumistapahtumissa havaittujen muuttujien avulla identiteetin kirjautumiskonteksti voidaan profiloida, ja muuttujista koostettua profiilia voidaan käyttää riskiarvon

laskemisessa vertailemalla profiloituja arvoja seuraavan kirjautumistapahtuman muuttujiin. Adaptiivisen autentikoinnin tarkoituksena on parantaa autentikoinnin käytettävyyttä vähentämällä MFA-metodien määrää normaaleiksi määritellyissä kirjautumiskonteksteissa. (Adaptive Multi-Factor Authentication s.a.)

Adaptiivisen autentikoinnin profilointia on kuitenkin pidetty käyttäjien yksityisyyttä loukkaavana. Käyttäjien profilointi on Arias-Cabarcosin ym. (2019, 24) mukaan ongelmallista, sillä kolmannet osapuolet voivat käyttää dataa identiteettien seurantaan. Kontekstuaalisen datan väärentäminen on myös mahdollista, jolloin hyökkääjä kykenee vähentämään MFA-metodien määrää kirjautumistilanteessa. Arias-Cabarcos ym. (2019, 20) kertovatkin uusien, implisiittisten autentikointimekanismien kehityksen olevan tarpeellista adaptiivisen autentikoinnin kehittämiseksi.

3.6 Implisiittinen autentikointi

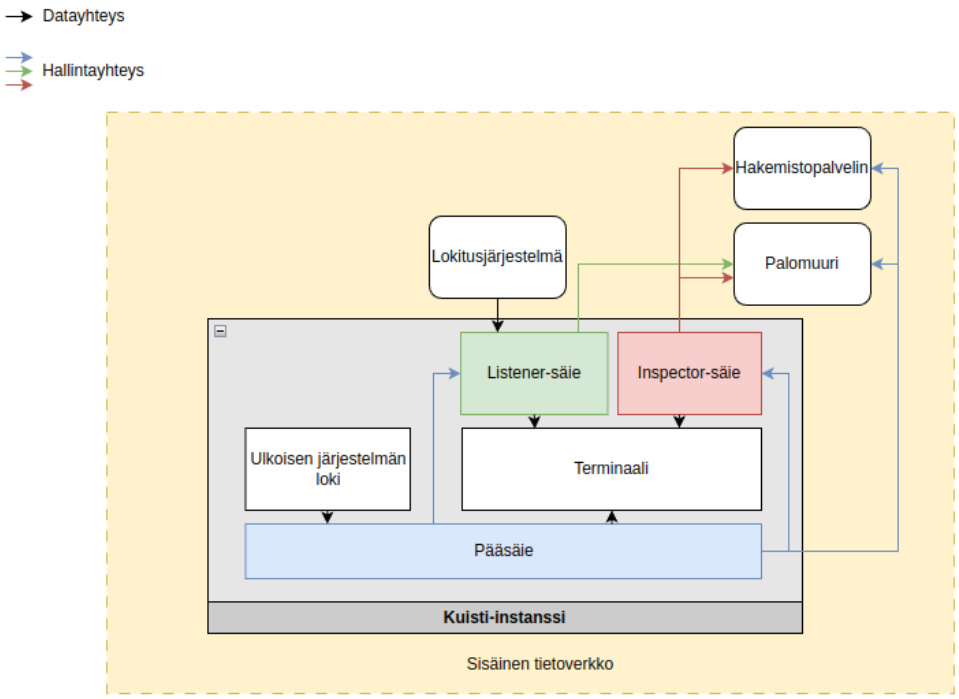
Implisiittiset autentikointimetodit ovat käyttäjälle näkymättömiä tunnistautumistapoja, jotka hyödyntävät autentikoinnissa käyttäjän normaaleja rutiineja, jolloin autentikointimetodien käyttö ei vaatisi käyttäjältä erillisiä toimenpiteitä. Jakobssonin ym. (2015) ja Yaon ym. (2017) implisiittisiin autentikointimetodeihin liittyneet tutkimukset ovat painottuneet biometristen tunnistetietojen käyttöön, mutta deterministisiä autentikointimenetelmiä ei ole tutkimuksissa ehdotettu. Khan ym. (2015, 1–2) mainitsevatkin implisiittisten autentikointimetodien tutkimuksen painottuvan biometristen tunnistautumistapojen kehitykseen ja keskittyvän mobiililaitteiden käyttäjien autentikointiin.

Biometristen tunnistautumistapojen toimivuus perustuu kuitenkin todennäköisyyteen, tehden tunnistautumistavasta haavoittuvaisen virheelliselle autentikoinnille. Khan ym. (2015, 2) kertovatkin biometriikkaa käyttävien, implisiittisten autentikointimetodien toimivan joko välimaastona täysin suojaamattomissa laitteissa tai omana tunnistautumistapanaan laitteen ensisijaisen autentikointimetodin ohella. Havainnot avaavat mahdollisuuden tutkia, pystyisikö implisiittistä autentikointia suorittaa deterministisin metodein siten, että käyttäjien päivittäisiä rutiineja hyödynnetään autentikointiprosessissa.

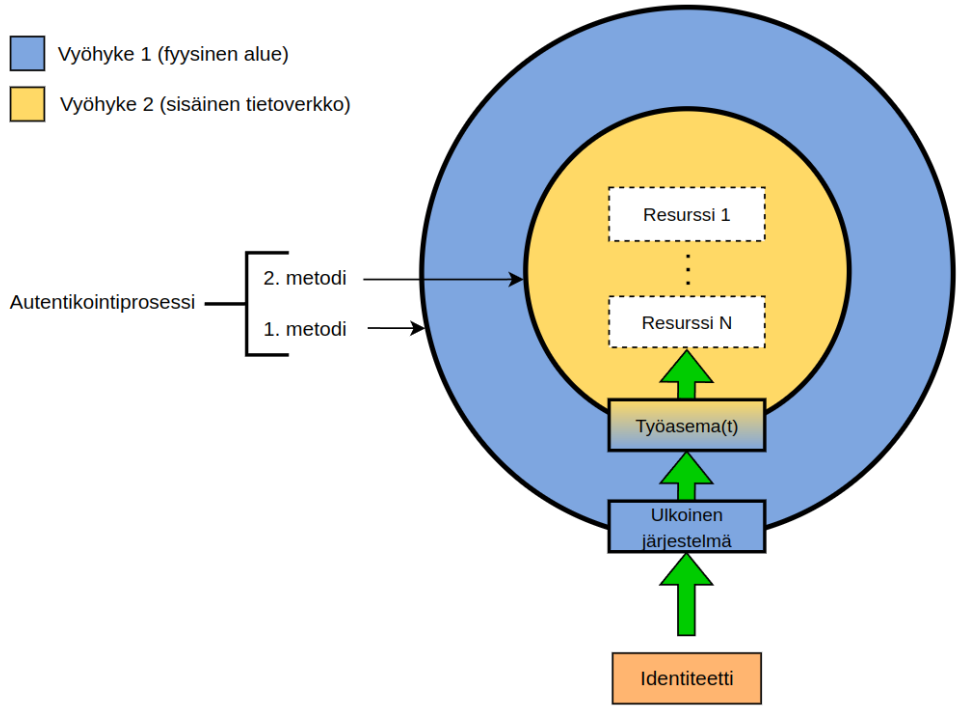
4 TOTEUTUS

4.1 Kartoitus

Ennen varsinaisten muutosten aloittamista Kuisti-järjestelmän nykytila kartoitettiin Kuistin alustavan dokumentaation pohjalta kehitettävien komponenttien löytämiseksi ja rajaamiseksi. Tutkimuksen pääkohteena oli autentikointimenetelmien kehitys, joten järjestelmän nykytilan kartoituksessa pyrittiin tunnistamaan autentikointimenetelmiin ja pääsynhallintaan liittyneet komponentit. (Kuva neljä.) Kartoituksen aikana tarkasteltiin myös Kuistin haluttua vaikutusta monivaiheiseen autentikointiprosessiin, joka on esitettynä kuvassa viisi.



Kuva 4. Kuistin komponenttikartta ennen muutoksia

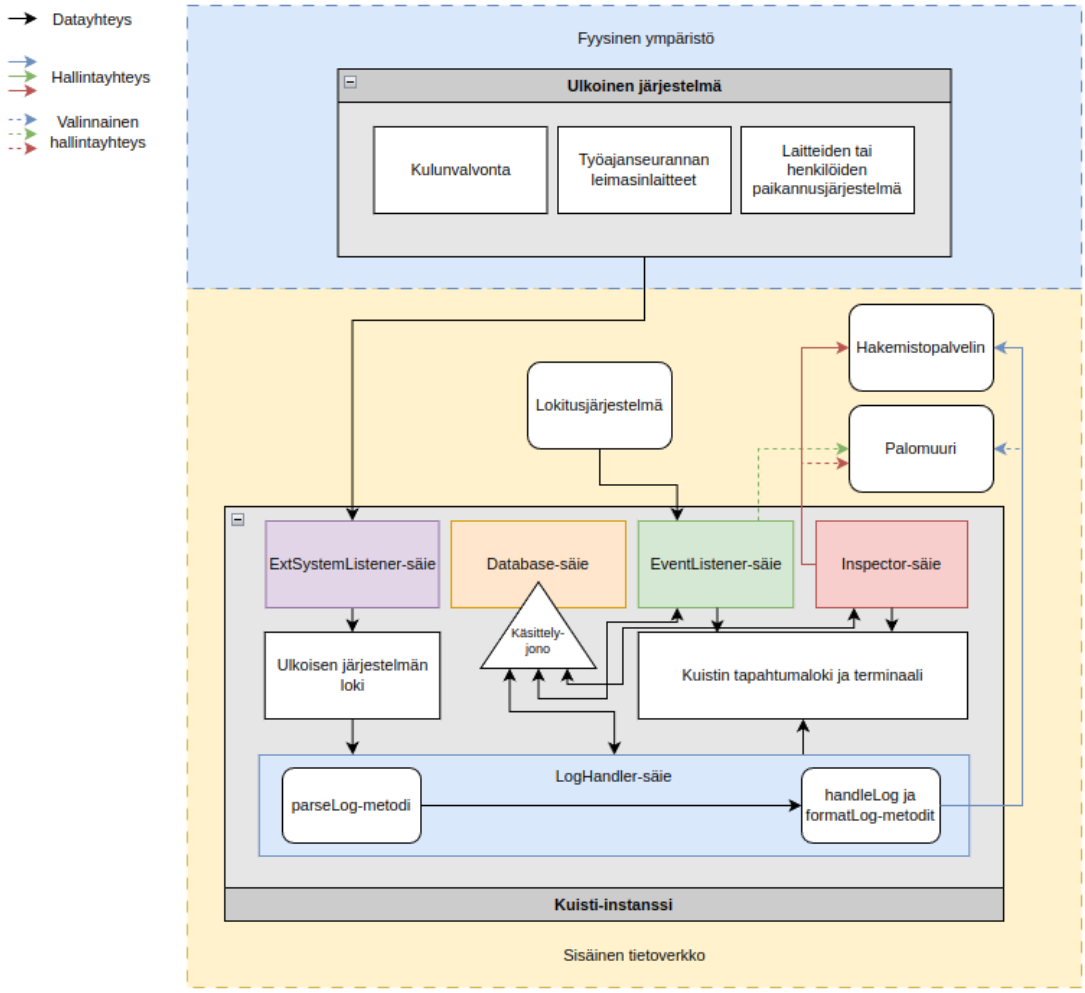


Kuva 5. Monivaiheinen autentikointiprosessi Kuistin kanssa

Kuvan viisi autentikointiprosessissa ulkoinen järjestelmä toimii ensimmäisenä ja työasemalle kirjautuminen toisena autentikointimetodina. Autentikoitava identiteetti pääsee tällöin sisäisen tietoverkon resursseihin käsiksi ensin tunnistautumalla ympäristössä olevan laitteiston avulla fyysiselle alueelle, minkä jälkeen käyttäjän tulee kirjautua työasemalle käyttäjätunnuksellaan, jolloin autentikointiprosessi on monivaiheinen. Tällä tavoin ulkoisten järjestelmien dataa pyrittiin hyödyntämään uuden, implisiittisen autentikointimetodin luonnissa, joka tulisi lisätä sisäisen tietoverkon autentikointiprosessiin Kuistia käyttäen. Kuisti-järjestelmän komponenttikartoitus paljasti puutteita asetettujen tavoitteiden suhteen: kartoitushetkellä järjestelmä oli suunniteltu asennettavaksi suoraan ulkoista järjestelmää ylläpitäneelle palvelimelle, jolloin useampien lähteiden käyttö käyttäjän sijainnin varmentamisessa oli mahdotonta. (Kuva neljä.) Kartoituksen aikana järjestelmän komponenttien havaittiin olleen myös tiukasti yhteydessä Kuistin ydintoimintoihin, mikä teki komponenttien muokkauksesta ja luonnista haastavaa. Näitä ongelmia päätettiin lähteä korjaamaan ensin kytkemällä ydintoimintoja irti toisistaan omiksi moduuleikseen, jotta olemassa olleiden toimintojen kehitys ja uusien toiminnallisuuksien luonti helpottuisi.

4.2 Muutokset

Lähtötilanteessa Kuisti tuki vain kulunvalvontajärjestelmän paikallisen lokin seurantaa, jolloin Kuisti-ohjelmisto oli asennettava kulunvalvonnan lokia ylläpitäneelle palvelimelle, mikä esti useampien järjestelmien käytön autentikointiprosessissa. Tämä oli ongelmallista, sillä Kuistin olisi tarkoitus varmentaa identiteettien sijainnit yhden tai useamman lähteen perusteella. Ydintoimintojen irtikytkennän jälkeen edellä mainittu ongelma korjattiin lisäämällä kuvassa kuusi esitetty ExtSystemListener-säie, jonka tehtävänä oli vastaanottaa ulkoisten järjestelmien lähettämät lokimerkinnät ja kirjata ne Kuisti-palvelimen lokiin käsittelyä ja seurantaa varten.



Kuva 6. Kuistin komponenttikartta muutosten jälkeen

ExtSystemListener-säikeen on tarkoitus vastaanottaa ulkoisten järjestelmien syöttämää dataa, ja kirjata vastaanotetut lokimerkinnät erilliseen lokitiedos-

toon odottamaan käsittelyä. Tämän jälkeen LogHandler-säie lukee ja käsittelee lokimerkinnät. (Kuva kuusi.) Aikaisemmassa versiossa Kuisti käsitteli lokimerkinnät pääsäikeessä, mikä lukitsi pääsäikeen käytön Kuistin ydintoiminnon (lokien käsittely) suoritukseen. Pääsäikeen vapautus datan käsittelystä mahdollisti pääsäikeen käytön myös muun koodin suoritukseen Kuisti-instanssin ylläpidon ohella. Säikeitä käytettiin yksinkertaisemman implementoinnin ja siirräntään (engl. IO bound) painottuneiden operaatioiden vuoksi aliprosessien sijasta, vaikka Python Software Foundation (s.a.) mainitseekin aliprosessien hyödyntävän moniytimisten prosessoreiden laskennallisia resursseja säikeitä paremmin.

Datan käsittely piti olla konfiguroitavissa, sillä integroitavien järjestelmien lähettämän datan laatu voi vaihdella paljonkin eri järjestelmien välillä. Lokitietojen käsittelyn konfiguroitavuus mahdollistettiin muokattavin parseLog- ja handleLog-metodein, joiden avulla järjestelmän ylläpitäjä kykeni itse määrittämään ohjelmalogiikan lokitietojen jäsentelylle ja käsittelylle. Oletusarvoisesti Kuisti jäsensi ja käsitteli lokimerkinnät konfiguraatiotiedostossa määritetyin tavoin. Konfiguraatiotiedoston käyttöä käsitellään myöhemmin, ja lokitietojen oletusarvoinen käsittelyprosessi on kuvattuna Kuistin prosessikaaviossa liitteessä yksi.

Kuistin ja palomuurin väliset hallintayhteydet erotettiin Kuistin ydintoiminnoista valinnaisiksi ominaisuuksiksi, sillä Kuisti oli lähtötilanteessa yhteensopiva vain OPNsense-palomuurituotteen kanssa. Valinnaisuuden tarkoituksena oli modularisoida palomuuriin liitetyt ominaisuudet, jotta järjestelmän ylläpitäjä kykenisi tarvittaessa itse implementoimaan käyttämänsä palomuurituotteen toiminnallisuudet Kuistiin verkkotason pääsynhallintaa varten, jolloin Kuisti ei ole riippuvainen käytettävästä palomuurituotteesta. (Kuva kuusi.) Palomuuriin kytkettyjen ominaisuuksien valinnaisuus mahdollisti Kuistin käyttöönoton myös sellaisissa ympäristöissä, joissa suodatussääntöjen hallinta ei ole mahdollista. Hakemistopalvelimen käyttöä sen sijaan vaadittiin, sillä työasemille kirjautumista rajoitettiin hakemistopalvelimen ryhmäkäytäntöjen kautta (liite yksi). Työasemille kirjautuminen toimi osana kuvassa viisi esitettyä autentikointiprosessia, jolloin hakemistopalvelimen käyttö oli välttämätöntä.

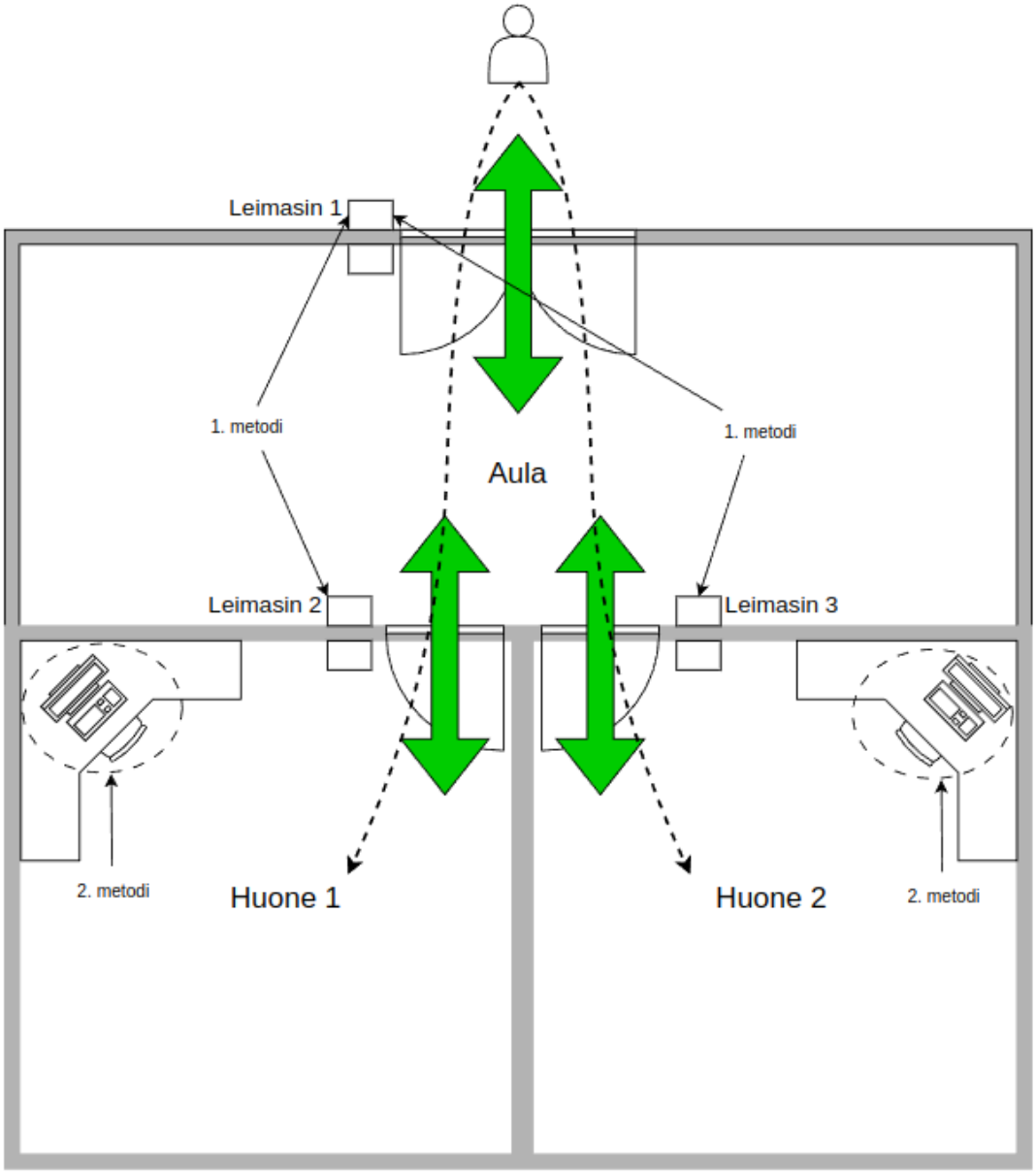
Database-säikeen ja säikeeseen liitetyn käsittelyjonon tehtävänä oli käyttäjä-, suodatussääntö- ja huonetietokannan ylläpito. Kuistin aiemmassa versiossa aktiivisia käyttäjiä ja käyttäjiin liittyneitä suodatussääntöjä säilöttiin ohjelman muistissa sanakirja-tietotyypin (engl. dictionary) sisällä, joka oli saatavilla kaikille säikeille. Sanakirja-tietotyypin avulla pystytään muodostamaan avain-arvo-pareja, jolloin määrätty avain (käyttäjä) pystytään yhdistämään haluttuun arvoon (suodatussääntö tai huone). Sanakirja-tietotyypin käyttö säikeiden välillä riitti Kuistin aiemmissa versioissa konseptin testaamisen vuoksi, mutta säikeiden määrän lisääntyessä tietokanta siirrettiin erillisen Database-säikeen alle (kuva kuusi). Tietokannan hallinta siirrettiin erilliseen säikeeseen, sillä samanaikainen pääsy monesta eri lähteestä voi aiheuttaa virheellisiä luku- ja kirjoitusoperaatioita korruptoiden tietokantaan talletetun datan (SolarWinds s.a.). Uudistetussa versiossa tietokannan luku- ja kirjoitusoperaatiot suoritettiin Googlen (s.a.) suositusten mukaisesti käsittelyjonon kautta, jolloin Database-säie kykeni käsittelemään jonoon syötetyt operaatiot aikajärjestyksessä.

4.3 Ulkoisten järjestelmien hyödyntäminen

Ulkoisten järjestelmien lokitietoja oli tarkoitus hyödyntää autentikointiprosessin ensimmäisessä metodissa, joten lokitietojen täytyi sisältää vähintään tunnistetun identiteetin ja alueen nimi, jotta tunnistettu identiteetti kyettäisiin yhdistämään tiettyyn fyysiseen alueeseen. Tämän lisäksi lokitietojen olisi hyvä kertoa tunnistetun identiteetin kulkusuunta, jotta tiedettäisiin, onko identiteetti saapunut vai lähtenyt alueelta. Nämä vaatimukset rajaavat käytettävien järjestelmien valikoiman sellaisiin järjestelmiin, joiden toimintaperiaatteet painottuvat fyysiseen pääsynhallintaan tai seurantaan, kuten kulunvalvontajärjestelmiin. Identiteettien sijaintia voidaan seurata myös olemassa olevalla verkkolaitteistolla, jos laitteisto tukee WiFi RTLS -tekniikan käyttöä. Elisa (s.a.) ja Peppin (2024) mainostavat WiFi RTLS -tekniikan paikantavan laitteet ja henkilöt reaaliaikaisesti, joten kyseiseen tekniikkaan pohjautuvien järjestelmien dataa voitaisiin kulunvalvontajärjestelmien tapaan hyödyntää autentikointiprosessin ensimmäisessä metodissa.

Ulkoisten järjestelmien lokitietojen avulla käyttäjän fyysinen sijainti varmennettiin ennen työasemalle kirjautumista joko yhtä tai useampaa lokimerkintää käyttäen. Viimeisimmästä vastaanotetusta lokimerkinnästä saatiin selville

käyttäjän reaaliaikaisin sijainti, mutta yksittäinen lokimerkintä ei kertonut miten tai mitä reittiä pitkin käyttäjä pääsi mainitulle alueelle. Saapumisreitin varmentaminen on tärkeää, jos tietty resurssi on saatavissa vain ennalta määrätyn reitin kautta, kuten kuvan seitsemän esimerkistä voidaan havaita.



Kuva 7. Autentikointiprosessin metodit ja reitit sovitettuna fyysiseen ympäristöön

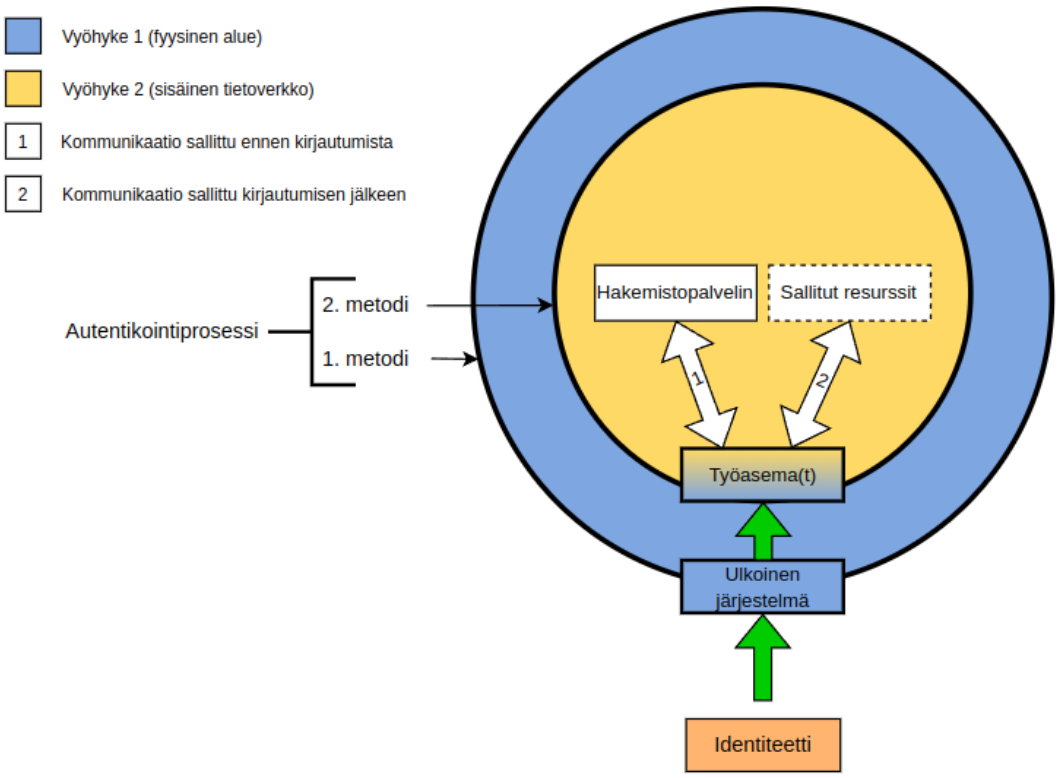
Kuvassa seitsemän on esitetty kaksi leimasinlaitteista koostuvaa reittiä, joita voidaan käyttää autentikointiprosessin ensimmäisenä metodina. Kuvassa seitsemän henkilön on ensin tunnistauduttava kiinteistön pääovien luona olevalla leimasimella, jonka jälkeen henkilö voi siirtyä haluamaansa työhuoneeseen tunnistautumalla ovien vieressä olevilla leimasinlaitteilla kirjautuakseen työasemille normaalisti käyttäjätunnuksellaan. Fyysinen reitti työasemille voidaan

tällä tavoin päättää henkilön läsnäolon varmentamiseksi, jolloin työasemille kirjautuminen voidaan estää, jos henkilö ei ole kulkenut ennalta määrättyä reittiä työhuoneeseen. Tällöin ulkoisen järjestelmän tuottamaa dataa voidaan hyödyntää implisiittisessä autentikoinnissa, sillä ensimmäisen autentikointimetodin aikana tehdyt toimet kuuluvat henkilön normaaliin rutiiniin, eikä autentikointi vaadi henkilöltä lisätoimenpiteitä.

Ennen muutostöitä Kuisti-järjestelmä tuki vain yhden ulkoisen järjestelmän lokin seurantaa, kuten kuvasta neljä voidaan havaita. Kuvassa kuusi esitetty ExtSystemListener-säie mahdollisti lokitietojen keruun useammasta ulkoisesta järjestelmästä, jolloin lokien yhdistäminen reiteiksi oli mahdollista. Autentikoinnissa käytettyjen järjestelmien ja reittien konfigurointia käsitellään myöhemmin.

4.4 Verkkotason pääsynhallinta

Verkkotason pääsynhallinta toteutettiin dynaamisin suodatussäännöin, jotka luotiin tai poistettiin sisään- ja uloskirjausten yhteydessä. Dynaamisten suodatussääntöjen avulla työaseman pääsy tietoverkon resursseihin pystyttiin estämään ennen autentikointiprosessin toista vaihetta, mikä vähensi työaseman implisiittisiä oikeuksia verkkoon (kuva kahdeksan). Implisiittisen pääsyn rajaus on tärkeää etenkin Zero Trust -arkkitehtuuria noudattavissa ympäristöissä, jotta käytettävien resurssien pääsynhallintaan liittyvät epävarmuustekijät, kuten implisiittiset pääsyoikeudet, saataisiin minimoitua (Rose ym. 2020).



Kuva 8. Verkkotason pääsynhallinnan vaiheet sovitettuna autentikointiprosessiin

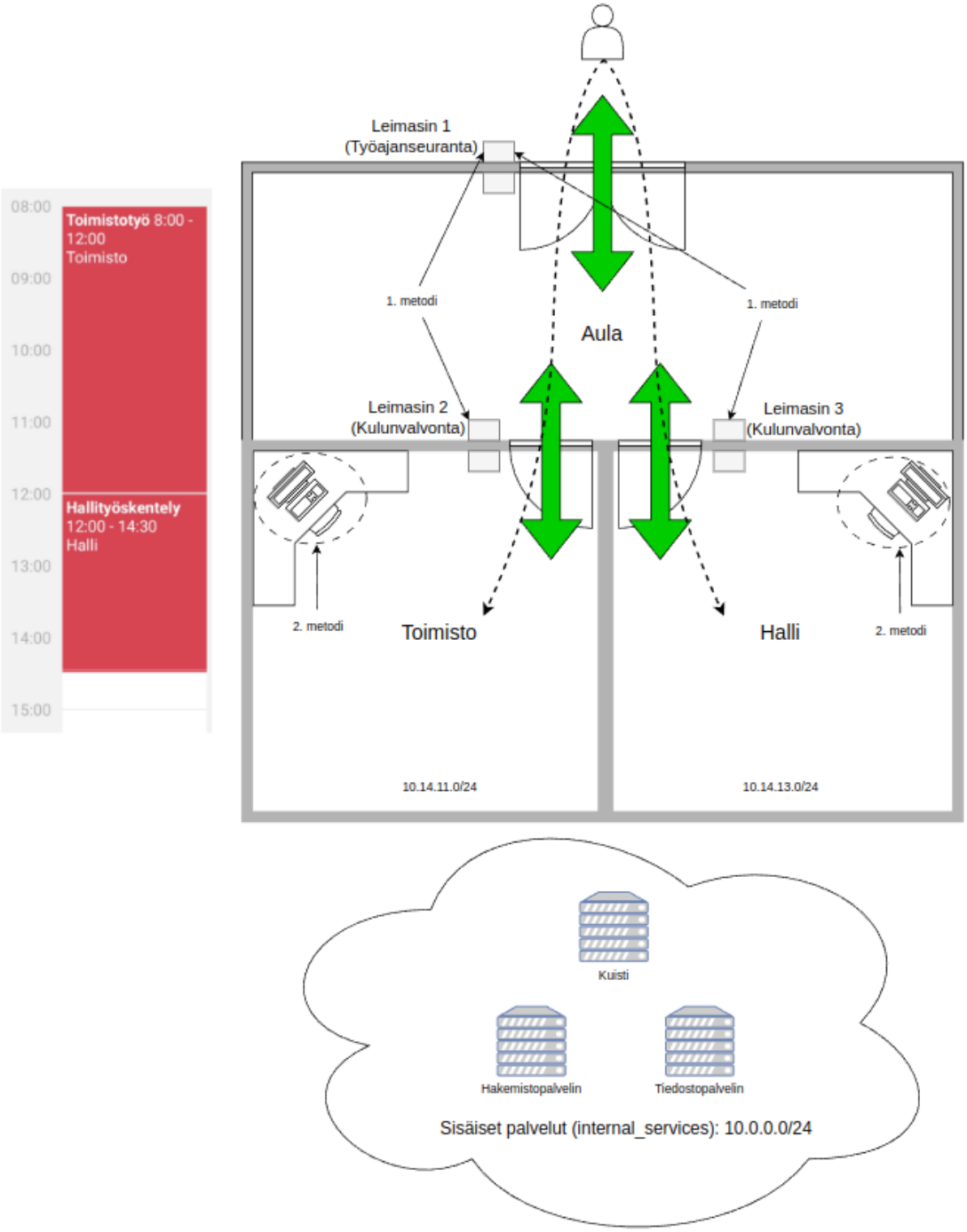
Kuvassa kahdeksan visualisoidaan, miten verkkotason pääsynhallinta rajasi työaseman implisiittisiä pääsyoikeuksia tietoverkon resursseihin Kuistin muokkaamassa autentikointiprosessissa. Ennen työasemalle kirjautumista työasema pystyi kommunikoimaan vain hakemistopalvelimen kanssa synkronoidakseen uusimmat verkkoympäristöön liittyneet tiedot, ja pitääkseen kirjautumiseen vaaditun tietoliikenneyhteyden avoinna. Onnistuneen kirjautumisen jälkeen palomuurille luotiin automaattisesti käyttäjälle määritetyt, roolikohtaiset suodatussäännöt, jotka linkitettiin kirjautumistapahtumaan seurantaa varten. Suodatussääntöjen seurantaa ylläpiti Inspector-säie, joka liitteessä yksi esitetyin tavoin poisti käyttäjää varten luodut palomuurisäännöt, jos ne eivät olleet tarkastushetkellä aktiivisessa käytössä. Tarkastusten intervallit määritettiin Kuistin konfiguraatiotiedostoon, jonka käyttöä tarkastellaan myöhemmin.

4.5 Testaus

4.5.1 Testiskenaario

Kehitetyn Kuisti-järjestelmän toimintaa testattiin kuvan yhdeksän kaltaisessa, simuloidussa testiympäristössä. Testiskenaario suoritettiin kahdesti tulosten

reliabiliteetin varmentamiseksi. Testien tarkoituksena oli varmistaa, että käyttäjä pääsi ennalta määrättyihin verkkoresursseihin käsiksi työasemaltaan ainoastaan silloin, kun käyttäjä oli tunnistautunut monivaiheisesti verkkoympäristöön fyysisiä elementtejä ja käyttäjätunnustaan käyttäen. Testien aikana varmistettiin myös Kuisti-järjestelmän asennukseen liittyvien muutosten ja skriptien toiminta.



Kuva 9. Testiskenaario visualisoituna

Kuvan yhdeksän skenaario visualisoi fiktiivistä työpäivää, joka koostuu toimisto- ja hallityöskentelystä. Testiskenaario alkaa, kun työntekijä leimaa itsenä sisään kiinteistöön työajanseurannan leimasinpäätteellä, ja siirtyy toimistoon aulan kautta. Klo 10.30 työntekijä lukitsee työasemansa tunnin ajaksi ja poistuu kiinteistöstä toimiston ja työajanseurannan leimasinten kautta lounaalle, jonka jälkeen hän jatkaa työskentelyään toimistossa klo 11.30. Toimistotyön päätyttyä työntekijä siirtyy hallin puolelle normaalisti leimaamalla itsensä ulos toimistosta, ja leimaamalla itsensä halliin. Hallityöskentelyn päätyttyä työntekijä poistuu kiinteistöstä leimaamatta itseään ulos hallista. Työntekijä ei myöskään leimaa itseään ulos aulan työajanseurannan leimasinlaitteella työpäivän päätyttyä, jolloin hallin työaseman sessiot jäävät avoimiksi.

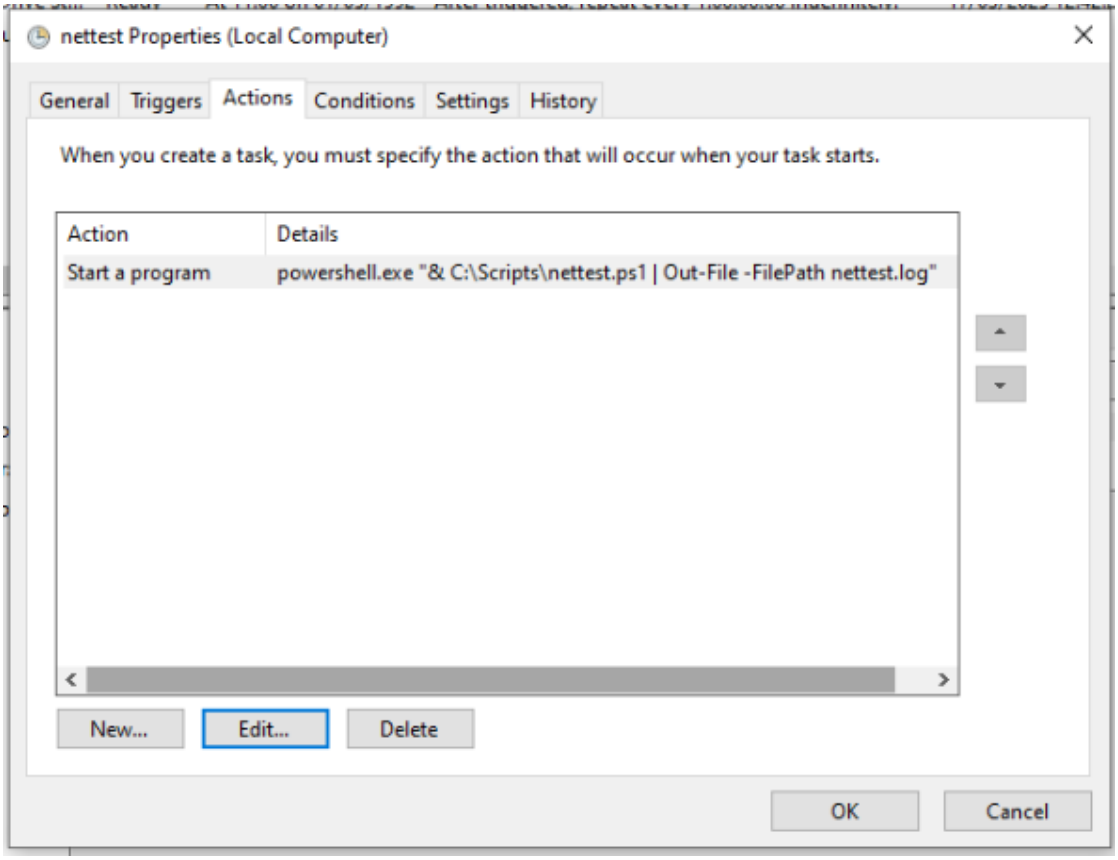
4.5.2 Verkkotason pääsynhallinnan monitorointi

Testiskenaarion aikana sisäisten ja ulkoisten palveluiden saatavuutta monitoroitiin työasemilta käsin kahdessa eri osiossa. Testiskenaarion ensimmäisen osion (toimistotyö) tarkoituksena oli varmentaa Kuistin toimivuus normaaleissa olosuhteissa eli silloin, kun työntekijä käyttää kiinteistön fyysisiä elementtejä suunnitellusti työpäivän aikana. Skenaarion toinen osio (hallityöskentely) testasi Kuistin Inspector-säikeen toimintaa, jonka tarkoituksena oli poistaa epäaktiiviset suodatussäännöt aikakatkaisun yhteydessä implisiittisen pääsyn minimoimiseksi. (Kuvat 10 ja 11).

nettest.ps1

```
1   $remoteHosts = @("virtual.ictlab.fi", "learn.xamk.fi")
2
3
4   while ($true) {
5
6       echo "----- Sisäiset palvelut -----"
7       echo "Päiväys: $(Get-Date -UFormat '%Y-%m-%d %H:%M:%S')"
8
9       & "C:\Program Files (x86)\Nmap\nmap.exe" "-n" "10.0.0.0/24"
10
11      echo "----- Ulkoiset palvelut -----"
12      echo "Päiväys: $(Get-Date -UFormat '%Y-%m-%d %H:%M:%S')"
13
14      foreach ($remoteHost in $remoteHosts) {
15
16          Test-NetConnection -port 443 $remoteHost
17
18      }
19
20      Sleep -Seconds 300
21
22  }
```

Kuva 10. Kuvankaappaus palveluiden saatavuutta mitanneesta PowerShell-skriptistä

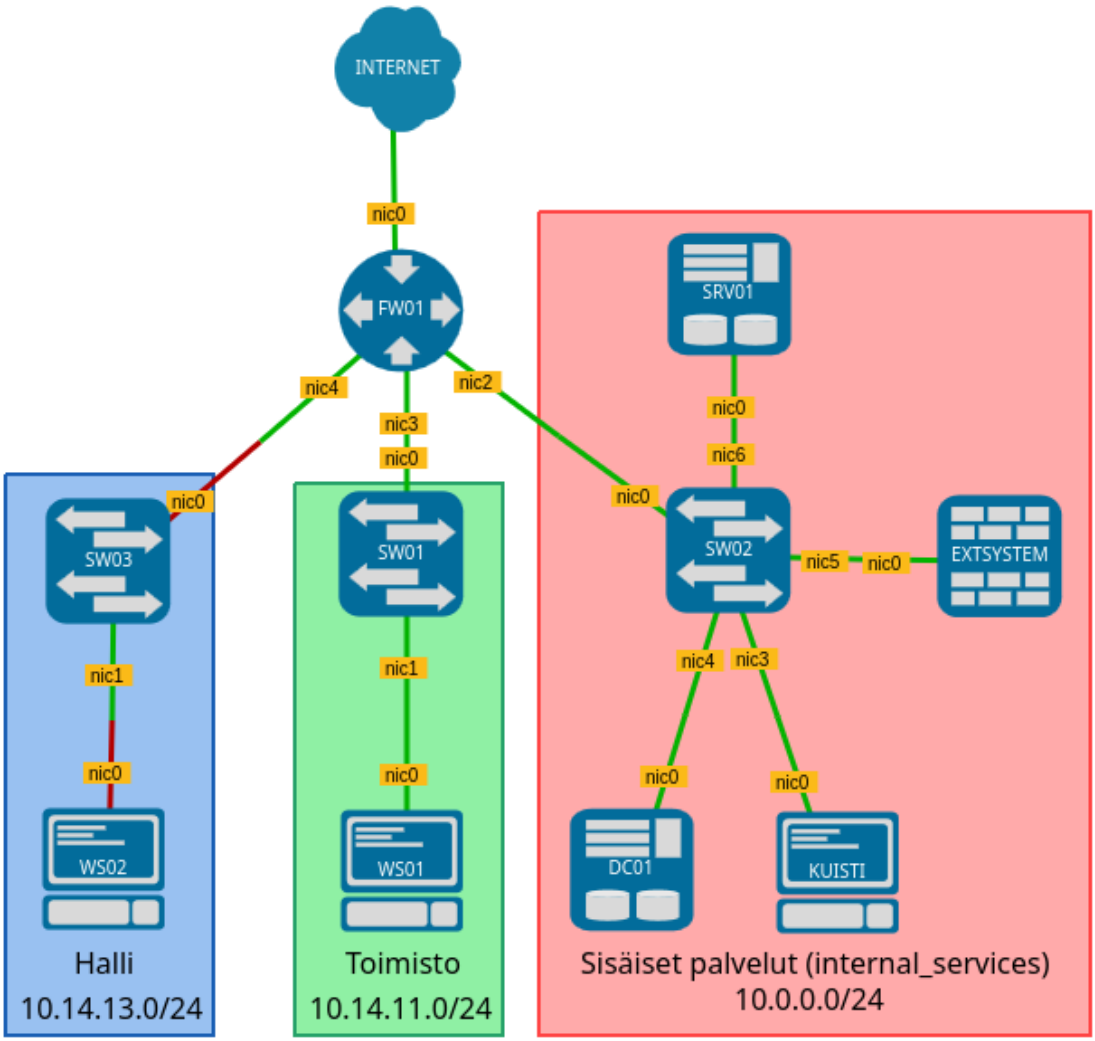


Kuva 11. Kuvankappaus työasemalle WS01 luodusta ajoitetusta tehtävästä

Kuvan 10 skripti tarkasti viiden minuutin välein sisäisten ja ulkoisten palveluiden saatavuutta. Sisäisten palveluiden saatavuutta monitoroitiin nmap-työkalulla, ja ulkoisten palveluiden saatavuutta monitoroitiin Test-NetConnection-komennolla, joiden tulosteet ohjattiin nettest.log-tiedostoon analyysiä varten. Skripti määritettiin käynnistymään automaattisesti työaseman käynnistyksen yhteydessä kuvan 11 mukaisella, ajoitetulla tehtävällä.

4.5.3 Virtuaalinen ympäristö

Edellä mainittu testiskenaario suoritettiin kuvan 12 mukaisessa virtuaalisessa ympäristössä Kaakkois-Suomen ammattikorkeakoulun VirtualLab-alustalla. Virtuaalisen ympäristön topologia simuloi kuvassa yhdeksän esitettyä testiympäristöä, ja työajanseuranta- sekä kulunvalvontajärjestelmää simuloitiin EXTSYSTEM-palvelimen avulla. Virtuaalisessa ympäristössä käytettyjen laitteiden verkkoasetukset ovat listattuna taulukossa yksi.



Kuva 12. Virtuaalisen ympäristön looginen topologia

Taulukko 1. Virtuaalisen ympäristön IP-osoitetaulukko

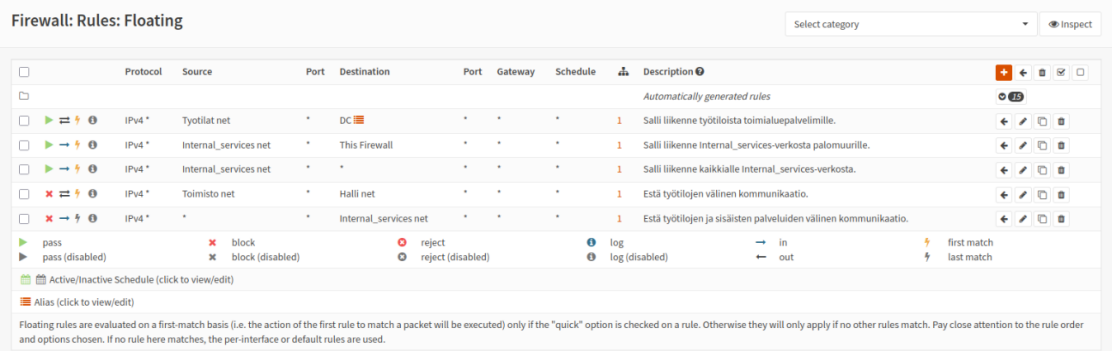
LAITTEEN NIMI	**TUOTE** **ROOLI(T)**	**LIITÄNTÄ**	**IP-OSOITE/CIDR** **[YHDYSKÄYTÄVÄ]** **[DNS]**
FW01	OPNsense 24.7.12_4-amd64 Palomuuri	nic0	DHCP
		nic2	10.0.0.1/24
		nic3	10.14.11.1/24
		nic4	10.14.13.1/24
DC01	Windows Server 2019 Toimialue- ja nimipalvelin, lokitus-järjestelmä	nic0	10.0.0.10/24 10.0.0.1 127.0.0.1
KUISTI	Ubuntu 22.04.3 LTS Kuisti-palvelin	nic0	10.0.0.12/24 10.0.0.1 10.0.0.10

EXTSYSTEM	Ubuntu 22.04.3 LTS Työajanseuranta- ja kulunvalvontajärjestelmä	nic0	10.0.0.13/24 10.0.0.1 10.0.0.10
SRV01	Ubuntu 22.04.3 LTS FTP-palvelin	nic0	10.0.0.14/24 10.0.0.1 10.0.0.10
WS01	Windows 10 22H2 Toimiston työasema	nic0	DHCP
WS02	Windows 10 22H2 Hallin työasema	nic0	DHCP

Virtuaalinen testiympäristö koostui kolmesta eri verkkosegmentistä, joita yhdisti OPNsense-palomuurilaite (kuva 12). Työtiloille tarkoitetut verkkosegmentit nimettiin VirtualLab-ympäristöön luodussa topologiassa vastaamaan kuvan yhdeksän testiympäristön huoneita, ja sisäisten palveluiden verkkosegmentti yhdistettiin OPNsense-palomuurin nic2-verkkoliitäntään. Sisäisiin palveluihin kuulunut tiedostopalvelin oli virtuaalisessa ympäristössä yksinkertainen FTP-palvelin, ja koko topologian lokitusjärjestelmänä toimi WEC-palvelin (Windows Event Collector), joka soveltuu Windows-laitteiden tietoturvatapahtumien keruuseen ja seurantaan (Microsoft 2024b). WEC-palvelin valittiin testiympäristön lokitusjärjestelmäksi, sillä siihen liittyvät ominaisuudet olivat sisäänrakennettuina topologiaan lisätyissä Windows-laitteissa. Lokitusjärjestelmäksi käy myös kaikki sellaiset ratkaisut, jotka kykenevät seuraamaan työaseman kirjautumisiin ja lukituksiin liittyviä tapahtumia, ja lähettämään ne eteenpäin Kuisti-palvelimelle.

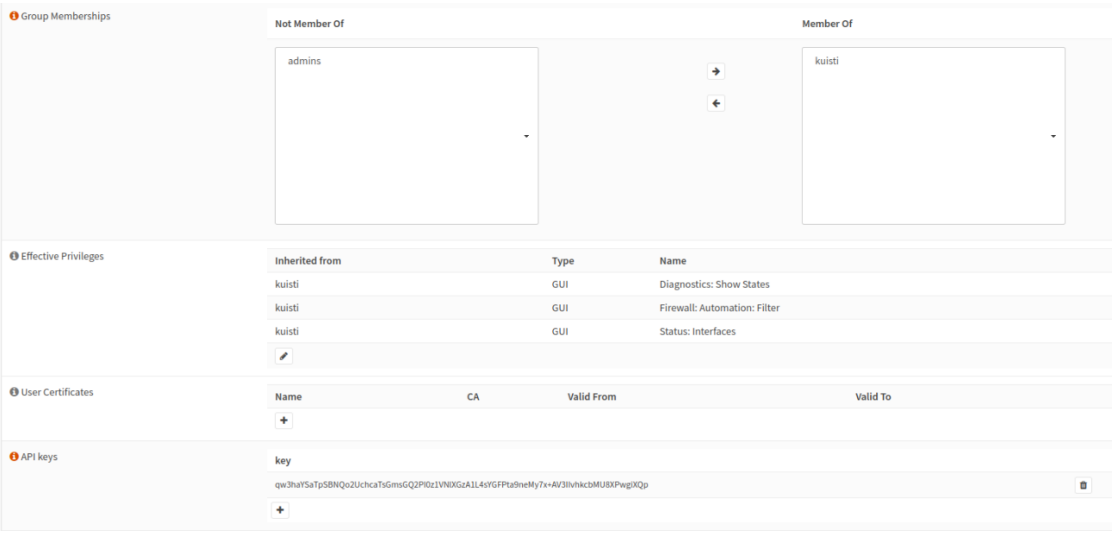
4.5.4 OPNsense-palomuurin konfigurointi

Palomuurilaitteen tehtävänä oli oletusarvoisesti sallia toimiston ja hallin työasemien sekä toimialuepalvelimen välinen kommunikointi työasemakirjautumisia varten ja estää kaikki muu liikenne sisäisten palveluiden verkkosegmenttiin. Tällä tavoin pääsyä sisäisiin palveluihin kyettiin hallinnoimaan Kuisti-palvelimelta käsin. Pääsy ulkoverkkoon oli oletusarvoisesti sallittu vain sisäisten palveluiden verkkosegmentistä, kuten kuvassa 13 esitetyistä suodatussäännöistä voidaan havaita.



Kuva 13. Palomuurille FW01 luodut palomuurisäännöt

Kuva 13 sisältää OPNsense-palomuurille luodut suodatussäännöt, joita käytettiin testiskenaarion suorituksen aikana. Kuvassa näkyvät lähde- ja kohdeosoitteet käyttivät verkkosegmentteihin ja toimialuepalvelimeen linkitettyjä aliaksia, jotka yhdistyivät suoraan kuvassa 12 esitettyihin verkkoalueisiin ja taulukossa yksi mainittuun toimialuepalvelimen IP-osoitteeseen. Näiden suodatussääntöjen tarkoituksena oli rajata verkkotason implisiittiset pääsyoikeudet riittävän pieniksi, jotta Kuistin pääsynhallinnan ominaisuuksia pystyttäisiin monitoroimaan ja seuraamaan. OPNsense-palomuurille luotiin myös palvelukäyttäjä ja -ryhmä Kuistia varten, jotta Kuisti-palvelin voisi hallita suodatussääntöjä ja sääntöihin linkitettyjä tilamerkintöjä automaattisesti (kuva 14).



Kuva 14. Kuisti-palvelukäyttäjän pääsyoikeudet palomuurilla FW01

Kuvan 14 pääsyoikeudet sallivat OPNsense-palomuurille luodun Kuisti-palvelukäyttäjän pääsyn suodatussääntöjen ja sääntöihin linkitettyjen tilatietojen hallintaan. Pääsyoikeudet palomuurille pyrittiin pitämään mahdollisimman pie-

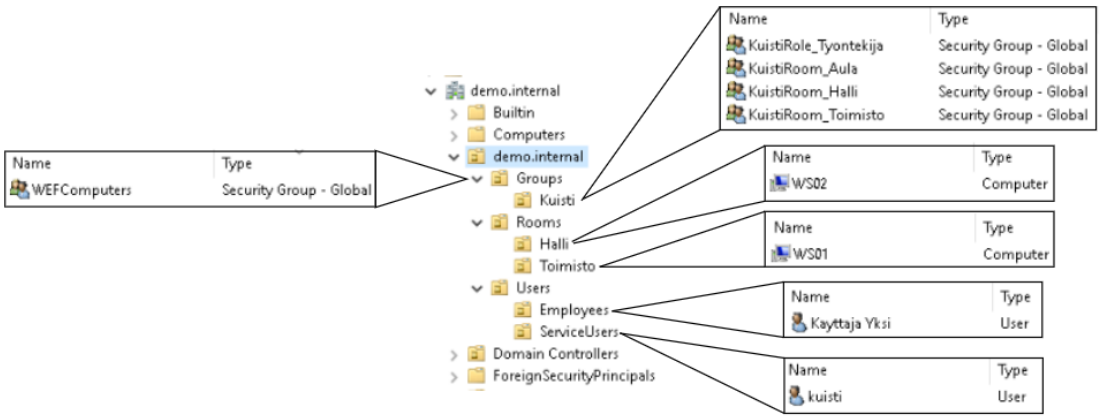
ninä PoLP-periaatetta (Principle of Least Privilege) noudattaen, minkä tarkoituksena on antaa pääsyoikeudet vain tarvittaviin ominaisuuksiin. Turhat tai liian laajat pääsyoikeudet voivat vaarantaa resurssin ylläpitämän datan luottamuksellisuuden ja eheyden. (Cole ym 2008, 154.) Kuisti-palvelukäyttäjälle anottiin suodatussääntöjen ja tilatietojen hallinnan lisäksi oikeus tarkastella palomuurin verkkoliitäntöjen tietoja, jotta Kuisti kykenisi yhdistämään suodatussäännöt oikeisiin verkkoliitäntöihin.

OPNsense-palomuurin nic3- ja nic4-verkkoliitäntöihin määritettiin myös DHCP-palvelimet työasemia varten. Kumpaankin verkkoliitäntään määritetyt DHCP-palvelimet jakoivat IP-osoitteita 10.14.1X.50–100/24-osoitealueiden väliltä, ja kummankin verkkosegmentin oletusyhdyskäytävänä toimi taulukossa yksi mainitut FW01:n nic3- ja nic4-verkkoliitäntöjen IP-osoitteet. Edellä mainittujen muutosten ohella OPNsense-palomuuri toimi täysin oletusasetustensa mukaisesti.

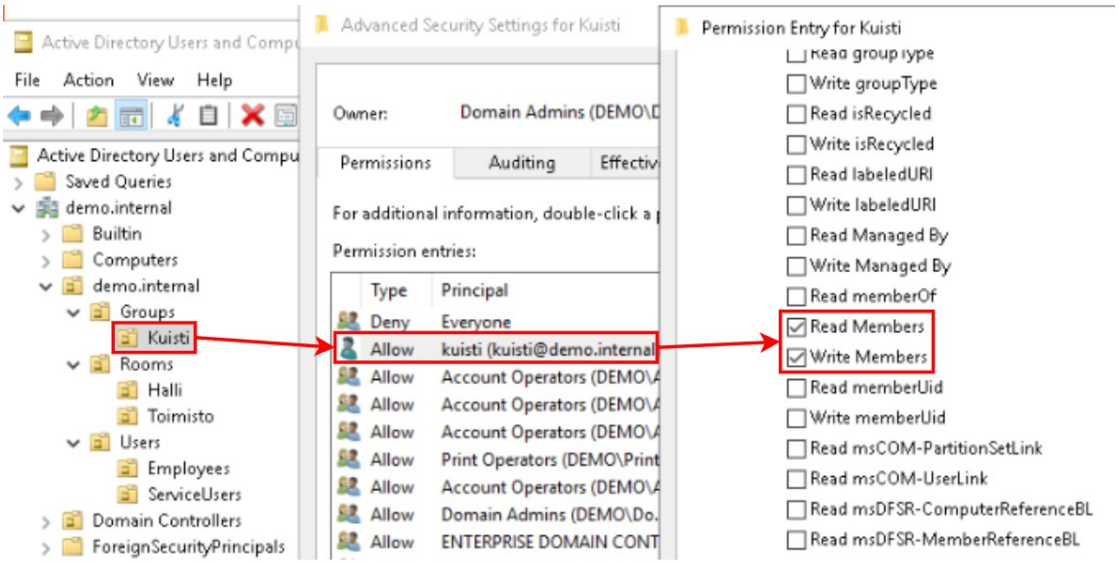
4.5.5 Toimialuepalvelimen konfigurointi

Toimialuepalvelimen konfigurointi aloitettiin roolituksella, jonka yhteydessä palvelimelle määritettiin taulukon yksi mukainen DNS-nimi ja IP-osoite. Roolituksen aikana palvelimelle asennettiin AD DS-, AD CS- ja DNS-roolit, joita tarvittiin AD-hakemistopalvelimen ja LDAPS-yhteyden käyttöönottoa varten. Kaikki roolitukset tehtiin oletusarvojen mukaisesti, ja toimialueen nimeksi määritettiin demo.internal.

Roolituksen jälkeen toimialuepalvelimelle luotiin testiskenaariota varten työntekijän käyttäjätili, Kuistin palvelukäyttäjätili, tarvittavat AD-ryhmät ja OU-rakenne (Organizational Unit), minkä jälkeen topologiassa olleet työasemat WS01 ja WS02 liitettiin toimialueeseen. Kuvan 15 OU-rakenteessa kaikki OU:t olivat vakioasetuksin luotuja, paitsi Kuisti-niminen OU, jonka asetuksia muutettiin Kuistin toimintaa varten. Kuvassa 16 on esitettynä Kuisti-nimiseen OU:hun tehdyt muutokset.



Kuva 15. Toimialuepalvelimen OU-rakenne



Kuva 16. Kuisti-nimiselle OU:lle tehdyt muutokset

Kuvassa 15 on esitettynä, miten testiympäristön käyttäjä- ja konetilit sekä ryhmät sijoittuivat OU-rakenteeseen. Rooms- ja Users-nimisten OU:iden alaiset OU:t sisälsivät topologian työasemiin linkitetyt konetilit sekä työntekijän normaalin käyttäjätilin että Kuistia varten luodun palvelukäyttäjätilin. Työntekijän käyttäjätiliä tarvittiin työasemille kirjautumista varten, ja palvelukäyttäjätilin avulla Kuisti-palvelin pystyi hallinnoimaan huonekohtaisten AD-ryhmien jäsenlistoja LDAP-protokollaa käyttäen. Ryhmien jäsenlistojen hallinnointi Kuisti-palvelimelta käsin vaati kuitenkin Kuisti-nimisen OU:n pääsyoikeuksien muokkausta LDAP-hallintayhteyttä varten. Tässä tapauksessa kuisti-palvelukäyttäjälle annettiin oikeudet lukea ja kirjoittaa Kuisti-nimisen OU:n alaisten AD-ryhmien jäsenlistoja, kuten kuvasta 16 voidaan havaita.

Kuvassa 15 Kuistiin liittyneet huonekohtaiset AD-ryhmät ovat tunnistettavissa etuliitteestä KuistiRoom_, ja roolikohtaiset AD-ryhmät etuliitteestä KuistiRole_. Huonekohtaisilla AD-ryhmillä hallittiin työasemakirjautumisia, ja roolikohtaisilla ryhmillä tarkastettiin, mitkä suodatussäännöt AD-käyttäjille tulee luoda työasemalle kirjautumisen yhteydessä (liite yksi). Testiskenaarion tapauksessa työntekijälle tarkoitettu käyttäjätili oli KuistiRole_Tyontekija-ryhmän jäsen, jolloin työntekijän rooliin linkitetyt suodatussäännöt luodaan palomuurille työasemalle kirjautumisen yhteydessä. Kuisti-järjestelmä käytti huonekohtaisia ryhmiä myös tunnistaakseen kaikki ensimmäiseen autentikointimetodiin liittyneet huoneet, jos autentikoinnin ensimmäiseksi metodiksi oli määritetty huoneista koostunut polku. Kuvassa yhdeksän esitetyssä testiskenaariossa ensimmäisinä autentikointimetodeina toimi aulasta ja työhuoneista koostuneet polut, joten toimialuepalvelimelle lisättiin kaikki työasemia sisältäneiden huoneiden lisäksi aulaan linkitetty AD-ryhmä.

4.5.6 Ryhmäkäytännöt

Kuisti-järjestelmä hallinnoi työasemille kirjautumisia ryhmäkäytäntöjen ja edellä mainittujen huonekohtaisten AD-ryhmien avulla. Ryhmäkäytäntöjen ja AD-ryhmien yhdistelmällä työasemille kirjautuminen kyettiin sallimaan vain huonekohtaisiin ryhmiin kuuluneille jäsenille, jolloin vain kiinteistöön ja huoneeseen tunnistautuneet identiteetit kykenivät kirjautumaan työasemille, kuten liitteen yksi vuokaaviossa on esitetty. Huonekohtaisen hallinnan takaamiseksi jokaiselle työasemia sisältävälle tilalle oli luotava oma Kuisti_LocalLogin-ryhmäkäytäntö. Testiskenaariossa käytetyt Kuisti_LocalLogin-ryhmäkäytännöt ovat esitettynä liiitteessä kaksi.

Liitteen kaksi huonekohtaiset ryhmäkäytännöt sallivat työasemille kirjautumisen vain järjestelmänvalvojien ryhmiin tai huonekohtaiseen ryhmään kuuluneille käyttäjille. Ryhmäkäytännöissä paikallinen kirjautuminen sallittiin myös järjestelmänvalvojille, jotta työasemalle kirjautuminen olisi mahdollista myös mahdollisissa vika- ja vianselvitystilanteissa. Kummatkin liitteessä kaksi esitetyistä Kuisti_LocalLogin-ryhmäkäytännöistä olivat huonekohtaista ryhmää lukuun ottamatta asetuksiltaan täysin identtisiä.

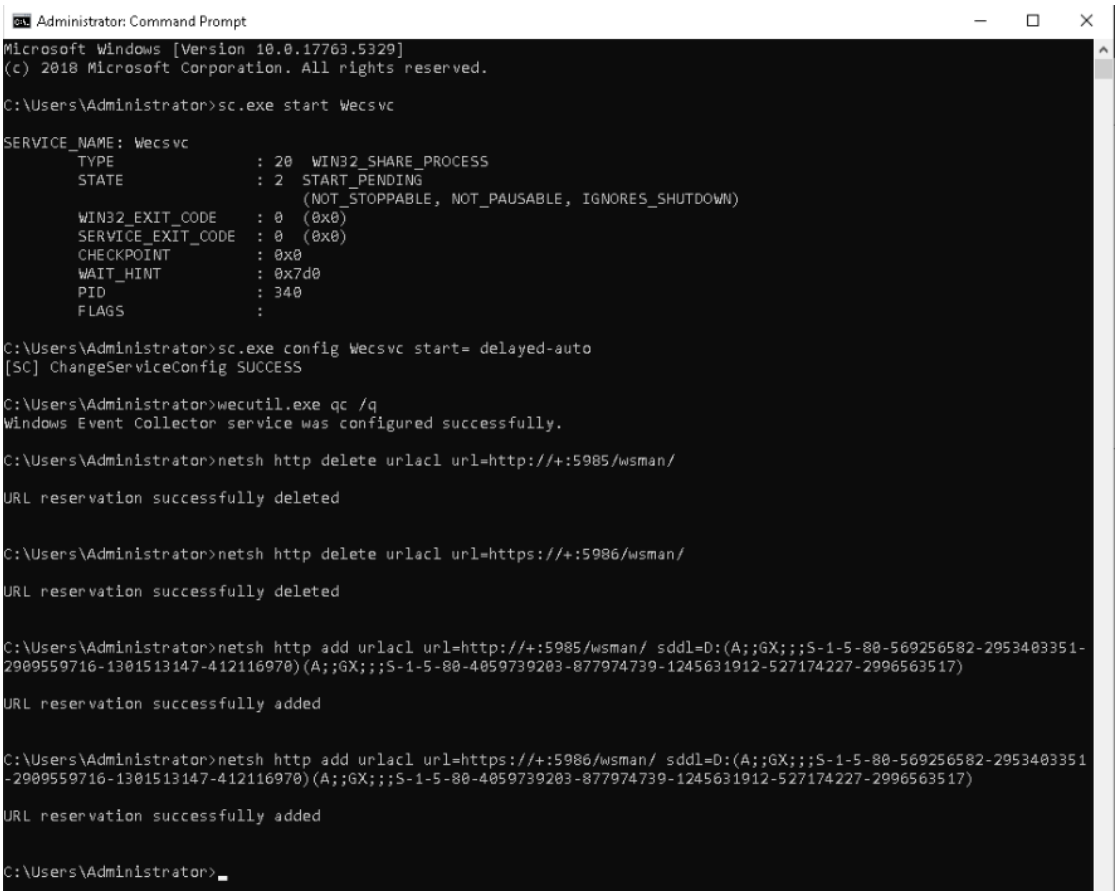
Edellä mainittujen muutosten lisäksi toimialuepalvelimelle luotiin Kuisti_SecuritySettings- ja Kuisti_Screensaver-ryhmäkäytännöt, jotka ovat esitettynä liitteessä kaksi. Kuisti_SecuritySettings-ryhmäkäytännön tehtävänä oli pakottaa työasemat kommunikoimaan toimialuepalvelimen kanssa kirjautumisen yhteydessä. Muutoin työasemien välimuistiin talletettuja tunnuksia olisi voinut käyttää työasemalle kirjautumiseen, jos toimialuepalvelin ei olisi ollut saatavilla. Tämä olisi mahdollistanut käyttäjätunnusten käytön työasemilla ilman ryhmäjäsenyyksien tarkastamista, sillä käyttäjätunnusten ryhmäjäsenyydet tarkastetaan toimialuepalvelimelta ainoastaan Kerberos-autentikoinnin TGT-tiketin luonnin yhteydessä, jota ei luoda, jos työasemalle kirjaudutaan välimuistissa olleilla tunnuksilla (Microsoft 2024c). Tällöin ryhmäjäsenyyksien tarkastamatta jättäminen olisi mahdollistanut käyttäjätunnusten käytön työasemilla, vaikka ne eivät olisi huonekohtaisten ryhmien jäseniä.

Liitteen kaksi Kuisti_Screensaver-ryhmäkäytännön tarkoituksena oli asettaa käyttäjille oletusarvoinen näytönsäästäjä, jotta Kuisti_SecuritySettings-ryhmäkäytännön Machine Inactivity Limit -asetus toimisi halutulla tavalla. Kyseinen asetus aktivoi ryhmäkäytäntöön määritetyn ajanjakson jälkeen näytönsäästäjän, jolloin käyttäjän sessio lukkiutuu automaattisesti. Automaattisen lukituksen tunnistusta tarvittiin, sillä työaseman lukitukseen liittyvästä tapahtumasta ei saada selville, onko lukitus tapahtunut käyttäjän vai automatiikan toimesta. Näytönsäästäjän avulla manuaalinen ja automaattinen lukitus voidaan erottaa toisistaan. (Windows Security Log Event ID 4800 s.a.; Windows Security Log Event ID 4802 s.a.) Lukituksen tyypin tunnistus oli Kuistin toiminnan kannalta tärkeää. Manuaalisen lukituksen kautta käyttäjä kykenee indikoimaan Kuistille palaavansa työasemallensa myöhemmin, mikä sisältyy liitteessä yksi esitettyyn paluuliikenteen hallinnointiin.

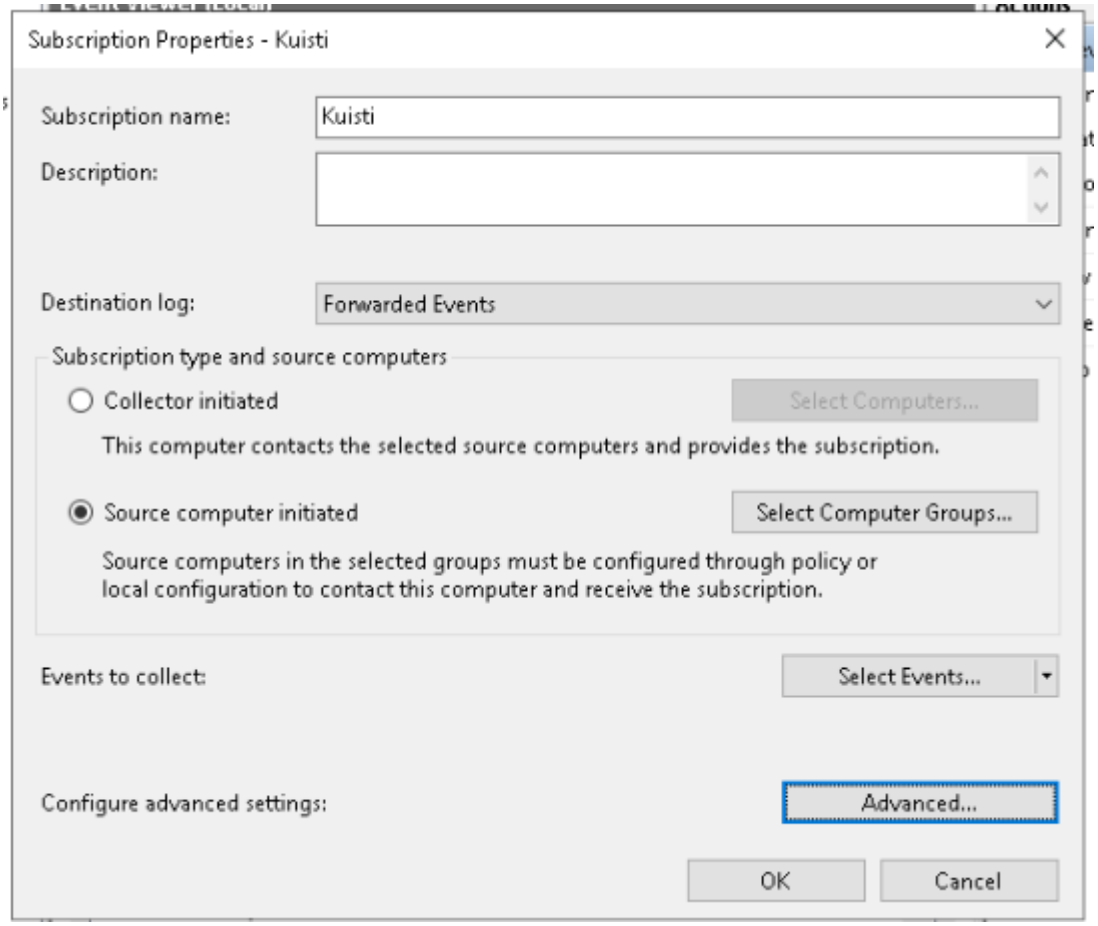
4.5.7 WEC-palvelimen asennus ja konfigurointi

Testiympäristön lokitusjärjestelmänä toimi WEC-palvelin, joka asennettiin ja konfiguroitiin toimialuepalvelimelle kuvan 17 mukaisin komennoin. WEC-palvelimelle luotiin asennuksen ja konfiguroinnin jälkeen kuvassa 18 esitetty tapahtumatilaus (Event Subscription), jonka avulla toimialueen työasemat määritettiin lähettämään vain sisään- ja uloskirjautumisiin sekä lukituksiin liittyvät tapahtumat toimialuepalvelimelle (tapahtumatunnisteet 4624, 4647, 4800,

4801 ja 4802). Tapahtumatilauksen kohderyhmäksi valittiin kuvassa 15 esitetty WEFComputers-ryhmä, jonka jäseninä olivat topologian työasemat WS01 ja WS02.

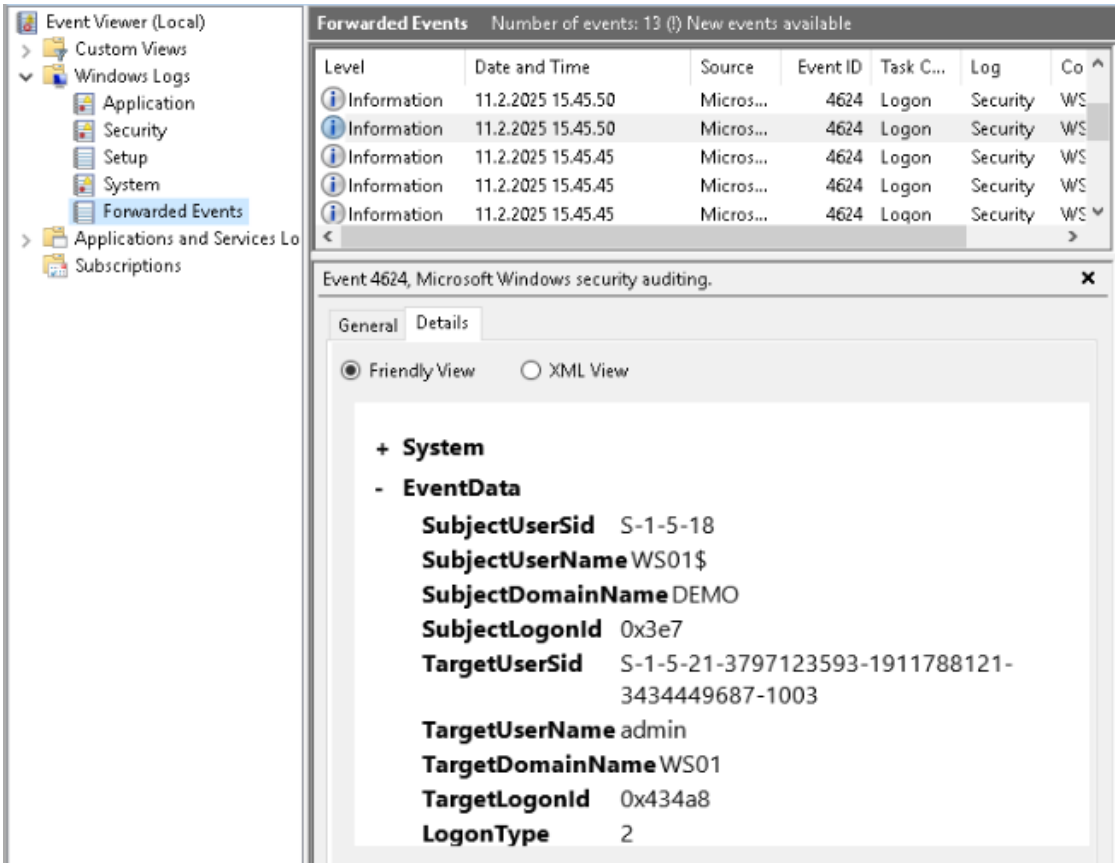


Kuva 17. Kuvankaappaus WEC-palvelimen asennukseen käytetyistä komennoista



Kuva 18. Kuvankaappaus Kuistia varten luodusta tapahtumatilauksesta

Kuvissa 17 ja 18 näytetyt komennot ja konfiguraatiot mahdollistivat lokitietojen vastaanoton WEFComputers-ryhmään kuuluneilta työasemilta, mitkä ohjautuivat toimialuepalvelimen Forwarded Events -lokiin käsittelyä varten (kuva 19). Kuvan 18 tapahtumatilauksen jakelu työasemille hoidettiin liitteen kaksi Kuisti_WEF-ryhmäkäytännöllä, joka käynnisti lokien keruuseen ja lähetykseen tarvittavan WinRM-palvelun, ja määritti työasemille vastaanottavan WEC-palvelimen URL-osoitteen. Edellä mainittujen muutosten lisäksi toimialuepalvelimella suoritettiin komennot wecutil ss "Kuisti" /cm:Custom ja wecutil ss "Kuisti" /dmi:1, jotta lokitietojen lähetys olisi ollut mahdollisimman reaaliaikaista. Komentojen suorituksen jälkeen Wecsvc-palvelu tuli käynnistää uudelleen.



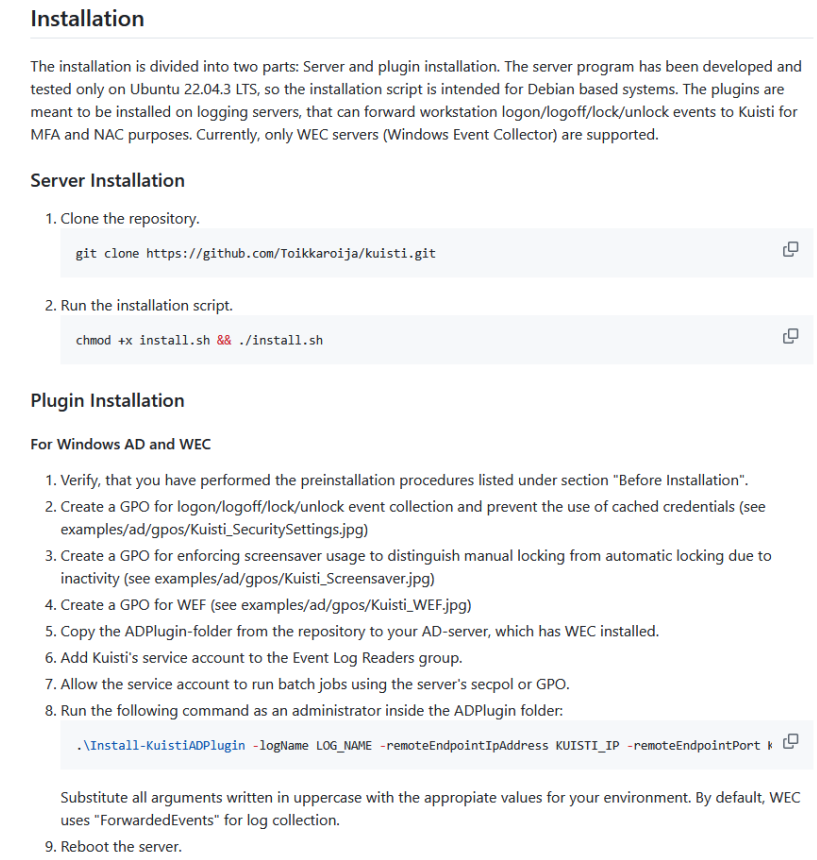
Kuva 19. Kuvankaappaus vastaanotetuista lokimerkinnöistä WEC-palvelimella

Kuva 19 on kuvankaappaus toimialuepalvelimen Forwarded Events -lokista, johon työasemien lokit ohjattiin. Kuvasta 19 voidaan havaita, että WEC-palvelimen asennus oli onnistunut, sillä työaseman WS01 paikallisen järjestelmänvalvojan sisäänkirjautumiseen liittyneet lokimerkinnät vastaanotettiin onnistuneesti asennuksen jälkeen. Nämä lokimerkinnät ohjattiin Kuisti-palvelimelle verkkotason pääsynhallintaa varten PowerShell-skriptikielellä toimineella AD-Plugin-lisäosalla. ADPlugin-lisäosan asennusta käsitellään Kuisti-järjestelmän asennuksen yhteydessä.

4.5.8 Kuisti-järjestelmän asennus

Kuisti-järjestelmä asennettiin taulukossa yksi mainitulle, KUISTI-nimiselle palvelimelle, minkä jälkeen toimialuepalvelimelle asennettiin Kuistin ADPlugin-lisäosa lokien lähetystä varten. Järjestelmäkomponenttien asennusten helpottamiseksi Kuisti-järjestelmän lähdekoodi ja asennusohjeistus ladattiin Github-palveluun, jotta järjestelmää voisi kokeilla vapaasti myös muissa ympäristöissä testitulosten toistamiseksi. Kuvassa 20 on esitettynä Github-palveluun

luotu asennusohje, jota käytettiin palvelinohjelman ja ADPlugin-lisäosan asennuksessa.



Installation

The installation is divided into two parts: Server and plugin installation. The server program has been developed and tested only on Ubuntu 22.04.3 LTS, so the installation script is intended for Debian based systems. The plugins are meant to be installed on logging servers, that can forward workstation logon/logoff/lock/unlock events to Kuisti for MFA and NAC purposes. Currently, only WEC servers (Windows Event Collector) are supported.

Server Installation

1. Clone the repository.

```
git clone https://github.com/Toikkaroija/kuisti.git
```

2. Run the installation script.

```
chmod +x install.sh && ./install.sh
```

Plugin Installation

For Windows AD and WEC

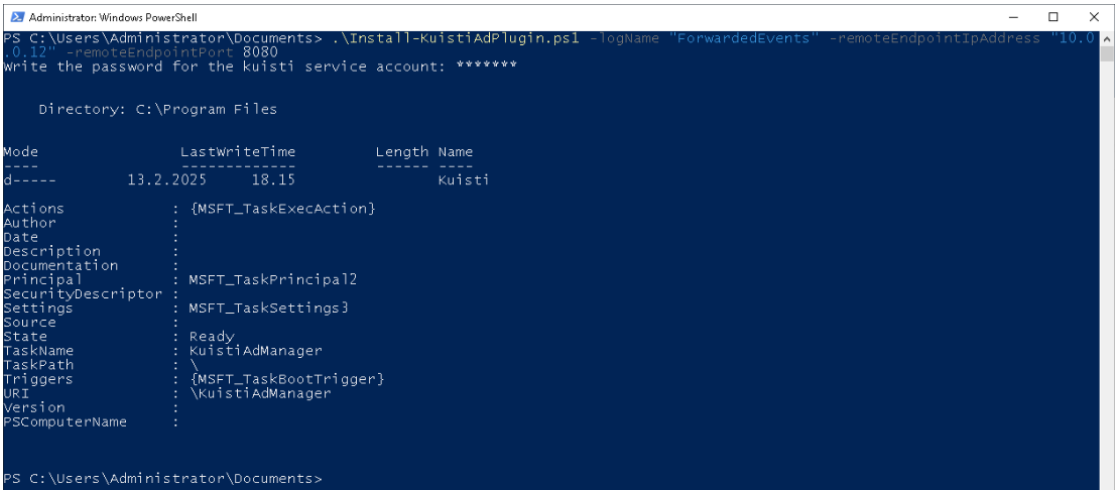
1. Verify, that you have performed the preinstallation procedures listed under section "Before Installation".
2. Create a GPO for logon/logoff/lock/unlock event collection and prevent the use of cached credentials (see examples/ad/gpos/Kuisti_SecuritySettings.jpg)
3. Create a GPO for enforcing screensaver usage to distinguish manual locking from automatic locking due to inactivity (see examples/ad/gpos/Kuisti_Screensaver.jpg)
4. Create a GPO for WEF (see examples/ad/gpos/Kuisti_WEF.jpg)
5. Copy the ADPlugin-folder from the repository to your AD-server, which has WEC installed.
6. Add Kuisti's service account to the Event Log Readers group.
7. Allow the service account to run batch jobs using the server's secpol or GPO.
8. Run the following command as an administrator inside the ADPlugin folder:

```
.\Install-KuistiADPlugin -logName LOG_NAME -remoteEndpointIpAddress KUISTI_IP -remoteEndpointPort k
```

   Substitute all arguments written in uppercase with the appropiate values for your environment. By default, WEC uses "ForwardedEvents" for log collection.

9. Reboot the server.

Kuva 20. Kuvankaappaus Kuisti-järjestelmän komponenttien asennukseen käytetystä ohjeistuksesta (Kuisti 2025)



Kuva 21. Kuvankaappaus ADPlugin-lisäosan asennuksesta toimialuepalvelimelle

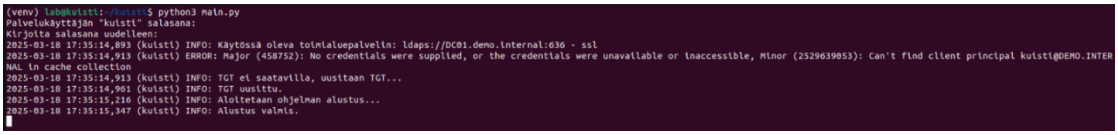
Kuisti-palvelimen asennus aloitettiin kloonaamalla Kuistin lähdekoodi komennolla git clone https://github.com/Toikkaroija/kuisti.git, minkä jälkeen varsinai-

nen asennus suoritettiin komennolla chmod +x install.sh && ./install.sh. Asennusskripti asensi kaikki Kuisti-palvelimen ajoon tarvitut ohjelmistopaketit. Asennuksen jälkeen ohjelma oli valmis konfigurointia varten, jota käsitellään seuraavassa osiossa. Kuistiin liittyvän palvelinohjelman asennuksen jälkeen toimialuepalvelimelle asennettiin Kuistin ADPlugin-lisäosa kuvan 20 ohjeistusten mukaisesti. ADPlugin-lisäosan asennukseen tarkoitettu skripti lisäsi palvelimelle ajoitetun tehtävän, joka suorittaa palvelimen käynnistyessä ADPlugin.ps1-skirptin lukeakseen ja lähettääkseen työasemilta kerättyjä lokimerkintöjä Kuisti-palvelimelle (kuva 21).

4.5.9 Kuisti-palvelimen konfigurointi

Kuisti-palvelimen asennuksen yhteydessä palvelimelle luotiin automaattisesti kuisti/confs-kansio, joka sisälsi esimerkkikonfiguraatiotiedostoja konfiguroinnin helpottamiseksi. Kuisti-järjestelmään liittyneet, kuisti/confs-polussa sijainneet esimerkkikonfiguraatiot olivat jaettu kolmeen eri osioon: ympäristöön, suodatussääntöihin ja lokien käsittelyyn. Ympäristöön liittynyt konfiguraatiotiedosto environment_template.json sisälsi Kuistin toimintaympäristöön liittyneet asetukset, kuten verkkoympäristön, palomuurien ja huoneiden asetukset, joita Kuisti tarvitsi toimialuepalvelimen ja palomuurin kanssa kommunikointiin sekä huonekohtaisiin määrityksiin. Filtersets_template.json-konfiguraatiotiedosto sisälsi käyttäjien roolikohtaiset suodatussäännöt, joilla hallittiin käyttäjien verkkotason pääsyä eri resursseihin työasemakirjautumisten yhteydessä. Lokien käsittelyyn liittynyt konfiguraatiotiedosto log_detection_template.json liittyi LogHandler-komponentin toimintaan, mitä se käytti tekstipohjaisten lokimerkintöjen käsittelyyn. Lokien käsittelyyn tarkoitettua konfiguraatiotiedostoa ei tarvita, jos järjestelmän ylläpitäjä luo oman LogHandler-komponentin lokien käsittelyä varten.

Testiskenaarion aikana käytetyt konfiguraatiotiedostot ovat nähtävissä liitteessä kolme. Konfiguraatioiden luontiin käytettiin edellä mainittuja esimerkkikonfiguraatioita ja liitteen neljä referenssitaulukkoja, jotka toimivat tukimateriaalina konfiguraatioiden luontiin. Konfiguraatiotiedostojen luonnin jälkeen konfiguraatiot siirrettiin ohjelman juurihakemistoon ja Kuisti-järjestelmä käynnistettiin komennolla python3 main.py (kuva 22).

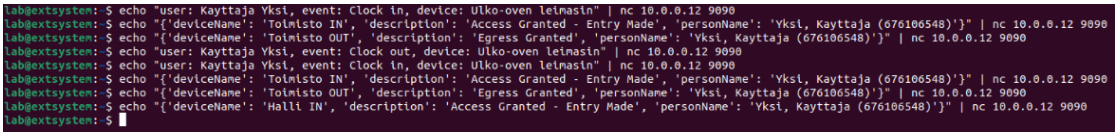


Kuva 22. Kuisti-järjestelmän käynnistys

Kuvasta 22 voidaan havaita, että Kuistin asennus ja konfigurointi oli onnistunut. Kuistin ensimmäisen käynnistyksen yhteydessä järjestelmä kysyi kuisti-palvelukäyttäjän salasanaa TGT-tiketin hakua varten, jota palvelin tarvitsi hakemistopalvelimen kanssa kommunikointiin. Tämän jälkeen Kuisti suoritti liitteessä yksi esitetyn käynnistysprosessinsa onnistuneesti, minkä jälkeen ympäristö oli valmis testin suoritusta varten.

4.5.10 Testin suoritus

Testiskenaario suoritettiin kahdesti kuvassa yhdeksän esitetyn aikataulun mukaisesti 19.3.2025 ja 31.3.2025. Testien aikana verkkotason pääsynhallintaa seurattiin OPNsense-palomuurin hallintapaneelista ja Kuisti-palvelimen terminaalista käsin. Testien aikana tarkastettiin myös monivaiheisen tunnistautumisen toimivuus yrittämällä kirjautua sisään työasemille ilman simuloitua huoneeseen siirtymistä. Ulkoisia järjestelmiä simuloitiin lähettämällä EXTSYSTEM-palvelimelta työajanseurannan ja kulunvalvonnan simuloituja lokimerkintöjä Kuisti-palvelimelle kuvan 23 mukaisin komennoin.



Kuva 23. Kuvankaappaus simuloiduista lokimerkinnöistä

Kuvassa 23 näkyy kaikki testien aikana käytetyt, simuloidut lokimerkinnät, jotka lähetettiin Kuisti-palvelimelle. Lyhyemmät Clock in- ja Clock out -merkkijonot sisältävät lokimerkinnät simuloivat työajanseurannan leimasinlaitetta, ja pidemmät lokimerkinnät simuloivat puolestaan kulunvalvontajärjestelmää. Lokimerkinnät lähetettiin kuvan yhdeksän aikataulun mukaisesti mahdollisimman realististen tulosten takaamiseksi. Kuvan neljäs ja viides työajanseurannan lokimerkintä simuloi käyttäjän pitämää, tunnin mittaista taukoa klo 10.30–11.30, ja muut lokimerkinnät simuloivat siirtymistä työtilojen välillä. Testien päätyttyä, Kuistin generoimat lokitiedostot kuisti.log ja ext_system.log kerättiin talteen

testitulosten analysointia varten. Myös kuvan 10 testiskriptin generoimat net-test.log-tiedostot kerättiin talteen WS01- ja WS02-työasemilta.

5 TULOKSET

Tutkimuksen primääriaineiston analyysin avulla oli tarkoitus selvittää, soveltuiko tutkimuksen aikana kehitetty Kuisti-järjestelmä hyödyntämään fyysisiä tunnisteita ja elementtejä osana monivaiheista autentikointiprosessia, ja miten niiden hyödyntäminen muutti kohteena olleen sisäisen tietoverkon turvallisuutta yhdessä dynaamisten suodatussääntöjen kanssa. Primääriaineiston analyysi pyrki vastaamaan vain kolmanteen tutkimuskysymykseen. Ensimmäiseen ja toiseen tutkimuskysymykseen on pyritty vastaamaan teoriaosuuden aikana monivaiheisen tunnistautumisen prosessin ja siihen liittyvien ongelmien tunnistamiseksi. Näiden alustavien kysymysten vastaukset antoivat pohjan tutkimuksen suoritukselle ja kolmannen tutkimuskysymyksen esittämiselle, sillä ongelmaa ei olisi voinut lähteä ratkaisemaan, jos autentikointiprosessia ja siihen liittyviä ongelmia ei olisi voitu tunnistaa.

5.1 Monivaiheisen tunnistautumisen prosessi

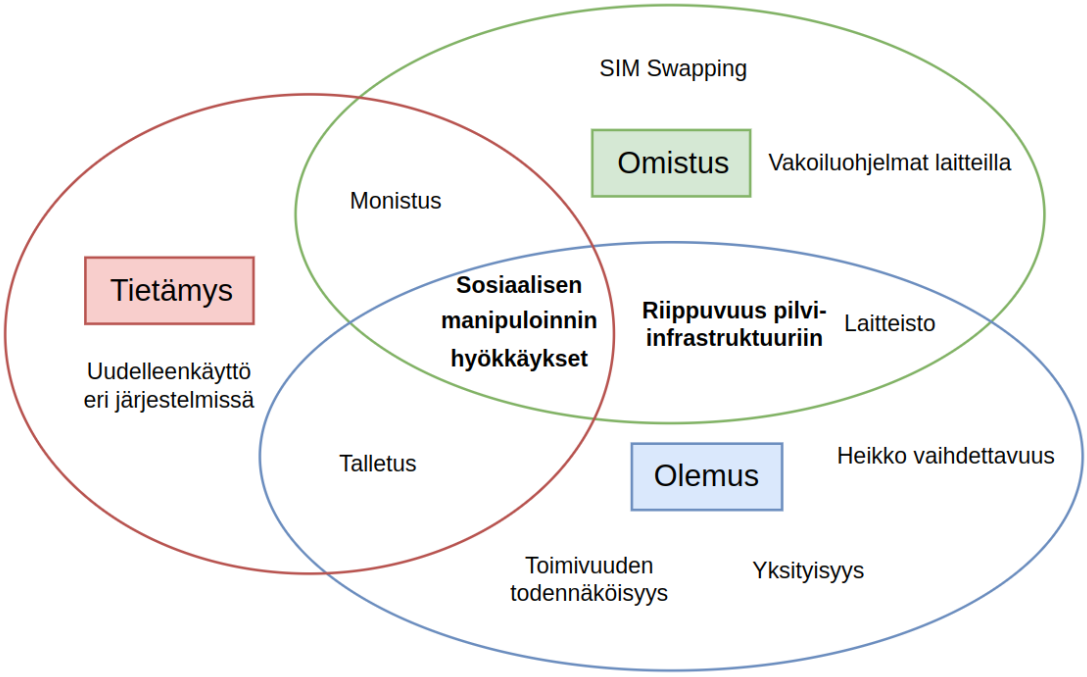
Ensimmäisessä tutkimuskysymyksessä pyrittiin selvittämään monivaiheisen autentikointiprosessin kriteerit ja tunnusmerkit, jotta autentikointiprosessin muokkaus olisi mahdollista. Kuten aiemmin on mainittu, monivaiheisessa tunnistautumisessa identiteetti pyritään varmentamaan kahta tai useampaa autentikaattoria käyttäen. Autentikaattorit ovat uniikkeja, ja ne sisältävät aina julkisen osan, josta autentikaattorit tunnistetaan. Autentikaattorit yleensä sisältävät myös salaisen osan, jonka vain tunnistettava identiteetti ja autentikoiva järjestelmä tietävät. Järjestelmä voi näin ollen pyytää identiteetiltä halutun autentikaattorin julkista osaa (käyttäjätunnus), jolloin identiteetti pystyy antamaan julkisen osan ja siihen linkitetyn salaisuuden (salasana) järjestelmälle, jolloin järjestelmä voi todeta identiteetin tunnistetuksi, jos identiteetin antama salainen osa vastaa järjestelmään talletettua salaista osaa.

Ensimmäiseen tutkimuskysymykseen vastatessa myös pelkän julkisen osan sisältäneiden autentikaattoreiden todettiin olleen käyttökelpoisia monivaiheisessa autentikointiprosessissa. Pelkän julkisen osan sisältäviä autentikaattoreita tulee käyttää toisten, salaisen osan sisältävien autentikaattoreiden

kanssa, kuten tutkimuksessa on aiemmin mainittu. Pelkän julkisen osan sisältäviä autentikaattoreita, kuten biometrisiä tunnistetietoja, hyödynnetään tunnistautumisen sijaan identiteettien tunnistuksessa.

5.2 Perinteisten autentikaattorien ongelmat

Toiseen tutkimuskysymykseen vastatessa perinteisten autentikaattorien suurimmaksi ongelmaksi tunnistettiin sosiaalisen manipuloinnin hyökkäykset, kuten kuvan 24 diagrammista voidaan havaita. Sosiaalisen manipuloinnin hyökkäykset juurtuvat inhimillisiin virheisiin autentikaattorien käytössä, ja virheiden poistamiseksi on kehitetty erilaisia implisiittisiä autentikointimetodeja, jolloin autentikointi tapahtuu käyttäjälle näkymättömällä tavalla. Tällöin käyttäjän tekemät, eksplitiittiset toimet voidaan poistaa autentikointiprosessista, jolloin inhimillisiä virheitä voidaan vähentää ja ehkäistä.



Kuva 24. Venn-diagrammi autentikaattorityyppien tunnistetuista ongelmista

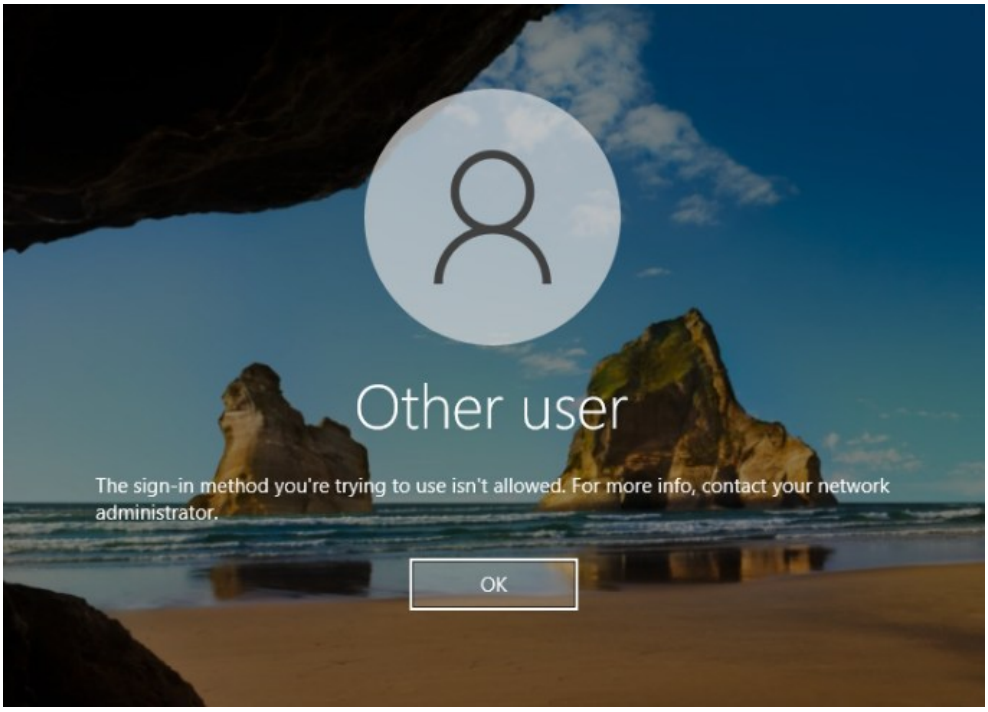
Kuvan 24 diagrammissa on kartoitettuna tutkimuksen aikana havaittuja ongelmia eri autentikaattorityyppeihin liittyen. Diagrammista voidaan havaita, että kaikki autentikaattorityypit ovat alttiita sosiaalisen manipuloinnin hyökkäyksille, joita on pyritty estämään implisiittisin metodein tutkimuksessa aiemmin mainituin tavoin. Diagrammista on myös nähtävissä, että olemukseen ja omistukseen pohjautuvat autentikaattorit ovat riippuvaisia pilvi-infrastruktuurista, sillä

nämä autentikaattorityypit käyttävät kolmansien osapuolten palveluja autentikointiprosessin aikana. Tämä riippuvuus voi lamauttaa autentikointiprosessin suorituksen, jos yhteys pilvi-infrastruktuuriin katkeaa, tai jos pilvi-infrastruktuurissa tapahtuu joko tahaton tai tahallinen häiriö. Nämä kaksi ongelmakohtaa olivat tutkimusongelman keskipisteessä, ja näihin ongelmiin pyrittiin saamaan ratkaisu kolmannen tutkimuskysymyksen vastauksen kautta.

Kuten aiemmin mainittiin, inhimillisiä virheitä on pyritty poistamaan implisiittisin autentikointimetodein. Inhimillisiä virheitä ehkäisevien, implisiittisten autentikointimetodien kehityksen kuitenkin havaittiin keskittyneen biometrisiin menetelmiin. Biometristen menetelmien toiminta perustuu todennäköisyyteen, jolloin autentikointimetodi voi virheellisesti joko evätä tai sallia identiteetin pääsyn resursseihin, jolloin biometriset todennusmenetelmät ovat haavoittuvaisia virheellisille autentikoinneille. Tämä avasi mahdollisuuden tutkia deterministisin tavoin toimineiden menetelmien käyttöä implisiittisessä autentikoinnissa, jota Kuisti-järjestelmällä yritettiin saavuttaa fyysisten elementtien ja tunnisteiden avulla. Kolmas tutkimuskysymys pyrki vastaamaan siihen, miten fyysisiä tunnisteita ja elementtejä voitaisiin hyödyntää implisiittisessä autentikoinnissa ja tietoverkon turvallisuuden parantamisessa. Seuraavassa alaluvussa käsitellään kolmanteen tutkimuskysymykseen saatuja tuloksia ja vastauksia.

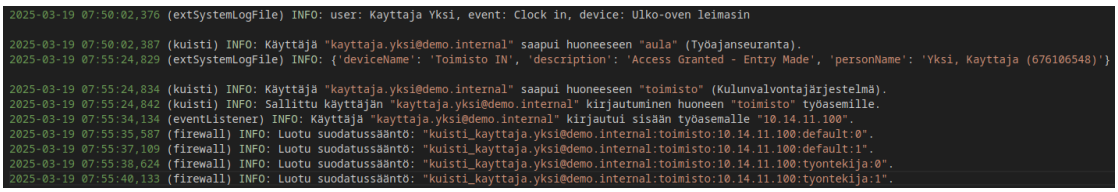
5.3 Fyysisten elementtien hyödyntäminen autentikointiprosessissa

Fyysisten elementtien soveltuvuutta monivaiheisessa autentikoinnissa analysoitiin tarkastelemalla työasemakirjautumisen toimintaa testiskenaariossa ennen ja jälkeen ennalta määrätyn reitin kulkua. Testiskenaarion ensimmäisen autentikointimetodin läpäisemiseksi käyttäjän oli kuljettava aulan kautta työhuoneeseen kirjautuakseen sisään työasemalle WS01. Kirjautumisen estoa kokeiltiin kirjautumalla sisään työasemalle WS01 ilman simuloidun reitin kulkemista (kuva 25).



Kuva 25. Työaseman WS01 esittämä virheilmoitus kirjautumisikkunassa

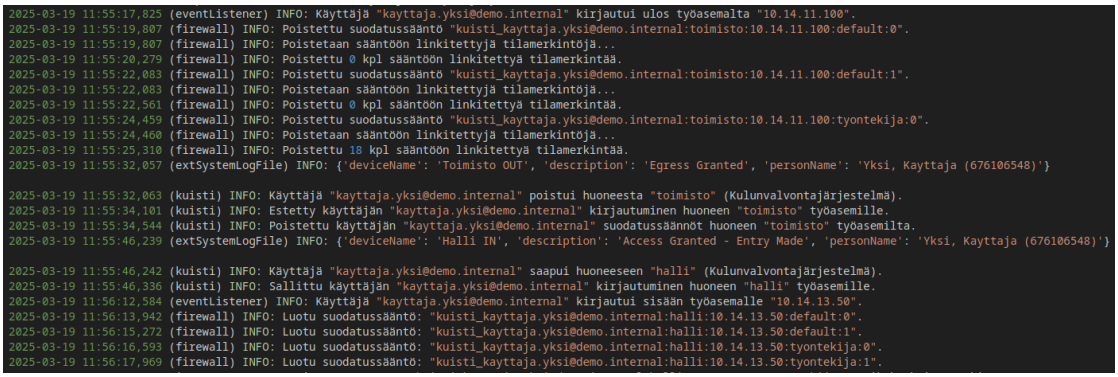
Kuvassa 25 esitetty virhe ilmeni, kun käyttäjä yritti kirjautua sisään työasemalleen kulkematta ennalta määrättyä reittiä. Virheen vuoksi käyttäjä ei kyennyt kirjautumaan sisään työasemalle. Havainnon jälkeen käyttäjä kulki testiskenaarion mukaisen reitin toimistoon, jolloin Kuisti-palvelin tulkitsi käyttäjän olleen läsnä toimistossa. Vasta reitin kulkemisen jälkeen käyttäjä pystyi kirjautumaan sisään työasemalleen (kuva 26).



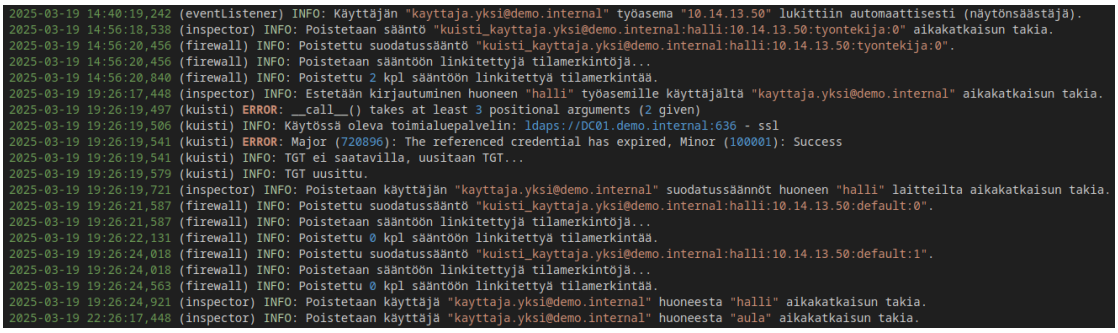
Kuva 26. Kuvankaappaus Kuisti-palvelimen lokista, kun käyttäjän kirjautuminen sallittiin toimiston työasemille

Kuvasta 26 voidaan havaita, miten paikallinen kirjautuminen työasemalle sallittiin vasta työajanseurannan ja kulunvalvonnan leimausten jälkeen. Sisäänkirjautuminen sallittiin vasta silloin, kun Kuisti oli todennut käyttäjän kulkeneen ennalta määrätyn reitin toimistoon työajanseurannan ja kulunvalvontajärjestelmän lähettämien lokimerkintöjen perusteella. Sama efekti oli havaittavissa, kun käyttäjä siirtyi toimistosta hallin puolelle, jolloin toimiston työasemille kir-

jautuminen estettiin, ja hallin työasemille kirjautuminen sallittiin (kuva 27). Työasemakirjautumisten rajauksen havaittiin toimineen oikein myös silloin, kun huoneelle asetettu aikakatkaisu astui voimaan (kuva 28).



Kuva 27. Kuvankaappaus Kuisti-palvelimen lokista, kun käyttäjä siirtyi toimistosta halliin



Kuva 28. Kuvankaappaus Kuisti-palvelimen lokista, kun kirjautuminen hallin työasemille estettiin aikakatkaisun vuoksi

Kuvassa 27 esitetystä lokista nähdään, miten työasemille kirjautumista hallittiin toimistosta halliin siirtymisen aikana. Ensin, käyttäjän poistuessa toimistosta, kaikki toimiston työasemalla olleet suodatussäännöt poistettiin ja työasemille kirjautuminen estettiin. Tämän jälkeen käyttäjä siirtyi hallin puolelle, jolloin kirjautuminen hallin työasemille sallittiin. Tällöin käyttäjä oli läpäissyt hallin työasemille määritetyn ensimmäisen autentikointimetodin. Ensimmäisen autentikointimetodin kriteerit täyttyivät, sillä käyttäjä oli vielä läsnä aulassa, jolloin käyttäjä oli kulkenut määrätyn polun halliin.

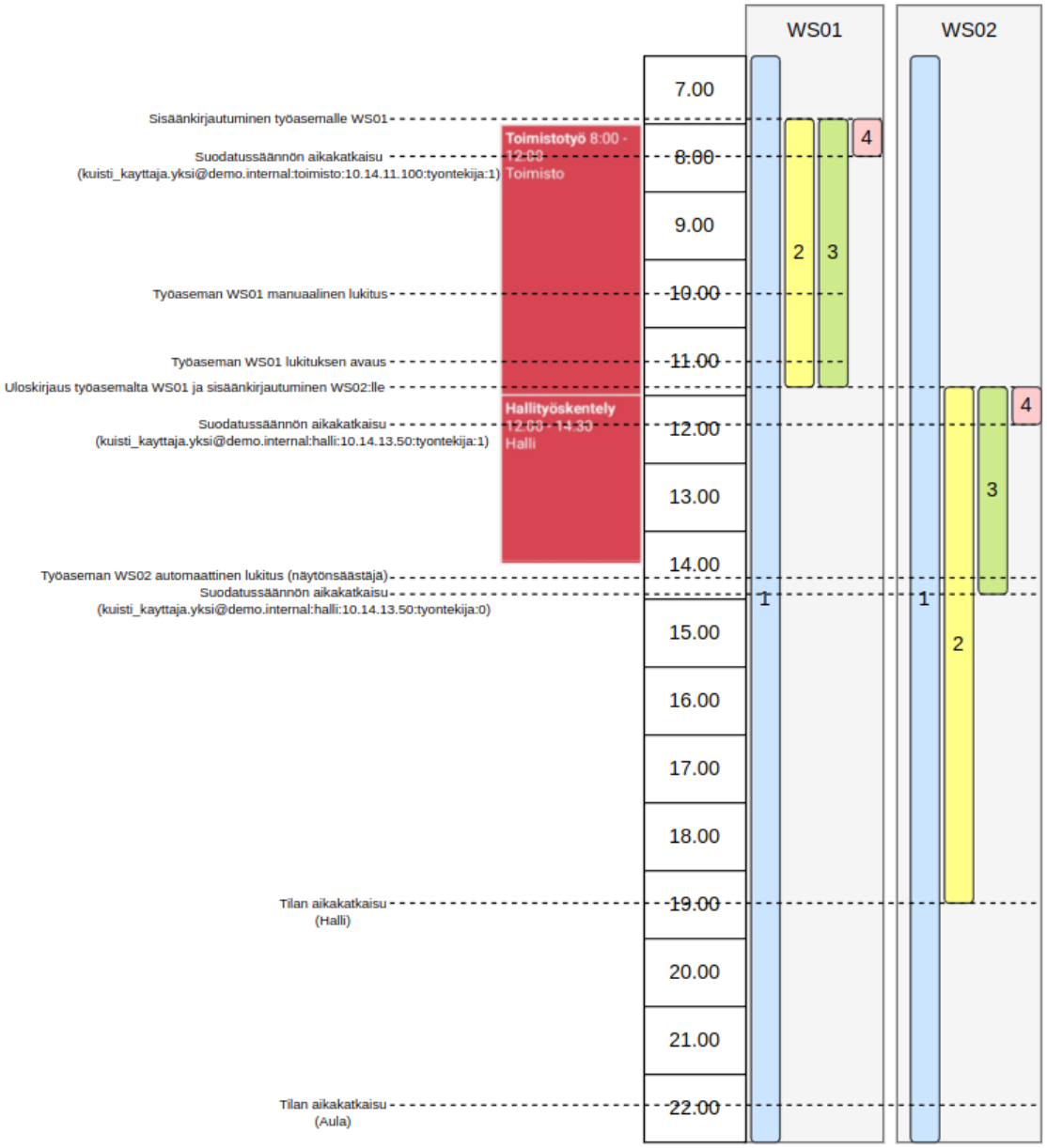
Kuvasta 28 voidaan havaita, miten aikakatkaisu esti hallin työasemille kirjautumisen, kun käyttäjä oli poistunut hallista ilman leimauksia. Käyttäjä poistettiin huonekohtaisesta ryhmästä, jolloin kirjautuminen työasemille estyi ja käyttäjää varten luodut suodatussäännöt poistettiin. Kuistin LDAP-yhteys oli ehtinyt kat-

keta klo 19.26, jolloin lokiin oli kirjattu virheviesti kredentiaalien vanhentumisesta. Kuisti oli virheen havaittuaan luonut uuden LDAP-yhteyden ja hakenut uuden TGT-tiketin toimialuepalvelimelta, viimeistellen käyttäjän ryhmäjäsenyyden poiston onnistuneesti.

Fyysisiin elementteihin ja tunnisteisiin perustuvien järjestelmien lokitietoja voidaan näin ollen hyödyntää kuvan viisi mukaisessa autentikointiprosessissa ensimmäisenä autentikointimetodina. Lokitietojen tulee sisältää riittävän tarkat tiedot identiteetin, alueen ja kulkusuunnan tunnistukseen autentikointipäätöstä varten. Tämä vastaa kolmannen tutkimuskysymyksen ensimmäiseen osaan, jossa tarkoituksena oli selvittää, miten fyysisiä tunnisteita ja elementtejä voitaisiin hyödyntää monivaiheisessa tunnistautumisessa.

5.4 Verkkotason pääsynhallinnan vaikutus tietoverkon turvallisuuteen

Kuisti-järjestelmän rikastaman pääsynhallinnan vaikutusta tietoverkon turvallisuuteen mitattiin testin aikana työasemille asennetuin skriptein. Skriptien antamien tulosteiden perusteella työasemien pääsy testiskenaarion sisä- ja ulkoverkkoon pyrittiin varmentamaan tarkastelemalla työasemien skriptien tulosteista saatavilla olleiden verkkopalveluiden määrää eri mittaushetkinä. Mittaushetkinä toimi Kuisti-palvelimen lokiin merkityt ajankohdat, jolloin käyttäjään linkitettyjä suodatussääntöjä luotiin tai poistettiin joko kirjautumisten tai aikakatkaisujen yhteydessä. Työasemien skriptien ja Kuistin lokimerkintöjen aikaleimojen avulla mittaushetket ja mittaustulokset voitiin yhdistää ja sovittaa kuvan yhdeksän testiskenaarion aikatauluun analyysiä varten. Kuvassa 29 esitettyjen tulosten visualisoinnin tulkintaan liittyvät selitteet ovat listattuna taulukossa kaksi.



Kuva 29. Kuisti-palvelimen lokin ja nettest.ps1-skriptien mittaukset sovitettuna testiskenaarion aikatauluun

Taulukko 2. Selitetaulukko kuvan 28 tulosten tulkintaan

NUMERO	SELITE
1	Hakemistopalvelimen saatavuus
2	ICMP-liikenteen saatavuus sisäisille palvelimille (pl. hakemistopalvelin)
3	Ulkoisten verkkopalveluiden saatavuus
4	Sisäisen tiedostopalvelimen saatavuus

Kuvassa 29 on esitettynä testiskenaarion tapahtumat ja aikataulu, johon on sovitettu Kuistin tapahtumalokiin kirjatut tapahtumat ja suodatussäännöin rajatut saatavuudet eri palveluihin ja verkkoihin eri kellonaikoina. Kuvan 29 oikealla puolella on visualisoitu testiskenaariossa käytettyjen työasemien pääsy sisäisiin ja ulkoisiin palveluihin testiskenaarion edetessä. Taulukossa kaksi on listattuna numeroin ja selittein testin aikana mitatut saatavuusjaksot, joiden perusteella Kuistin ohjaaman pääsynhallinnan toimivuutta tarkasteltiin.

Kuten kuvan 29 visualisoiduista tuloksista voidaan havaita, Kuisti-järjestelmä kykeni rajaamaan verkkotason pääsyä sisäisiin ja ulkoisiin resursseihin liitteen kolme filterset.json-konfiguraation määritysten ja liitteen yksi prosessikaavion mukaisesti. Pääsy hakemistopalvelimelle oli avoinna koko testiskenaarion ajan FW01-palomuurin pohjasäännöstön takia, jotta työasemille kirjautuminen ja ryhmäkäytäntöjen synkronointi oli mahdollista (kuva 13). ICMP-liikenteen ja ulkoisiin palveluihin pääsyn sallineet säännöt myös luotiin ja poistettiin oikein kirjautumistapahtumien ja määritettyjen aikakatkaisujen mukaisesti. Kuvasta 29 voidaan myös havaita, että ulkoisten palveluiden saatavuus ei katkennut työaikana aikakatkaisun myötä, kun käyttäjän sessio ulkoiseen palveluun oli vielä kesken. Tämä noudattaa liitteessä yksi esitettyä Inspector-säikeen toimintaa, missä suodatussäännöt tulisi poistaa vain epäaktiivisuuden tai aikakatkaisun takia. Testiskenaarion aikana kumpikaan työasemista ei ottanut yhteyttä tiedostopalvelimeen, joten tiedostopalvelimelle pääsyn sallinut suodatussääntö poistettiin kummallakin työasemalla heti aikakatkaisun jälkeen.

Kun kuvassa 29 summattuja tuloksia tarkastellaan, voidaan Kuistin todeta rajanneen pääsyä verkotettuihin resursseihin onnistuneesti. Pääsy verkotettuihin resursseihin sallittiin ainoastaan silloin, kun pääsylle oli tarve. Samoin pääsy resursseihin estettiin silloin, kun pääsylle ei todettu enää olleen tarvetta joko uloskirjautumisen tai aikakatkaisun yhteydessä. Tällainen tarpeeseen perustuva rajaus on tärkeää, kun tietoverkon resurssien pääsynhallinnan turvallisuutta tarkastellaan PoLP-periaatteen suunnasta. Käyttäjän tulisi päästä verkkoresursseihin käsiksi ollessaan työasemansa luona, ja yhteydet verkkoresursseihin tulisi katkaista, jos käyttäjän ei enää tarvitse käyttää niitä. PoLP-periaatteeseen pohjautuva mittaus näyttäytyi hyvänä vaihtoehtona myös Zero Trust -arkkitehtuurin näkökulmasta, missä implisiittinen pääsy verkkoresurs-

seihin pyritään estämään, jos pääsylle ei ole tarvetta (Rose ym. 2020). Kuvassa 29 esitettyjen tuloksien voidaan todeta parantaneen testiskenaarion tietoverkon resurssien turvallisuutta pääsynhallinnan osalta, mikä vastaa kolmannen tutkimuskysymyksen jälkimmäiseen osaan.

6 JOHTOPÄÄTÖKSET

Tutkimuksen aikana fyysisiin tunnisteisiin ja elementteihin perustuneita järjestelmiä kyettiin liittämään osaksi monivaiheista tunnistautumista Kuisti-järjestelmän avulla. Tällöin voidaan todeta, että fyysisiä tunnisteita ja elementtejä voidaan hyödyntää monivaiheisessa autentikointiprosessissa, kunhan tunnisteita ja elementtejä käyttävät järjestelmät kykenevät lähettämään Kuistille tai vastaavanlaiselle järjestelmälle tarpeeksi verbaalista dataa autentikointipäätösten tekoa varten. Tutkimuksen aikana ulkoisten järjestelmien lähettämän datan todettiin olleen riittävää autentikointitarkoituksiin, kun data sisälsi vähintään identiteetin nimen tai tunnistetiedon ja fyysisen alueen tunnisteen, jotta tunnistettu identiteetti pystyttäisiin yhdistämään tiettyyn fyysiseen alueeseen autentikointiprosessin ensimmäisessä metodissa. Tuloksista voidaan myös todeta, että autentikointiprosessin ensimmäisenä metodina voidaan käyttää useammasta fyysisestä alueesta koostunutta reittiä, jolloin identiteetin kulkureitti työaseman luokse oli varmistettavissa ennen työasemien käytön sallimista.

Ulkoisia järjestelmiä pystyttiin hyödyntämään monivaiheisen autentikoinnin ohella myös digitaalisessa pääsynhallinnassa. Kuten tutkimuksessa on aiemmin mainittu, soveltuvimmat fyysisiä tunnisteita ja elementtejä käyttävät järjestelmät autentikointiprosessin monivaiheistukseen ovat kulunvalvontaan eli fyysiseen pääsynhallintaan keskittyneitä, sillä kyseisten järjestelmien tuottama data sisältää kaikki tarvittavat tiedot ensimmäistä autentikointimetodia varten. Ja koska Kuistin muokkaamassa autentikointiprosessissa digitaalisiin resursseihin pääsee käsiksi vasta toisen autentikointimetodin jälkeen, on ulkoisista järjestelmistä koostuva, ensimmäinen autentikointimetodi osa digitaalista pääsynhallintaa, sillä kaikki autentikointimetodit on suoritettava ennen resursseihin pääsyä (kuva viisi). Tällä tavoin ulkoisia järjestelmiä kyettiin käyttämään digitaalisessa pääsynhallinnassa estämään auktorisoimattomien identiteettien pääsy verkossa olleisiin resursseihin.

Tutkimustulosten analyysin ja tutkimuskysymyksiin vastaamisen jälkeen tutkimusongelma todettiin osittain ratkaistuksi. Tutkimuksessa pyrittiin ratkaisemaan kaksi monivaiheisen autentikoinnin keskeistä ongelmaa: riippuvuus pilvi-infrastruktuuriin ja sosiaalisen manipuloinnin hyökkäykset. Määritellyn tutkimusongelman ensimmäinen osa saatiin ratkaistua, sillä tutkimus osoitti, että monivaiheinen tunnistautuminen on mahdollista suorittaa paikallisesti toimivalla järjestelmällä, joka ei ole riippuvainen kolmansien osapuolten pilvi-infrastruktuurista. Tutkimusongelman toisena keskeisenä osana pyrittiin poistamaan eksplisiittinen, autentikointiin liittynyt toiminta autentikointiprosessista fyysisten tunnisteiden ja elementtien avulla, mikä todettiin toimivaksi ratkaisuksi. Tämä ei kuitenkaan täysin eliminoi inhimillisistä virheistä johtuvia sosiaalisen manipuloinnin hyökkäyksiä autentikointiprosessista. Tätä ongelmakohtaa käsitellään tarkemmin seuraavassa luvussa.

7 POHDINTA

7.1 Tutkimuksen onnistuminen yleisesti

Tutkimus oli onnistunut, ja se vastasi kaikkiin asetettuihin tutkimuskysymyksiin, vaikka tutkimusongelman ratkaisu jäikin hieman puutteelliseksi. Toteutusvaiheessa paranneltu Kuisti-järjestelmä toimi odotetulla tavalla, ja järjestelmän dokumentaatiota saatiin laajennettua siten, että tulosten toistaminen on mahdollista vastaavanlaisissa testiympäristöissä. Kuisti-järjestelmä kykeni myös osoittamaan, että sisäisen tietoverkon resursseja varten voidaan luoda paikallisesti toimiva, monivaiheinen autentikointiprosessi, jossa identiteetti voidaan varmentaa salasanan lisäksi implisiittisesti fyysisiä tunnisteita tai elementtejä hyödyntäen.

Fyysisten tunnisteiden käyttö implisiittisessä autentikoinnissa ei kuitenkaan täysin poista sosiaalisen manipuloinnin hyökkäysten aiheuttamaa uhkaa. Tutkimuksessa toteutettu implisiittinen autentikointi poisti käyttäjän eksplisiittisen toiminnan autentikointiprosessin ensimmäisestä metodista, mutta käyttäjän oli silti kirjauduttava työasemalle niin sanotuin perinteisin menetelmin käyttäjätunnusta ja salasanaa käyttäen. Kuten tutkimuksessa on aiemmin kerrottu, salasanat ja tietämykseen perustuvat autentikaattorit ovat erittäin alttiita sosiaalisen manipuloinnin hyökkäyksille, jolloin niitä voidaan varastaa mm. kalasteluviestein. Kaapatun salasanan käyttö ei kuitenkaan riitä autentikointiprosessin

läpäisyyn, jos autentikointiprosessi sisältää tutkimuksessa ehdotetun ja toteutetun implisiittisen autentikointimetodin, sillä molemmat metodit on läpäistävä ennen resursseihin pääsyä.

Salasanojen tavoin myös fyysiset tunnisteet voivat olla alttiita sosiaalisen manipuloinnin hyökkäyksille. Tämä johtuu fyysisten tunnisteiden, etenkin RFID-tunnisteiden, yksinkertaisesta toimintaperiaatteesta. Esimerkiksi HID:n (2025) iCLASS-perheeseen kuuluvat kulunvalvonnan leimasinlaitteet käyttävät MIFARE Classic -protokollaa. MIFARE Classic -protokolla käyttää ISO 14443-3 -standardin mukaisia UID-tunnisteita identiteettien todentamisessa, jotka ovat neljän, seitsemän tai kymmenen tavun mittaisia (NXP Semiconductors 2018). Yksinkertaisimmat lukijalaitteet eivät käytä muita tunnistetietoja identiteetin todennuksessa, jolloin UID-tunnisteen kloonaamalla hyökkääjä voi teoriassa läpäistä Kuistin ylläpitämän autentikointiprosessin, jos hyökkääjällä on hallussaan myös käyttäjän salasana. Tällainen hyökkäysmenetelmä on ulkopuoliselle hyökkääjälle kuitenkin suhteellisen hankala toteuttaa käytännössä, sillä hyökkääjän on oltava fyysisesti läsnä kohdeorganisaation tiloissa käyttääkseen kloonattua UID-tunnistetta. Tässä tulee myös huomioida, että edellä mainittu hyökkäysmenetelmä toimii vain yksinkertaisimpien pääsynhallintajärjestelmien tunnisteilla, joten hyökkäysmenetelmä ei toimi, jos järjestelmä tunnistaa identiteetin monimutkaisemmalla tai turvallisemmalla tavalla.

Yhtenä turvallisempana tapana voidaan ehdottaa passkey-avainten käyttöä. Passkey-avaimet voitaisiin sisällyttää suoraan fyysisiin tunnisteisiin, jolloin kloonattuja tunnisteita ei pystyttäisi käyttämään autentikointiprosessissa niiden tietojenkalasteluhyökkäys-resistiivisyyden ansiosta. Tässä tutkimuksessa keskityttiin kuitenkin erilaisten ulkoisten järjestelmien tunnisteiden hyödyntämiseen, ei niiden muokkaukseen, joten tämä aihe jäi tutkimuksen ulkopuolelle. Tämä aihe tosin avaa myös mahdollisuuden tutkia, voisiko fyysisten tunnisteiden ja elementtien dataa käyttää tietojenkalastelu-resistiivisellä tavalla autentikointiprosessissa.

7.2 Palaute teoriaan

Tutkimus osoitti, että käyttäjien rutiininomaisesti ja päivittäin käyttämiä fyysisiä tunnisteita ja elementtejä voidaan hyödyntää identiteettien implisiittisessä autentikoinnissa, jolloin uudenlaisen, implisiittisen autentikointimetodin palauttaminen teoriaan on mahdollista. Aiemmat implisiittisiin autentikointimetodeihin liittyneet tutkimukset ovat keskittyneet biometristen tunnistetietojen käyttöön, joiden toiminta perustuu todennäköisyyteen. Tämä aiheuttaa mahdollisen tietoturvariskin, kuten tutkimuksessa on aiemmin mainittu.

Käyttäjän sijainnin varmentaminen olemassa olleiden järjestelmien avulla kykeni varmentamaan käyttäjän sijainnin deterministisesti, jolloin autentikointipäätös ei jäänyt todennäköisyyden varaan. Näin ollen tässä tutkimuksessa esitetty, deterministisin menetelmin toiminut implisiittinen autentikointimetodi voi toimia yhtenä mahdollisena tapana identiteetin monivaiheisessa autentikoinnissa. Aikaisemmissa tutkimuksissa ja tässä tutkimuksessa esitettyjä autentikointimetodeja ei kuitenkaan suositella käytettäväksi identiteetin autentikointiin sellaisenaan, sillä näiden metodien käyttämät autentikaattorit (loki- ja sijaintitiedot) ovat pelkkiä tunnistetietoja, jotka eivät sisällä autentikaattoreille tunnusomaista salaista osaa.

7.3 Luotettavuuden arviointi

7.3.1 Reliabiliteetti

Tutkimuksen reliabiliteetti on pyritty takaamaan toteutusvaiheen ja liitteiden kattavalla dokumentaatiolla testitulosten ja muutosten toistamiseksi testiskenaariota vastaavanlaisissa ympäristössä. Tutkimuksen reliabiliteettia arvioidaan dokumentaation laajuuden ja tulosten toistettavuuden saralta, sillä interventionistisissä tutkimuksissa tutkimuksen luotettavuutta tulisi arvioida muodostetun ratkaisun toimivuudella ja tutkimustulosten toistettavuudella (Kananen 2017, 68–70). Myös Pernaa (2013) mainitsee kehittämistutkimuksen luotettavuuden arvioinnin olevan tarpeellista tulosten siirrettävyyden ja vahvistettavuuden näkökulmasta, sillä kehittämistutkimuksissa käytetyt menetelmät vaihtelevat tutkimustilanteen mukaan, jolloin luotettavuuden arviointi on haasteellista.

Tutkimuksessa käytetyn testiympäristön lähtötilanne on kuvattu tarkasti, mikä mahdollistaa ympäristön uudelleenrakennuksen. Myös testiympäristöön toteutetut muutokset ovat dokumentoitu ja kuvattu riittävän laajasti, jotta replikoitu lähtötilanne voitaisiin muokata tutkimustilannetta vastaavaksi. Tutkimuksen reliabiliteettia vahvistaa myös se, että fyysisten tunnisteiden ja elementtien hyödynnettävyys monivaiheisessa autentikointiprosessissa pystyttiin todentamaan suorittamalla aiemmin käsitelty testiskenaario kahdesti. Kahdesti suoritettu testi varmisti, että autentikointiprosessin toimivuus on yhdistettävissä toteutusvaiheessa mainittuihin muutoksiin, eikä tutkimustulokset perustuneet todennäköisyyteen. Tutkimuksen toistettavuutta toisenlaisissa ympäristöissä on pyritty vahvistamaan julkaisemalla Kuisti-järjestelmän lähdekoodi Github-palveluun, jotta aiheesta kiinnostuneet voisivat toistaa tutkimustulokset itsenäisesti omissa ympäristöissään.

7.3.2 Validiteetti

Tutkimuksen validiteettia päätettiin arvioida tarkastelemalla, miten hyvin tutkimuskysymysten vastaukset auttoivat varsinaisen tutkimusongelman ratkaisussa, ja miten hyvin tutkimuksen toteutusvaiheen lopputulos pyrki ratkaisemaan tutkimusongelman. Tällainen arviointimenetelmä, jossa tutkimustulokset ja niistä johdetut vastaukset suhteutetaan tutkimusongelman ratkaisuun, kertoo siitä, miten hyvin tutkimus kokonaisuudessaan kohdistettiin. Tällä tavoin voidaan varmentua siitä, että tutkimuksessa tarkasteltiin oikeita asioita, mikä on Kanasen (2017, 71) mukaan tärkeää laadullisessa validiteetin arvioinnissa. Pernaa (2013) mainitsee, että laadullisia havaintoja pystytään tukemaan määrällisin mittauksin, mikä parantaa kehittämistutkimuksen luotettavuutta.

Tutkimuksessa saatiin onnistuneesti rakennettua paikallisesti toiminut järjestelmä, joka poisti autentikointiprosessiin liittyneet riippuvuudet pilvi-infrastruktuuriin. Kehitetty järjestelmä kykeni myös autentikoimaan identiteetin implisiittisesti fyysisiä tunnisteita ja elementtejä käyttäen, sekä yhdistämään luodun autentikointimetodin osaksi sisäisen tietoverkon monivaiheista autentikointiprosessia. Vaikka ratkaisu poistaa eksplisiittisen toiminnan autentikointiprosessin ensimmäisestä metodista, ei se kuitenkaan täysin kykene torjumaan inhimillisistä virheistä johtuvia uhkia tässä luvussa aiemmin mainituista syistä.

Näin ollen tutkimuksen voidaan todeta olleen validi, sillä tutkimuskysymysten vastaukset tukevat tutkimusongelman ratkaisua, vaikka ratkaisu jääkin hieman vajaaksi. Validiteettiin vaikuttaa positiivisesti myös toteutusvaiheen ja liitteiden dokumentaatio, jonka avulla tutkimus voidaan toistaa, ja toiston yhteydessä tutkimuksen kohdistus ja vaikuttavuus autentikointiprosessiin varmentaa. Vajaaksi jäänyttä ratkaisua tulisi täydentää jatkotutkimuksen ja -kehityksen yhteydessä.

7.4 Jatkotutkimus ja -kehitys

Pääsääntöiseksi jatkotutkimusaiheeksi tämän tutkimuksen jälkeen jäi kehitetyn autentikointimetodin ja -prosessin käyttö sosiaalisen manipuloinnin hyökkäysten estämisessä. Nykyisessä muodossaan Kuisti-järjestelmä kykenee ainoastaan lisäämään autentikointiprosessiin implisiittisen autentikointimetodin, mutta tämä autentikointimetodi ei täysin kykene estämään sosiaalisen manipuloinnin hyökkäyksiä tässä luvussa aiemmin perustelluista syistä, jotka liittyvät fyysisten tunnisteiden turvallisuuteen. Tämän vuoksi Kuisti-järjestelmän tarjoamaa konseptia tulisi kokeilla ja kehittää tietojenkalastelu-resistenttien tunnisteiden tai elementtien kanssa. Kuten aiemmin mainittiin, yksi varteenotettava vaihtoehto tällaiseen tutkimukseen olisi passkey-avaimet tai niiden toimintaan perustuvat tunnisteet tai järjestelmät.

Tutkimuksen aikana nousi esiin myös kysymys useamman palomuurilaitteen käytöstä. Tällä hetkellä Kuisti tukee vain yhden palomuurilaitteen käyttöä, mutta tutkimuksen toteutusvaiheen aikana Kuistin konfiguraatiotiedostot muokattiin tukemaan useampaa palomuurilaitetta jatkokehitystä ajatellen. Useamman palomuurilaitteen käyttö Kuistin toiminnoissa olisi hyödyllistä esimerkiksi Purdue-mallia käyttävissä ympäristöissä, joissa verkon eri alueita erotellaan toisistaan useampien palomuurien avulla (Fortinet s.a.).

Jatkokehityksen puitteissa tulisi myös tarkastella Kuisti-järjestelmän turvallisuutta. Tämän tutkimuksen aikana käytetty Kuisti-järjestelmä ei esimerkiksi varmenna sille syötettyä dataa, mikä avaa mahdollisuuden väärinkäytölle ja autentikointimetodien ohitukselle. Kuistiin pystytään kuitenkin lisäämään TLS-salaus lokitietojen eheyden ja luottamuksellisuuden takaamiseksi, sillä kom-

munikointi toimialuepalvelimen kanssa tapahtuu TCP-protokollan päällä. Ulkoiset järjestelmät eivät kuitenkaan välttämättä tue TCP-protokollaa, minkä takia ExtSystemListener-komponenttia voidaan käyttää myös UDP:n kanssa.

Kuistin suorituskykyä olisi myös hyvä mitata ja parantaa. Tässä tutkimuksessa järjestelmää testattiin vain yhdellä käyttäjätunnuksella hyvin valmistellussa ympäristössä, jolloin järjestelmän suorituskyky ei haitannut tutkimusongelman ratkaisua. Tuotantoympäristöissä järjestelmän suorituskyky voi kuitenkin olla huono käyttäjien ja tapahtumien määrän kasvaessa. Tämän vuoksi Kuistin suorituskyvyn mittaus ja mahdollinen parannus on perusteltua jatkokehityksen kannalta, jos järjestelmää halutaan käyttää tuotantoympäristöissä.

LÄHTEET

Adaptive Multi-Factor Authentication s.a. Silverfort. WWW-dokumentti. Saatavissa: https://www.silverfort.com/glossary/adaptive-multi-factor-authentication/ [viitattu 5.2.2025].

Arias-Cabarcos, P., Krupitzer, C. & Becker, C. 2019. A Survey on Adaptive Authentication. *ACM Computing Surveys* 52, 1–30. PDF-dokumentti. Saatavissa: https://doi.org/10.1145/3336117 [viitattu 5.2.2025].

Awati, R s.a. Risk-based authentication (RBA). TechTarget. WWW-dokumentti. Saatavissa: https://www.techtarget.com/searchsecurity/definition/risk-based-authentication-RBA [viitattu 2.2.2025].

Bayuk, J. L., Healey, J., Rohmeyer, P., Sachs, M. H., Schmidt, J. & Weiss, J. 2012. Cyber Security Policy Guidebook. Hoboken: Wiley.
BeyondTrust s.a. MFA Fatigue Attack. WWW-dokumentti. Saatavissa: https://www.beyondtrust.com/resources/glossary/mfa-fatigue-attack [viitattu 20.1.2025].

Bonneau, J., Bursztein, E., Caron, I., Jackson, R. & Williamson, M. 2015. Secrets, Lies, and Account Recovery: Lessons from the Use of Personal Knowledge Questions at Google. *WWW '15: Proceedings of the 24th International Conference on World Wide Web* 2015, 141–150. PDF-dokumentti. Saatavissa: https://doi.org/10.1145/2736277.2741691 [viitattu 23.1.2025].

Brooks, C. J., Grow, C., Craig, P. & Short, D. 2018. Cybersecurity essentials. Indianapolis: Sybex.

CISA. 2022a. Multi-factor Authentication. PDF-dokumentti. Saatavissa: https://www.cisa.gov/sites/default/files/publications/MFA-Fact-Sheet-Jan22-508.pdf [viitattu 23.1.2025].

CISA. 2022b. Implementing Phishing-Resistant MFA . PDF-dokumentti. Saatavissa: https://www.cisa.gov/sites/default/files/publications/fact-sheet-implementing-phishing-resistant-mfa-508c.pdf [viitattu 28.1.2025].

Cisco. 2023. Authentication Failures on DUO1. WWW-dokumentti. Päivitetty 1.9.2023. Saatavissa: https://status.duo.com/incidents/rw7g0q7ztj8f [viitattu 28.1.2025].

Cole, E., Krutz, R. L., Conley, J. W., Reisman, B., Ruebush, M., Gollman, D. & Reese, R. 2008. Network security fundamentals. Hoboken: Wiley.

Elisa s.a. Elisa Paikannuspalvelu. WWW-dokumentti. Saatavissa: https://yrityksille.elisa.fi/terveydenhuollon-paikannusratkaisu [viitattu 4.3.2025].

F5. 2023. Solving for Account Takeover: Why MFA is Only a First Step. WWW-dokumentti. Julkaistu 26.6.2023. Saatavissa: https://www.f5.com/resources/white-papers/solving-for-account-takeover-why-mfa-is-only-a-first-step [viitattu 3.2.2025].

FIDO Alliance s.a. How Passkeys Work. WWW-dokumentti. Saatavissa: https://www.passkeycentral.org/introduction-to-passkeys/how-passkeys-work [viitattu 30.1.2025].

Fortinet s.a. Purdue Model for ICS Security. WWW-dokumentti. Saatavissa: https://www.fortinet.com/resources/cyberglossary/purdue-model [viitattu 2.4.2025].

Gatlan, S. 2025. Microsoft MFA outage blocking access to Microsoft 365 apps. BleepingComputer. WWW-dokumentti. Päivitetty 13.1.2025. Saatavissa: https://www.bleepingcomputer.com/news/microsoft/microsoft-mfa-outage-blocking-access-to-microsoft-365-apps/ [viitattu 28.1.2025].

Google s.a. Google Python Style Guide. WWW-dokumentti. Saatavissa: https://google.github.io/styleguide/pyguide.html [viitattu 4.3.2025].

Google. 2019. Online Security Survey. PDF-dokumentti. Saatavissa: https://services.google.com/fh/files/blogs/google_security_infographic.pdf [viitattu 2.4.2025].

Grassi, P. A., Fenton, J. L., Newton, E. M., Perlner, R. A., Regenscheid, A. R., Burr, W. R. & Richer, J. P. 2017. NIST Special Publication 800-63B. National Institute of Standards and Technology. PDF-dokumentti. Päivitetty 2.3.2020. Saatavilla: https://doi.org/10.6028/NIST.SP.800-63b [viitattu 23.1.2025].

Grassi, P. A., Garcia, M. E. & Fenton, J. L. 2017. NIST Special Publication 800-63-3. National Institute of Standards and Technology. PDF-dokumentti. Päivitetty 2.3.2020. Saatavilla: https://doi.org/10.6028/NIST.SP.800-63-3 [viitattu 2.2.2025].

Grimes, R. 2019. 12+ Ways to Hack Multi-Factor Authentication. KnowBe4. PDF-dokumentti. Saatavissa: https://www.knowbe4.com/hubfs/12+_Ways_to_Hack_Two-Factor_Authentication-1.pdf [viitattu 3.2.2025].

HID. 2025. iCLASS® SE™ Reader Family Datasheet. WWW-dokumentti. Julkaistu 14.1.2025. Saatavissa: https://www.hidglobal.com/documents/iclass-se-reader-family-datasheet [viitattu 30.3.2025].

Hodges, J., Jones, J. C., Jones, M. B., Kumar, A. & Lundberg, E. 2021. Web Authentication: An API for accessing Public Key Credentials Level 2. W3C. WWW-dokumentti. Päivitetty 8.4.2021. Saatavissa: https://www.w3.org/TR/2021/REC-webauthn-2-20210408/ [viitattu 4.2.2025].

Jakobsson, M., Shi, E., Golle, P. & Chow, R. 2009. Implicit Authentication for Mobile Devices. *HotSec'09: Proceedings of the 4th USENIX conference on Hot topics in security* 2009, 9–15. PDF-dokumentti. Saatavissa: https://dl.acm.org/doi/10.5555/1855628.1855637 [viitattu 5.2.2025].

Kananen, J. 2017. Kehittämistutkimus interventiotutkimuksen muotona: Opas opinnäytetyön ja pro gradun kirjoittajalle. Jyväskylä: Jyväskylän ammattikorkeakoulu.

Khan, H., Hengartner, U., & Vogel, D. 2015. Usability and Security Perceptions of Implicit Authentication: Convenient, Secure, Sometimes Annoying. *Symposium on Usable Privacy and Security (SOUPS)* 2015, 225–239. PDF-dokumentti. Saatavissa: https://www.usenix.org/system/files/conference/soups2015/soups15-paper-khan.pdf [viitattu 6.2.2025].

Kuisti. 2025. GitHub. WWW-dokumentti. Päivitetty 22.3.2025. Saatavissa: https://github.com/toikkaroija/kuisti [viitattu 24.3.2025].

Lukka, K. 2001. Konstruktiivinen tutkimusote. Metodix Oy. WWW-dokumentti. Saatavissa: https://metodix.fi/2014/05/19/lukka-konstruktiivinen-tutkimusote/ [viitattu 21.1.2025].

Machani, S., Philpott, R., Srinivas, S., Kemp, J. & Hodges, J. 2020. FIDO UAF Architectural Overview. FIDO Alliance. PDF-dokumentti. Päivitetty 20.10.2020. Saatavilla: https://fidoalliance.org/specs/fido-uaf-v1.2-ps-20201020/fido-uaf-overview-v1.2-ps-20201020.pdf [viitattu 30.1.2025].

Meyer, L., Romero S., Bertoli, G., Burt, T, Weinert, A. & Ferres, J. 2023. How effective is multifactor authentication at deterring cyberattacks? arXiv. Verkkojulkaisu. Päivitetty 1.5.2023. Saatavissa: https://doi.org/10.48550/arXiv.2305.00945 [viitattu 19.1.2025].

Microsoft. 2024a. Identity and access management (IAM) fundamental concepts. WWW-dokumentti. Päivitetty 31.5.2024. Saatavissa: https://learn.microsoft.com/en-us/entra/fundamentals/identity-fundamental-concepts [viitattu 29.1.2025].

Microsoft. 2024b. Use Windows Event Forwarding to help with intrusion detection. WWW-dokumentti. Päivitetty 10.7.2024. Saatavissa: https://learn.microsoft.com/en-us/windows/security/operating-system-security/device-management/use-windows-event-forwarding-to-assist-in-intrusion-detection [viitattu 17.3.2025].

Microsoft. 2024c. [MS-KILE]: Kerberos Protocol Extensions. WWW-dokumentti. Päivitetty 7.6.2024. Saatavissa: https://learn.microsoft.com/en-us/openspecs/windows_protocols/MS-KILE/2a32282e-dd48-4ad9-a542-609804b02cc9 [viitattu 18.3.2025].

Microsoft. 2025. Getting started with the Azure Multi-Factor Authentication Server. WWW-dokumentti. Päivitetty 14.1.2025. Saatavissa: https://learn.microsoft.com/en-us/previous-versions/entra/identity/authentication/howto-mfaserver-deploy [viitattu 27.1.2025].

Moodley, E., Huo, G., Hsieh, M., Cai, S. & Yan, W. Q. 2014. Password Security and Protection. Teoksessa Thampi, S. M., Bhargava, B. & Atrey, P. K. (toim.) Managing Trust in Cyberspace. Boca Raton: CRC Press, 449–470.

NXP Semiconductors. 2018. AN10927. MIFARE product and handling of UIDs. PDF-dokumentti. Päivitetty 5.7.2018. Saatavissa: https://www.nxp.com/docs/en/application-note/AN10927.pdf [viitattu 30.3.2025].

OWASP s.a. Multifactor Authentication Cheat Sheet. Saatavissa: https://cheatsheetseries.owasp.org/cheatsheets/Multifactor_Authentication_Cheat_Sheet.html [viitattu 23.1.2025].

Passkey Security s.a. FIDO Alliance. WWW-dokumentti. Saatavissa: https://www.passkeycentral.org/introduction-to-passkeys/passkey-security [viitattu 30.1.2025].

Password Storage Cheat Sheet s.a. OWASP. WWW-dokumentti. Saatavissa: https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html [viitattu 2.4.2025].

Peppin, R. 2024. Simplify Device Tracking in Workplaces with Wi-Fi Real-Time Location Systems (RTLS). Cisco. WWW-dokumentti. Päivitetty 19.1.2024. Saatavissa: https://spaces.cisco.com/simplify-device-tracking-in-workplaces-with-wi-fi-real-time-location-systems-rtls/ [viitattu 4.3.2025].

Pernaa, J. 2013. Kehittämistutkimus tutkimusmenetelmänä. Julkaisussa J. Pernaa (toim.), Kehittämistutkimus opetusalalla. Jyväskylä: PS-kustannus, 9–26. Saatavilla: http://hdl.handle.net/10138/317958 [viitattu 21.1.2025].

Piscitello, D. 2015. What is Authorization and Access Control? ICANN. WWW-dokumentti. Päivitetty 2.12.2015. Saatavissa: https://www.icann.org/en/blogs/details/what-is-authorization-and-access-control-2-12-2015-en [viitattu 25.1.2025].

Python Software Foundation s.a. Thread-based parallelism. WWW-dokumentti. Saatavissa: https://docs.python.org/3/library/threading.html [viitattu 27.2.2025].

Rose, S., Borchert, O., Mitchell, S. & Connelly, S. 2020. NIST Special Publication 800-207. Zero Trust Architecture. National Institute of Standards and Technology. PDF-dokumentti. Julkaistu 10.8.2020. Saatavissa: https://doi.org/10.6028/NIST.SP.800-207 [viitattu 12.3.2025].

RSA Security. 2024. Understanding RSA Security: An overview of RSA security practices, operations, and controls. PDF-dokumentti. Saatavissa: https://www.rsa.com/wp-content/uploads/security-overview-rsa-whitepaper.pdf [viitattu 30.1.2025].

SolarWinds s.a. What is Database Concurrency? WWW-dokumentti. Saatavissa: https://www.solarwinds.com/resources/it-glossary/database-concurrency [viitattu 4.3.2025].

Spitzner, L. 2022. What is Phishing Resistant MFA? SANS. WWW-dokumentti. Päivitetty 6.9.2022. Saatavissa: https://www.sans.org/blog/what-is-phishing-resistant-mfa/ [viitattu 28.1.2025].

Taku, D. 2023. Anatomy of the Attack: The Rise and Fall of MFA. RSA Conference. Videoleike. Päivitetty 25.4.2023. Saatavissa: https://www.rsaconference.com/library/presentation/usa/2023/Anatomy%20of%20the%20Attack%20The%20Rise%20and%20Fall%20of%20MFA [viitattu 28.1.2025].

Traficom s.a. Monivaiheinen tunnistautuminen suojaa käyttäjätilejäsi. WWW-dokumentti. Saatavissa: https://www.kyberturvallisuuskeskus.fi/fi/ajankohtaista/ohjeet-ja-oppaat/monivaiheinen-tunnistautuminen-suojaa-kayttajatilejasi [viitattu 19.1.2025].

Traficom. 2023. Elektroninen SIM tarjoaa uuden hyökkäysvektorin rikollisille. WWW-dokumentti. Julkaistu 4.7.2023. Saatavissa: https://www.kyberturvallisuuskeskus.fi/fi/ajankohtaista/elektroninen-sim-tarjoaa-uuden-hyokkaysvektorin-rikollisille [viitattu 3.2.2025].

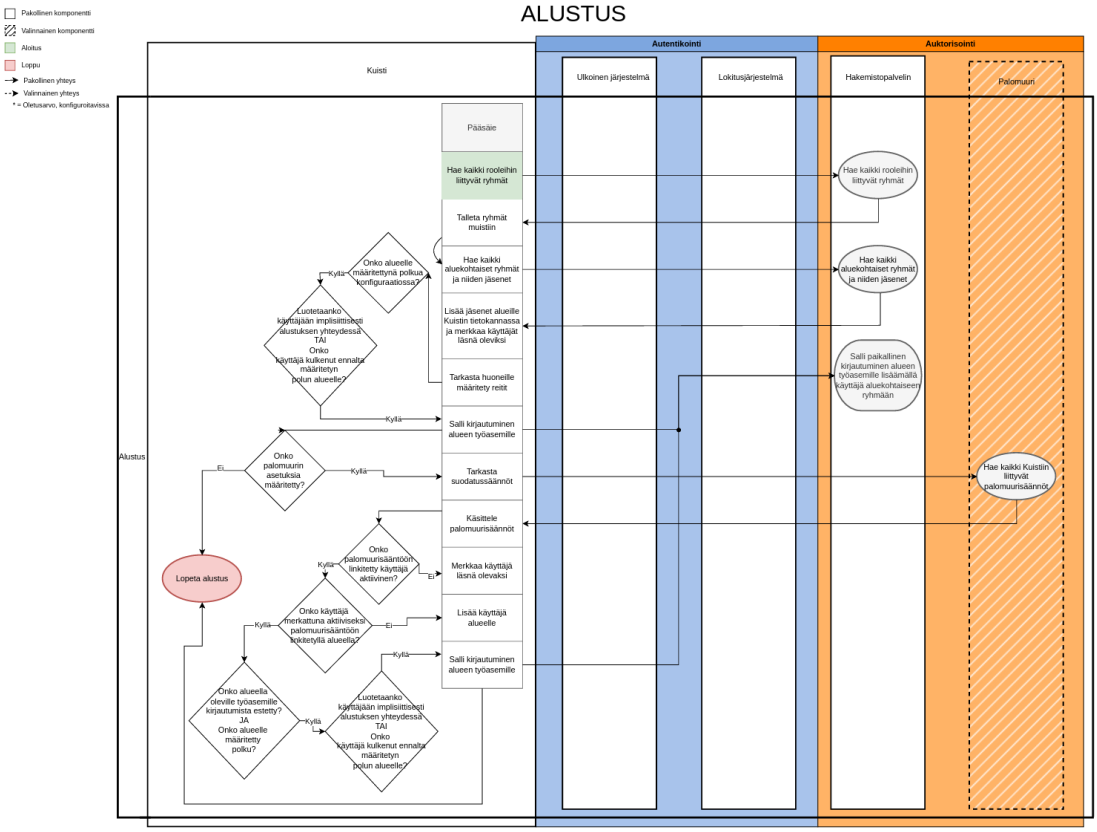
Veerubhotla, R. S. & Garg, R. 2014. A Walk-Through of Online Identity Management. Teoksessa Thampi, S. M., Bhargava, B. & Atrey, P. K. (toim.) Managing Trust in Cyberspace. Boca Raton: CRC Press, 239–262.

Windows Security Log Event ID 4800 s.a. Ultimate IT Security. WWW-dokumentti. Saatavissa: https://www.ultimatewindowssecurity.com/securitylog/encyclopedia/event.aspx?eventID=4800 [viitattu 18.3.2025].

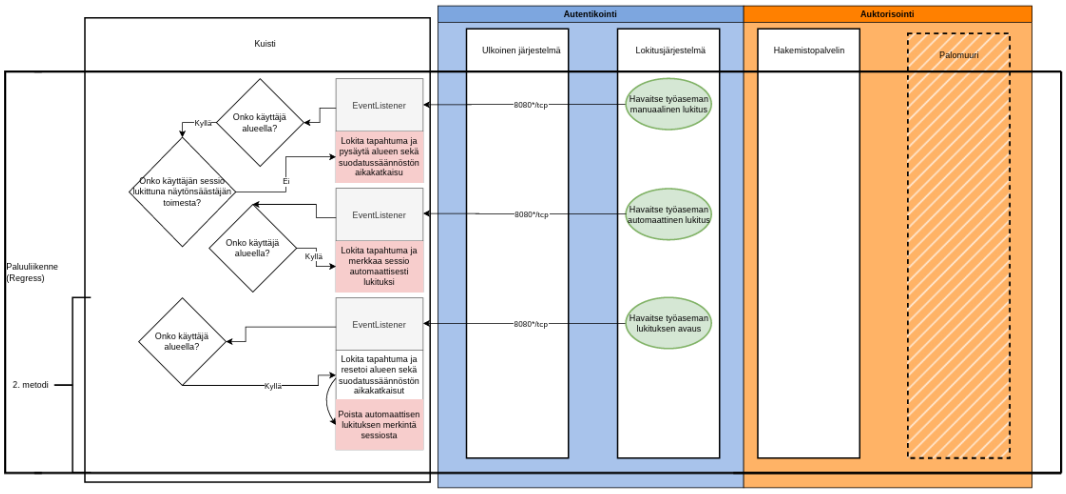
Windows Security Log Event ID 4802 s.a. Ultimate IT Security. WWW-dokumentti. Saatavissa: https://www.ultimatewindowssecurity.com/securitylog/encyclopedia/event.aspx?eventID=4802 [viitattu 18.3.2025].

Yao, F., Suleiman, Y. Y., Kang, B. & Sezer, S. 2017. Continuous implicit authentication for mobile devices based on adaptive neuro-fuzzy inference system. *International Conference on Cyber Security And Protection Of Digital Services (Cyber Security)* 2017, 1–7. PDF-dokumentti. Saatavissa: https://doi.org/10.1109/CyberSecPODS.2017.8074846 [viitattu 5.2.2025].

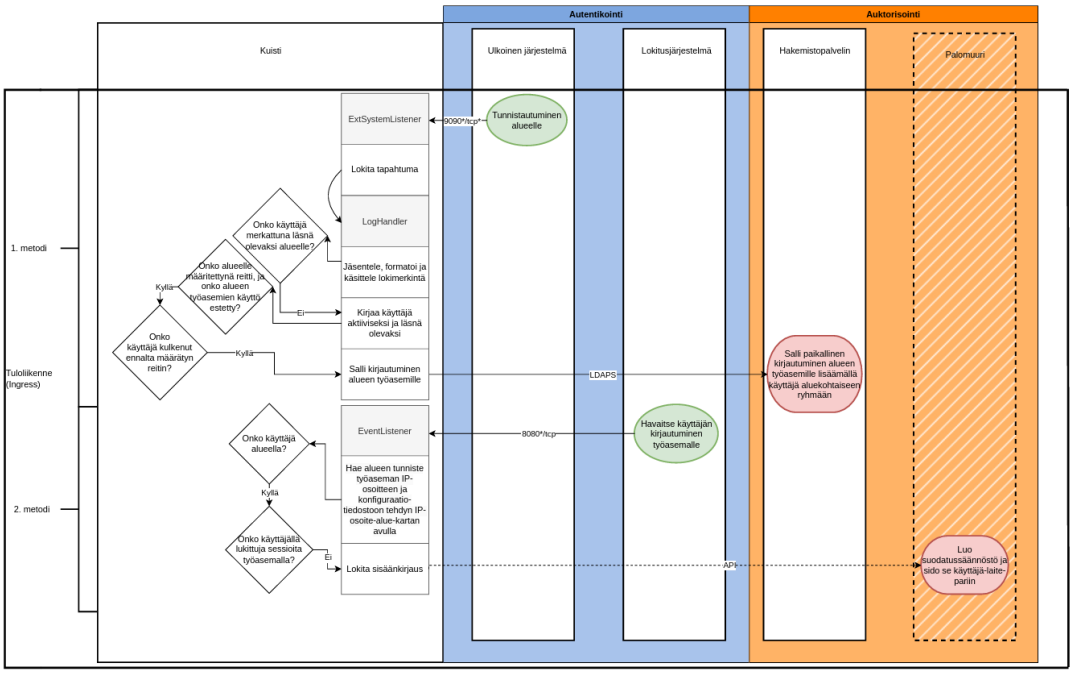
ALUSTUS



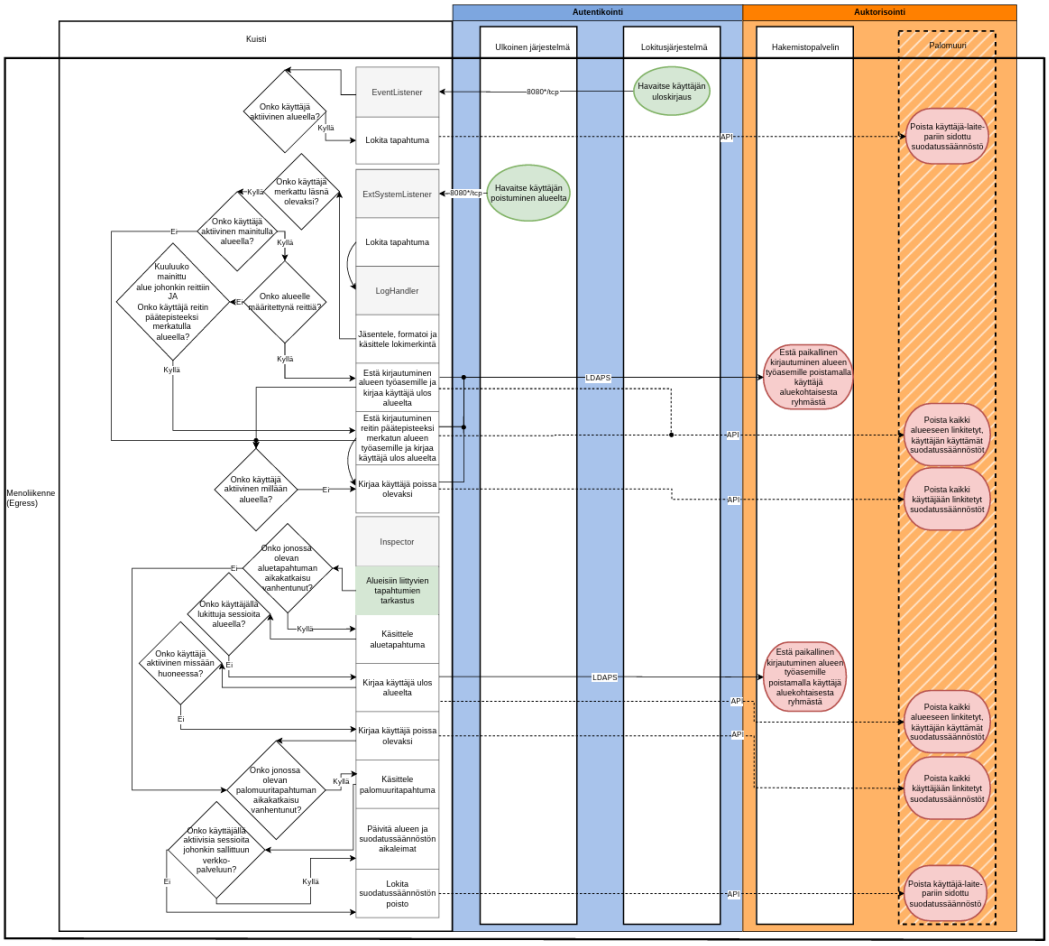
PALUULIIKENNE (REGRESS)

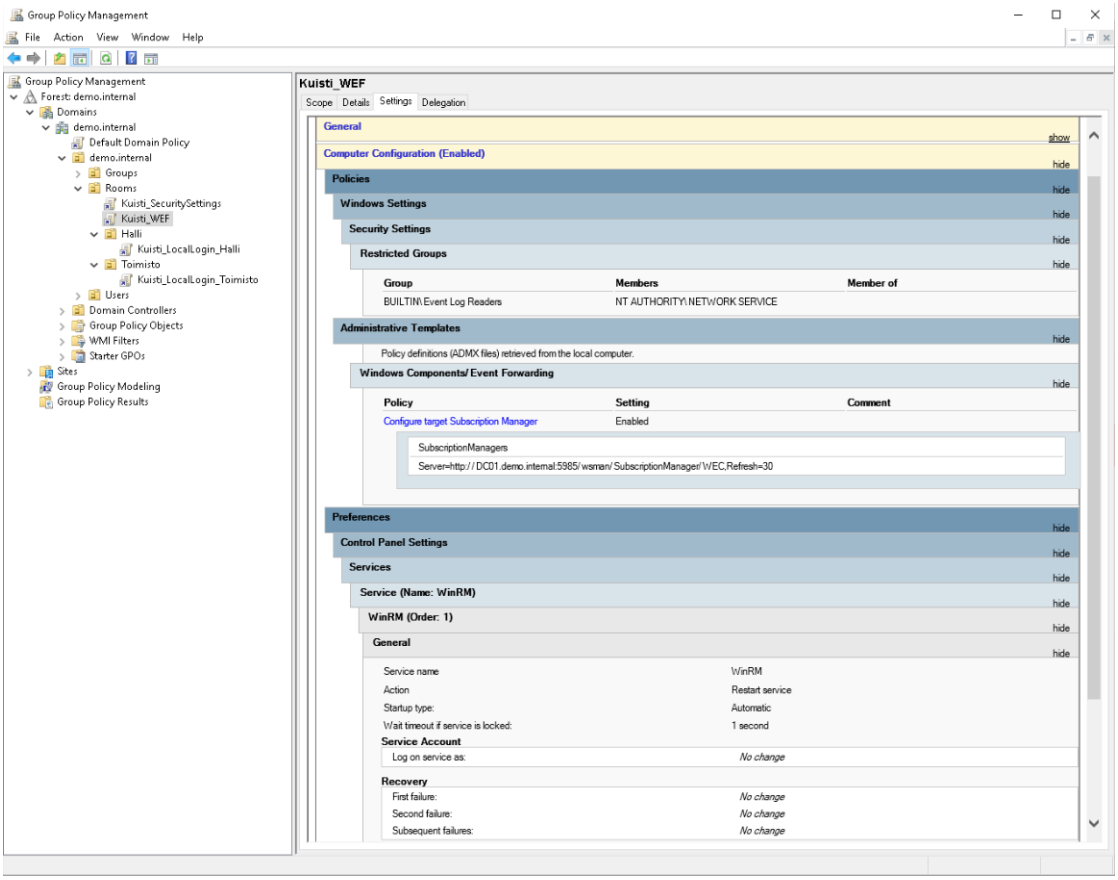


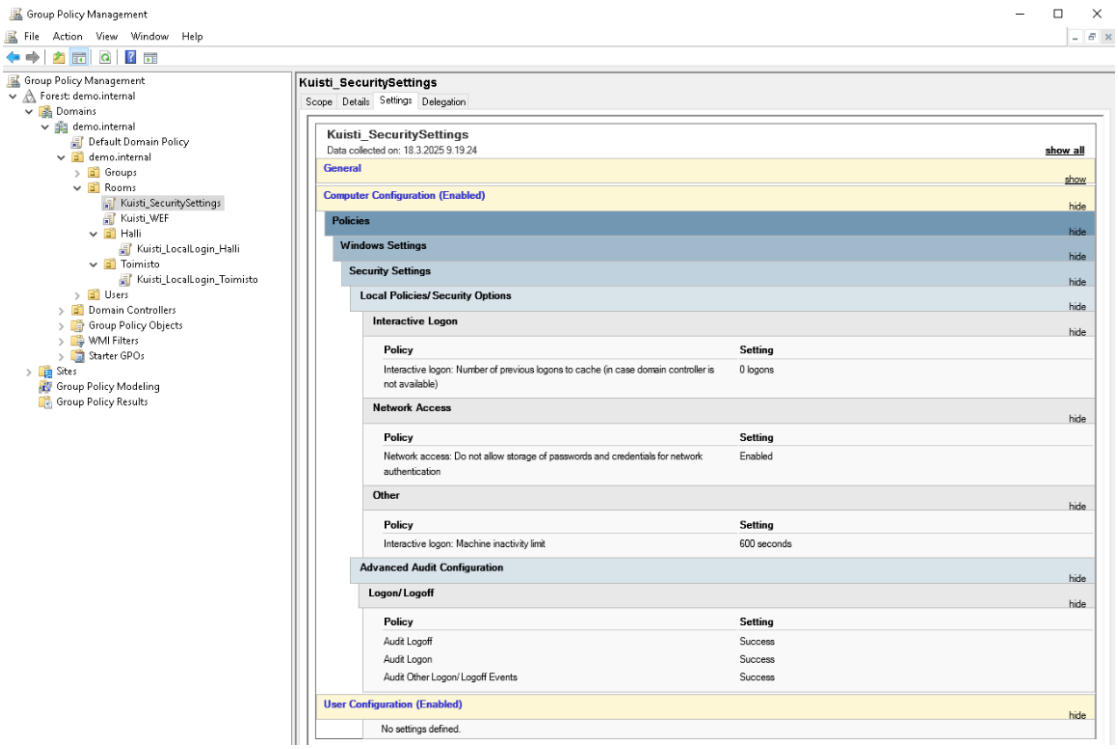
TULOLIIKENNE (INGRESS)

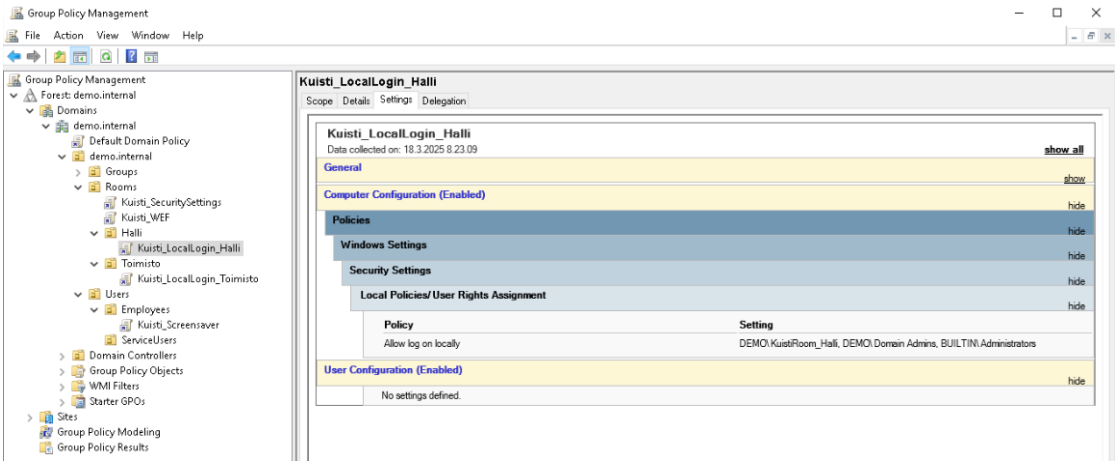


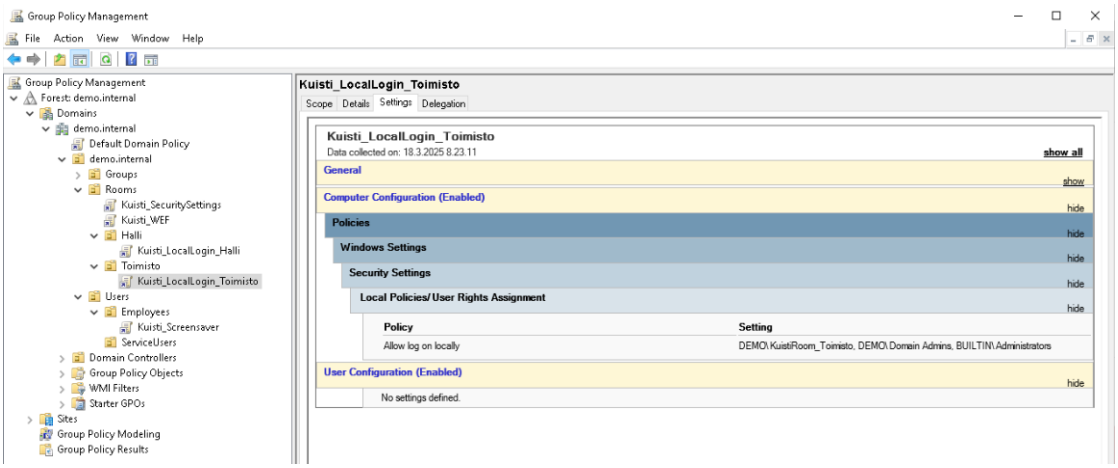
MENOLIIKENNE (EGRESS)

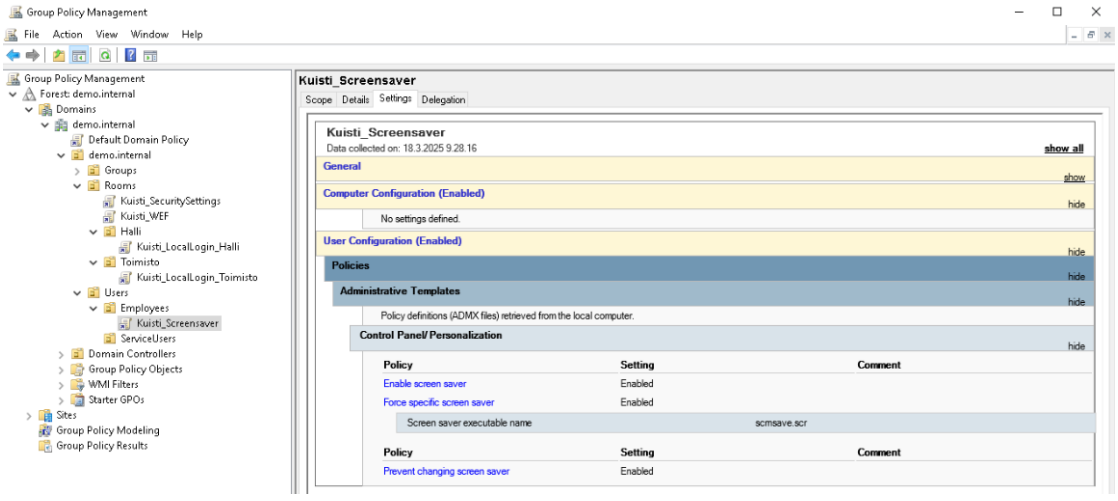



Group Policy Management
File Action View Window Help
Group Policy Management
Forest: demo.internal
Domains
demo.internal
Default Domain Policy
demo.internal
Groups
Rooms
Kuisti_SecuritySettings
Kuisti_WEF
Halli
Kuisti_LocalLogin_Halli
Toimisto
Kuisti_LocalLogin_Toimisto
Users
Domain Controllers
Group Policy Objects
WMI Filters
Starter GPOs
Sites
Group Policy Modeling
Group Policy Results
Kuisti_WEF
Scope Details Settings Delegation
General
show
Computer Configuration (Enabled)
hide
Policies
Windows Settings
Security Settings
Restricted Groups
Group
Members
Member of
BUILTIN\Event Log Readers
NT AUTHORITY\NETWORK SERVICE
Administrative Templates
Policy definitions (ADMX files) retrieved from the local computer.
Windows Components/ Event Forwarding
Policy
Setting
Comment
Configure target Subscription Manager
Enabled
SubscriptionManagers
Server=http://DC01.demo.internal:5985/wsman/SubscriptionManager/WEC,Refresh=30
Preferences
Control Panel Settings
Services
Service (Name: WinRM)
WinRM (Order: 1)
General
Service name
WinRM
Action
Restart service
Startup type:
Automatic
Wait timeout if service is locked:
1 second
Service Account
Log on service as:
No change
Recovery
First failure:
No change
Second failure:
No change
Subsequent failures:
No change


Group Policy Management
File Action View Window Help
Group Policy Management
Forest: demo.internal
Domains
demo.internal
Default Domain Policy
demo.internal
Groups
Rooms
Kuisti_SecuritySettings
Kuisti_WEF
Halli
Kuisti_LocalLogin_Halli
Toimisto
Kuisti_LocalLogin_Toimisto
Users
Domain Controllers
Group Policy Objects
WMI Filters
Starter GPOs
Sites
Group Policy Modeling
Group Policy Results
Kuisti_SecuritySettings
Scope Details Settings Delegation
Kuisti_SecuritySettings
Data collected on: 18.3.2025 9.19.24
show all
General
show
Computer Configuration (Enabled)
hide
Policies
Windows Settings
Security Settings
Local Policies/ Security Options
Interactive Logon
Policy
Setting
Interactive logon: Number of previous logons to cache (in case domain controller is not available)
0 logons
Network Access
Policy
Setting
Network access: Do not allow storage of passwords and credentials for network authentication
Enabled
Other
Policy
Setting
Interactive logon: Machine inactivity limit
600 seconds
Advanced Audit Configuration
Logon/ Logoff
Policy
Setting
Audit Logoff
Success
Audit Logon
Success
Audit Other Logon/ Logoff Events
Success
User Configuration (Enabled)
No settings defined.


Group Policy Management
File Action View Window Help
Group Policy Management
Forest: demo.internal
Domains
demo.internal
Default Domain Policy
demo.internal
Groups
Rooms
Kuisti_SecuritySettings
Kuisti_WEF
Halli
Kuisti_LocalLogin_Halli
Toimisto
Kuisti_LocalLogin_Toimisto
Users
Employees
Kuisti_Screensaver
ServiceUsers
Domain Controllers
Group Policy Objects
WMI Filters
Starter GPOs
Sites
Group Policy Modeling
Group Policy Results
Kuisti_LocalLogin_Halli
Scope Details Settings Delegation
Kuisti_LocalLogin_Halli
Data collected on: 18.3.2025 8.23.09
show all
General
show
Computer Configuration (Enabled)
hide
Policies
hide
Windows Settings
hide
Security Settings
hide
Local Policies/User Rights Assignment
hide
Policy
Setting
Allow log on locally
DEMO\KuistiRoom_Halli, DEMO\Domain Admins, BUILTIN\Administrators
User Configuration (Enabled)
hide
No settings defined.


Group Policy Management
File Action View Window Help
Group Policy Management
Forest: demo.internal
Domains
demo.internal
Default Domain Policy
demo.internal
Groups
Rooms
Kuisti_SecuritySettings
Kuisti_WEF
Halli
Kuisti_LocalLogin_Halli
Toimisto
Kuisti_LocalLogin_Toimisto
Users
Employees
Kuisti_Screensaver
ServiceUsers
Domain Controllers
Group Policy Objects
WMI Filters
Starter GPOs
Sites
Group Policy Modeling
Group Policy Results
Kuisti_LocalLogin_Toimisto
Scope Details Settings Delegation
Kuisti_LocalLogin_Toimisto
Data collected on: 18.3.2025 8.23.11
show all
General
show
Computer Configuration (Enabled)
hide
Policies
hide
Windows Settings
hide
Security Settings
hide
Local Policies/User Rights Assignment
hide
Policy
Setting
Allow log on locally
DEMO\KuistiRoom_Toimisto, DEMO\Domain Admins, BUILTIN\Administrators
User Configuration (Enabled)
hide
No settings defined.


Group Policy Management
File Action View Window Help
Group Policy Management
Forest: demo.internal
Domains
demo.internal
Default Domain Policy
demo.internal
Groups
Rooms
Kuisti_SecuritySettings
Kuisti_WEF
Halli
Kuisti_LocalLogin_Halli
Toimisto
Kuisti_LocalLogin_Toimisto
Users
Employees
Kuisti_Screensaver
ServiceUsers
Domain Controllers
Group Policy Objects
WMI Filters
Starter GPOs
Sites
Group Policy Modeling
Group Policy Results
Kuisti_Screensaver
Scope Details Settings Delegation
Kuisti_Screensaver
Data collected on: 18.3.2025 9.28.16
show all
General
show
Computer Configuration (Enabled)
hide
No settings defined.
User Configuration (Enabled)
hide
Policies
hide
Administrative Templates
hide
Policy definitions (ADMX files) retrieved from the local computer.
Control Panel/Personalization
hide
Policy
Setting
Comment
Enable screen saver
Enabled
Force specific screen saver
Enabled
Screen saver executable name
scrnsave.scr
Policy
Setting
Comment
Prevent changing screen saver
Enabled

filtersets.json

```
{

    "default": {

        "filters": [

            {

                "action": "pass",
                "dstAddr": "*",
                "dstPort": "*",
                "protocol": "icmp",
                "ipVersion": "4",
                "sequence": 1

            },

            {

                "action": "block",
                "dstAddr": "10.0.0.0/24",
                "dstPort": "443",
                "protocol": "tcp",
                "ipVersion": "4",
                "sequence": 2

            }

        ],

        "timeout": 0

    },

    "tyontekija": {

        "filters": [

            {

                "action": "pass",
                "dstAddr": "*",
                "dstPort": "443",
                "protocol": "tcp",
                "ipVersion": "4",
                "sequence": 11

            },

            {

                "action": "pass",
                "dstAddr": "10.0.0.14",
                "dstPort": "21",
                "protocol": "tcp",
                "ipVersion": "4",
                "sequence": 12

            }

        ],

        "monitoredServices": {

            "learn.xamk.fi": 443,
            "virtual.ictlab.fi": 443,
            "10.0.0.14": 21

        },

        "timeout": 30,
        "renewalAmount": 14

    }

}
```

environment.json

```
{

    "common": {

        "localIpAddress": "10.0.0.12",
        "localEventListenerPort": 8080,
        "localExtSystemListenerPort": 9090,
        "implicitTrustAtBoot": true

    },

    "ldap": {

        "domain": "demo.internal",
        "ditSearchBase": "DC=demo,DC=internal",
        "serviceUser": "kuisti",
        "roomPreFix": "KuistiRoom_",
        "roomDitAttr": "cn",
        "rolePreFix": "KuistiRole_",
        "roleDitAttr": "cn",
        "userDitAttr": "userPrincipalName"

    },

    "firewalls": {

        "fw01": {

            "ipAddress": "10.0.0.1",
            "apiKey": "qw3haYSaTpS8NQo2UchcaTsGmsGQ2PI0z1VH1XGzA1L4sYGFPta9meMy7x+AV3IIvhkcbMU8XPwgiXQp",
            "apiSecret": "4QjvFrBlmxX3fEAV4TlkumbFs2ha3GIpjSZr3TFwzbcydTh36IUrir77gfeQcIpk14+X8Xv7IdvVu3XK",
            "filtersetsPath": "filtersets.json"

        }

    },

    "networks": {

        "Toimisto": "10.14.11.0/24",
        "Halli": "10.14.13.0/24"

    },

    "routes": {

        "Toimisto": ["aula", "toimisto"],
        "Halli": ["aula", "halli"]

    },

    "roomTimeouts": {

        "aula": 480,
        "toimisto": 300,
        "halli": 300

    }

}
```

log_detection.json

```
{

    "kuluvalvontajärjestelmä": {

        "detection": {

            "user": {

                "regexp": "'personName': '(.+?,\\s.+?)\\s\\(\\d+\\)'",
                "matchInGroup": 1

            },

            "directionIn": {

                "regexp": "'description': 'Access Granted - Entry Made'",
                "matchInGroup": 0

            },

            "directionOut": {

                "regexp": "'description': 'Egress Granted'",
                "matchInGroup": 0

            },

            "room": {

                "regexp": "'deviceName': '(.+?)(?:\\s+.+?)?'",
                "matchInGroup": 1

            }

        },

        "formatting": {

            "user": {

                "pattern": "^(\\w+),\\s(\\w+).*$",
                "repl": "\\2.\\1@demo.internal"

            },

            "room": {

                "pattern": "^(\\w+)$",
                "repl": "\\1"

            }

        }

    },

    "työajanseuranta": {

        "detection": {

            "user": {

                "regexp": "user:\\s(.+?),",
                "matchInGroup": 1

            },

            "directionIn": {

                "regexp": "event: Clock in",
                "matchInGroup": 0

            },

            "directionOut": {

                "regexp": "event: Clock out",
                "matchInGroup": 0

            },

            "room": {

                "regexp": "device: (.+?)$",
                "matchInGroup": 1

            }

        },

        "formatting": {

            "user": {

                "pattern": "^(.+?) (.+)$",
                "repl": "\\1.\\2@demo.internal"

            },

            "room": {

                "pattern": "^.+$",
                "repl": "Aula"

            }

        }

    }

}
```

environment.json

PÄÄOSIO	ALAOSIOT		ARVON TYYPPI	SELITE
common	localIpAddress	-	str	Kuisti-palvelimen käyttämä IP-osoite.
	localEventListenerPort	-	int	EventListener-komponentin käyttämä portti.
	localExtSystemListenerPort	-	int	ExtSystemListener-komponentin käyttämä portti.
	implicitTrustAtBoot	-	bool	Asetus määrittää, luotetaanko käyttäjän kulkeneen konfiguraatiotiedostoon määritetyn polun huoneeseen, kun Kuisti tarkastaa tilassa olevat aktiiviset käyttäjät. Jos asetuksen arvo on false, käyttäjä noteerataan aktiiviseksi, mutta työasemille kirjautuminen estetään, jos käyttäjä ei ole kulkenut määritettyä polkua huoneeseen. Arvolla true käyttäjän luotetaan kulkeneen huoneeseen legitiimiä reittiä pitkin.
ldap	domain	-	str	Toimialue.
	ditSearchBase	-	str	DIT-rakenteen (Directory Information Tree) juuri. Kaikki Kuistin LDAP-operaatioihin liityvät haut suoritetaan tästä juuresta.
	serviceUser	-	str	Kuistin palvelukäyttäjän nimi hakemistopalvelimella.
	roomPrefix	-	str	Huonekohtaisten ryhmien etuliite. Kuisti käyttää tätä etuliitettä tunnistaakseen huonekohtaiset ryhmät hakemistorakenteessa.
	roomDitAttr	-	str	Ryhmäobjektin attribuutti, jossa huoneen nimi sijaitsee. Oletusarvoisesti huoneen nimi sijaistee cn-attribuutissa.
	rolePrefix	-	str	Roolikohtaisten ryhmien etuliite. Kuisti käyttää tätä etuliitettä tunnistaakseen roolikohtaiset ryhmät hakemistorakenteessa.
	roleDitAttr	-	str	Ryhmäobjektin attribuutti, jossa roolin nimi sijaitsee. Oletusarvoisesti roolin nimi sijaistee cn-attribuutissa.
	userDitAttr	-	str	Käyttäjäobjektin attribuutti, jossa käyttäjän uniikki tunniste sijaitsee. Oletusarvoisesti käyttäjät tunnistetaan UPN-arvon perusteella.
firewalls	PALOMUURIN_NIMI (oletusarvoisesti fw01)	ipAddress	str	Palomuurin IP-osoite.
		apiKey	str	Palomuurin hallintaan tarvittava API-avain.
		apiSecret	str	Palomuurin hallintaan tarvittava API-salaisuus.
		filtersetsPath	str	Polku suodatussääntöjen konfiguraatioon.
networks	HUONEEN_NIMI	-	str	Avain-arvo-parin avaimena toimii huoneen nimi ja arvona huoneen käyttämä IP-osoiteavaruus.
routes	HUONEEN_NIMI	-	list[str]	Avain-arvo-parin avaimena toimii huoneen nimi ja arvona lista, joka sisältää huoneeseen kulkevan reitin. Huoneet tulee järjestellä listaan reitin etenemisjärjestyksessä. Kaikille työasemia sisältäville huoneille tulee olla määritettynä reitti, ja reitin viimeisenä huoneena tulee aina olla avaimeksi määritetty huoneen nimi.
roomTimeouts	HUONEEN_NIMI		int	Avain-arvo-parin avaimena toimii huoneen nimi ja arvona huoneelle määritetty aikakatkaisu minuuteissa.

filtersets.json

PÄÄOSIO	ALAOSIOT			ARVON TYYPPI	SELITE
ROOLIN_NIMI (oletusarvoisesti default)	filters	list[dict]	action	str	Säännön suorittama toimenpide. Arvo voi olla pass, block tai reject.
			dstAddr	str	Säännön kohdeosoite. Kohdeosoite voi olla DNS-nimi tai IP-osoite. Hyväksyy myös *-jokerimerkin.
			dstPort	str	Säännön porttinumero. Hyväksyy myös *-jokerimerkin.
			protocol	str	Säännön protokolla. Hyväksyy kaikki protokollat, jota käytettävä palomuurituote tukee.
			ipVersion	str	Säännön käyttämän IP-protokollan versio.
			sequence	int	Säännön sekvenssinumero.
	monitoredServices	PALVELUN_NIMI_TAI_IP-OSOITE	-	int	Avain-arvo-parin avaimena toimii palvelun DNS-nimi tai IP-osoite ja arvona palvelun porttinumero. Tätä konfiguraatiota käytetään suodatussääntöjen käytön valvonnassa; Jos suodatussääntöä ei tarkastushetkellä käytetä mihinkään monitoroituun palveluun, niin sääntö poistetaan palomuurilta. Jos suodatussääntöä käytetään johonkin monitoroituun palveluun, sääntöön linkitetyn aikakatkaisun arvo uusitaan. Toimii ainoastaan silloin, jos timeout- ja renewalAmount-arvot ovat määritettyinä.
	timeout	-	-	int	Säännöille määritetty aikakatkaisu minuuteissa. Jos aikakatkaisuksi määritetään 0, aikakatkaisu on deaktivoitu.
	renewalAmount	-	-	int	Suodatussääntöön linkitetyn aikakatkaisun uusinnan maksimimäärä. Jos arvoksi määritetään 0, sääntö on voimassa timeout-arvoon määritetyn ajan verran.

log_detection.json

PÄÄOSIO	ALAOSIOT			ARVON TYYPPI	SELITE
ULKOISEN_JÄRJESTELMÄN_NIMI	detection	user	regexp	str	Regexp-lauseke, joka hakee tunnistettavan identiteetin nimen ulkoisen järjestelmän lokimerkinnästä.
			matchInGroup	int	Regexp-lausekkeen ryhmän numero, jossa identiteetin nimi sijaitsee.
		directionIn	regexp	str	Regexp-lauseke, joka tunnistaa saapuvan liikenteen (ingress).
			matchInGroup	int	Regexp-lausekkeen ryhmän numero, jossa tunnistetun kulkusuunnan merkkijono sijaitsee.
		directionOut	regexp	str	Regexp-lauseke, joka tunnistaa lähtevän liikenteen (egress).
			matchInGroup	int	Regexp-lausekkeen ryhmän numero, jossa tunnistetun kulkusuunnan merkkijono sijaitsee.
		room	regexp	str	Regexp-lauseke, joka tunnistaa alueen nimen ulkoisen järjestelmän lokimerkinnästä.
			matchInGroup	int	Regexp-lausekkeen ryhmän numero, jossa alueen nimi sijaitsee.
	formatting	user	pattern	str	Regexp-lauseke, joka tunnistaa identiteetin tunnisteen komponentit detection-lausekkeen kaappaamasta merkkijonosta, esim. etu- ja sukunimet.
			repl	str	Regexp-lauseke, joka formatoi tunnistetut komponentit haluttuun muotoon uniikin tunnisteen muodostamiseksi.
		room	pattern	str	Regexp-lauseke, joka tunnistaa alueen tunnisteen komponentit detection-lausekkeen kaappaamasta merkkijonosta, esim. huoneen nimen.
			repl	str	Regexp-lauseke, joka formatoi tunnistetut komponentit haluttuun muotoon uniikin tunnisteen muodostamiseksi.