Toni Hämäläinen

WINDOWS 10 KÄYTTÖJÄRJESTELMÄN KOVENTAMINEN RYHMÄPOLITIIKALLA

JYVÄSKYLÄN YLIOPISTO
INFORMAATIOTEKNOLOGIAN TIEDEKUNTA
2023

TIIVISTELMÄ

Hämäläinen, Toni
Windows 10 käyttöjärjestelmän koventaminen ryhmäpolitiikalla
Jyväskylä: Jyväskylän yliopisto, 2023, 46 s.
Kyberturvallisuus, pro gradu -tutkielma
Ohjaaja: Frantti, Tapio

Tämän tutkimuksen tavoitteena oli tarkastella Windows 10-käyttöjärjestelmällä toimivien työasemien koventamista ryhmäpolitiikan avulla. Windows 10 on yksi suosituimmista käytössä olevista käyttöjärjestelmistä, ja kyberturvallisuusuhkien tullessa yhä vahvemmin osaksi niin organisaatioiden kuin yksityishenkilöidenkin arkea, tutkimus tarjoaa tietoa siitä, miten ryhmäpolitiikan avulla voidaan suojata kyseistä käyttöjärjestelmää käyttäviä työasemia yksinkertaisella, toimivalla ja skaalautuvalla tavalla.

Tutkimuksen teoreettinen tausta koostuu tunnettujen haavoittuvuuksien ja hyökkäysmallien, Windows 10 -käyttöjärjestelmän tietoturvan ja ryhmäpolitiikalla koventamisen, sekä haavoittuvuusskannereiden tarkastelusta. Tämän teoreettisen taustan tukemana tutkimus toteutettiin tapaustutkimuksena, jossa Windows 10 -käyttöjärjestelmä kovennettiin valmiin ryhmäpolitiikan avulla. Käyttöjärjestelmän haavoittuvuudet skannattiin viiden eri skannerin avulla sekä ennen että jälkeen koventamisen, ja koventamisen merkitystä arvioitiin haavoittuvuusskannereiden tuottamien raporttien perusteella.

Tutkimuksen tulokset osoittivat, että ryhmäpolitiikan avulla toteutettu koventaminen vähensi skannereiden havaitsemien haavoittuvuuksien määrää, mutta eri skannereiden välillä oli selviä eroja sekä havaittujen haavoittuvuuksien määrässä, sijainnissa, että määrittelytavassa. Sen lisäksi, että tutkimus osoittaa ryhmäpolitiikan avulla toteutettavan koventamisen parantavan Windows 10-käyttöjärjestelmän tietoturvaa, tutkimus tarjoaa myös havaintoja siitä, millainen Windows 10 -käyttöjärjestelmän turvallisuuden taso yleisellä tasolla on, ja millaisia tuloksia erilaiset haavoittuvuusskannerit antavat käyttöjärjestelmän turvallisuudesta. Tutkimus vahvistaa ymmärrystä valmiiden ryhmäpolitiikkojen hyödyllisyydestä käyttöjärjestelmien koventamisessa, sekä korostaa useiden haavoittuvuusskannereiden käytön merkitystä haavoittuvuuksien tunnistamisessa. Tutkimuksen tuloksia voidaan hyödyntää organisaatioissa käyttöjärjestelmän haavoittuvuuksien tunnistamiseen ja ehkäisemiseen käytettävien työkalujen ja toimintatapojen suunnittelussa.

Asiasanat: Windows 10, Käyttöjärjestelmä, haavoittuvuus, haavoittuvuusskanneri, koventaminen

ABSTRACT

Hämäläinen, Toni
Windows 10 operating system hardening with group policy
Jyväskylä: University of Jyväskylä, 2023, 46 pp.
Cyber Security, Master's Thesis
Supervisor: Frantti, Tapio

The goal of this study was to inspect the hardening of Windows 10 operating system with group policy. Windows 10 is one of the most used operating systems and as cyber threats become more prevalent to organisations and individuals this study offers knowledge about how to leverage group policy in the process of hardening Windows 10 operating system in simple, efficient, and scalable way.

The theoretical background of this study consists of known vulnerabilities, attack models, the security of Windows 10, hardening with group policy, and inspection of vulnerability scanners. With the support of this theoretical background the study was implemented as a case study where Windows 10 operating system was hardened with a preconfigured group policy. The vulnerabilities related to operating system were discovered by using five different vulnerability scanners. The scans were run before and after the hardening of the operating system. The success of hardening was evaluated through the vulnerability reports that were provided by the scanners.

The results show that hardening with preconfigured group policy did lessen the number of detected vulnerabilities. The was a big variety in the number of detected vulnerabilities between scanners as well as in the type of detected vulnerabilities and in the classification of these vulnerabilities. In addition to the result that the hardening enhances the security of Windows 10, this study also provides insights about the level of security of the operating system in general and what kind of results different vulnerability scanners provide related to the operating system. The study verifies that preconfigured group policies are beneficial in the process of hardening the operating system and highlights the fact that multiple scanners should be used to detect vulnerabilities more comprehensively.
These results can be leveraged in organizations to help to plan what tools and procedures to use when detecting and preventing vulnerabilities.

Keywords: Windows 10, Operating system, vulnerability, vulnerability scanner, hardening

TAULUKOT

SISÄLLYS

1 JOHDANTO

Maailmanlaajuisesti kiristynyt poliittinen ilmapiiri ja kyberhyökkäysten kohonnut määrä tekevät kyberturvallisuuden parantamisesta ajankohtaisen aiheen, sillä kyberuhkien yleistymisen myötä tarvitaan lisää resursseja ja ohjeita kyberturvallisuuden parantamiseen. Tämän tutkielman tavoitteena on tarjota yksinkertainen, toimiva ja skaalautuva tapa suojata Windows työasemia ryhmäpolitiikan avulla.

Tutkimus keskittyy Windows 10-käyttöjärjestelmällä toimivien työasemien koventamiseen hyödyntämällä Windowsin ryhmäpolitiikkaa (eng. Group policy). Ryhmäpolitiikan avulla käyttöjärjestelmän asetuksia voidaan muuttaa kattavasti ja tehokkaasti. Tutkimuksessa toteutetaan käyttöjärjestelmän koventaminen hyödyntämällä valmista ryhmäpolitiikkapohjaa, jonka keskiössä on työaseman turvallisuus. Koventamisen toimivuus testataan suorittamalla haavoittuvuusskannaus ennen ja jälkeen käyttöjärjestelmän koventamisen viidellä eri haavoittuvuusskannerilla.

Työasemien koventaminen on haastava prosessi koventamisen laajuuden vuoksi. Ryhmäpolitiikkaa hyödynnetään tyypillisesti työasemien asetuksien muuttamiseksi organisaatioille sopivammaksi, mutta ryhmäpolitiikalla voidaan muuttaa myös työasemien turvallisuusasetuksia. Tätä varten on kehitetty myös valmiita ryhmäpolitiikkapohjia muun muassa CIS:in (Center of Internet Security) toimesta. Tämä tutkimus pyrkii todentamaan ryhmäpolitiikkapohjan toimivuuden ja täten tarjoamaan helpon ja kattavan tavan koventaa Windows 10 -käyttöjärjestelmä. Lisäksi arvioidaan Windows 10 -käyttöjärjestelmän turvallisuutta. Tutkimuksessa pyritään arvioimaan koventamisen toimivuutta hyödyntämällä viittä eri haavoittuvuusskanneria koventamisen toimivuuden todentamiseksi. Tutkimus täten avaa myös haavoittuvuusskannereiden eroavaisuuksia ja niiden hyödyntämistä koventamisen toimivuuden todentamisessa.

Tutkimuksessa huomattiin, että Windows 10 -käyttöjärjestelmästä ei löydy päivittämisen jälkeen kriittisiä (CVSS-pisteytykseltään 9.0–10.0) haavoittuvuuksia.

Käyttöjärjestelmästä havaittiin ennen koventamista kaksitoista haavoittuvuutta, joista kolme haavoittuvuutta oli korkean tason haavoittuvuuksia. Työaseman koventamisen jälkeen haavoittuvuuksien määrä laski seitsemään, mutta korkean tason haavoittuvuuksien määrä pysyi samana. Koventamisen jälkeen haavoittuvuusskannereiden havaitsemat konfiguraatiovirheet vähenivät. Tutkimuksessa havaittiin myös, että eri haavoittuvuusskannerit havaitsivat haavoittuvuuksia vaihtelevasti. Myös haavoittuvuuksien raportoinnissa oli suuria eroja haavoittuvuusskannereiden välillä. Haavoittuvuusskannereilla pystyttiin todentamaan koventamisen onnistuminen, vaikka kaikki haavoittuvuudet eivät korjaantuneetkaan koventamisen myötä. Sopivien haavoittuvuusskannereiden valitseminen tähän tehtävään on hyvin olennaista ja siksi haavoittuvuusskannereiden käyttö koventamisen onnistumisen mittaamisessa vaatii jatkotutkimusta. Ryhmäpolitiikka vaikuttaa olevan toimiva tapa koventaa työasema, mutta politiikka tulee räätälöidä organisaation kontekstiin, jotta työaseman käytettävyys ei kärsi liikaa. Sen lisäksi, että tutkimus tarjoaa tietoa työaseman koventamisesta ryhmäpolitiikan avulla, se tuottaa tietoa Windows 10 -käyttöjärjestelmän haavoittuvuuksista tilanteessa, jossa käyttöjärjestelmä on vasta asennettu ja päivitetty. Näin ollen tutkimuksen avulla voidaan arvioida, kuinka turvallinen käyttöjärjestelmä on kyseessä.

Tämän tutkimuksen rakenne koostuu kirjallisuuskatsauksesta, tutkimuksen aineiston, analyysin ja tulosten esittelystä, sekä pohdintaosiosta, jossa tarkastellaan tutkimuksen tuloksia ja niistä tehtyjä johtopäätöksiä, sekä tutkimuksen rajoitteita ja jatkotutkimushaasteita. Tutkimuksen seuraavassa luvussa tutustutaan tarkemmin aihealueen kirjallisuuteen ja keskeisiin käsitteisiin. Luvussa tarkastellaan haavoittuvuusskannereita, niiden toimintaperiaatteita, ominaisuuksia, käyttökohteita, sekä skannereiden välisiä eroja. Tämän lisäksi tutustutaan haavoittuvuuksiin, niiden määrittelemiseen, mittaamiseen, vertailemiseen, sekä hyväksikäyttöön. Tässä luvussa avataan myös tarkemmin Windows 10 -käyttöjärjestelmän turvallisuutta, käyttöjärjestelmän turvallisuuden kehitystä, sekä käyttöjärjestelmän turvallisuuteen liittyviä tutkimustuloksia. Luvun yhtenä aiheena on myös käyttöjärjestelmän koventaminen, jonka osalta tutustutaan aihealueen tutkimuksiin, sekä erilaisiin tapoihin ja työkaluihin, joilla käyttöjärjestelmää voidaan koventaa.

Teoreettisen taustan jälkeen Tutkimusmenetelmä-luvussa määritellään tutkielman tutkimuskysymykset, sekä määritellään tutkimuksessa tehdyt rajaukset. Luvussa esitellään tutkimusmetodi, sekä tutkimuksen aineisto. Neljäntenä, omassa luvussaan, esitellään Windows 10:n koventaminen, jossa käydään käytännön tasolla läpi, kuinka työasema kovennetaan hyödyntämällä valmista ryhmäpolitiikkapohjaa. Viidentenä lukuna on Tulokset, jossa käydään läpi haavoittuvuusskannereiden tuottamia raportteja, havaittuja haavoittuvuuksia, sekä skannereiden välisiä eroavaisuuksia. Viimeisessä luvussa, Pohdinta, analysoidaan saatuja tutkimustuloksia, arvioidaan tutkimuksen rajoitteita, sekä ehdotetaan tulevaisuuden tutkimusaiheita aihealueeseen liittyen.

2 TEOREETTINEN TAUSTA

Tässä luvussa käydään läpi tutkimuksen kannalta olennaista kirjallisuutta. Aiemman kirjallisuuden tuloksia verrataan toisiinsa, sekä pohditaan kuinka aiemmat tulokset sopivat yhteen tämän tutkimuksen kanssa. Kirjallisuuskatsaus koostuu kolmesta alaluvusta, jotka tarkastelevat tutkimukselle keskeisiä aiheita: Windows 10:n tietoturva, Windows 10:n koventaminen, sekä haavoittuvuusskannerit ja niiden käyttäminen.

2.1 Haavoittuvuudet ja niiden määritteleminen

Tässä alaluvussa tutustutaan tarkemmin haavoittuvuuksiin, niiden määritelmiin, standardisoimiseen, sekä haavoittuvuuksien luokittelemiseen.

Haavoittuvuudella tarkoitetaan yleisesti ominaisuutta tai heikkoutta järjestelmässä, jota hyödyntämällä voidaan aiheuttaa vahinkoa, saada informaatiota tai pysäyttää järjestelmä. Softic ja Vejzović (2022) määrittelevät haavoittuvuuden olevan sellainen ominaisuus tai heikkous, jonka hyödyntäminen vaikuttaa negatiivisesti joko luottamuksellisuuteen (eng. Confidentiality), tiedon eheyteen (eng. Integrity), tai saatavuuteen (eng. Availability). Nämä kolme ominaisuutta muodostavat kolmikantaisen CIA-turvallisuusmallin. Myös käyttöjärjestelmä on nimensä mukaisesti järjestelmä, ja siinäkin esiintyy haavoittuvuuksia. Näitä haavoittuvuuksia hyödyntämällä hyökkääjän on mahdollista vaarantaa käyttöjärjestelmä, tai mikä tahansa käyttöjärjestelmään asennettu sovellus tai järjestelmä. Käyttöjärjestelmähaavoittuvuuden voidaan katsoa johtuvan tehdyistä virheistä käyttöjärjestelmän kehitysprosessissa, käyttöjärjestelmän päivittämättä jättämisestä, tai vanhentuneesta käyttöjärjestelmäversiosta. (Softić & Vejzović, 2022.)

Haavoittuvuuksille on määritelty elinkaari. Vaikka käyttöjärjestelmähaavoittuvuuksien elinkaari on yhä tutkimuksen alaisena, tutkijat ovat yleensä samaa mieltä siitä, että tämä elinkaari on jaettu viiteen vaiheeseen (Softić & Vejzović, 2022). Nämä vaiheet ovat seuraavat: haavoittuvuuden syntyminen, haavoittuvuuden havaitseminen, haavoittuvuuden julkistaminen, korjaavan päivityksen tuleminen saataville, sekä korjaavan päivityksen asentaminen (Ruohonen ym. 2016). Gorbenko ym. (2020) ovat ehdottaneet, että elinkaareen lisättäisiin haavoittuvuuden hyväksikäyttövaihe ennen vaihetta "korjaavan päivityksen asentaminen", sillä haavoittuvuutta voidaan hyväksikäyttää ennen korjaavan päivityksen saataville ilmaantumista. Aika, joka eri vaiheiden välillä kuluu, vaihtelee haavoittuvuuden mukaan. Jotkin haavoittuvuudet saattavat olla pitkään olemassa ilman, että niiden olemassaolosta tiedetään. Toiset haavoittuvuudet taas havaitaan nopeasti. Myös aika korjaavan päivityksen ilmaantumisessa vaihtelee, riippuen siitä, miten haavoittuvuus on mahdollista korjata. On myös mahdollista,

että haavoittuvuutta ei voida korjata päivityksellä, vaan haavoittuvuus täytyy joko hyväksyä, tai kiertää.

Haavoittuvuus voi myös sijaita järjestelmässä ilman, että laitevalmistaja on tietoinen haavoittuvuudesta. Tällöin puhutaan nollapäivähaavoittuvuudesta (eng. Zero-day vulnerability). Microsoft (2022) määrittelee nollapäivähaavoittuvuuden olevan järjestelmässä sijaitseva vika tai heikkous, joka on löydetty ennen kuin valmistaja tai ohjelmoija on itse tietoinen haavoittuvuudesta. Koska valmistaja tai ohjelmoija ei ole haavoittuvuudesta tietoinen, ei haavoittuvuuteen löydy myöskään korjaavaa päivitystä (Microsoft, 2022). Myöskään haavoittuvuusskannerit eivät havaitse nollapäivähaavoittuvuuksia, sillä näitä haavoittuvuuksia ei löydy niiden tietokannoista.

Käyttöjärjestelmien haavoittuvuuksiin keskittyviä tutkimuksia on rajallisesti. Gorbenko ym. (2020) tutkivat kuuden eri käyttöjärjestelmän haavoittuvuuksia vuosilta 2012 ja 2016. Yaswinski ym. (2019) tutkivat taasen Linux-käyttöjärjestelmän turvallisuutta sekä sitä, kuinka käyttöjärjestelmän turvallisuutta voidaan parantaa korjaamalla haavoittuvuuksia päivitysten avulla. Näiden tutkimusten lisäksi Softić & Vejzović ovat tehneet kaksi tutkimusta, joista ensimmäinen (2021) keskittyy tutkimaan Windows 10:n, macOS:n ja Ubuntun sisältämiä haavoittuvuuksia, kun taas toinen (2022) tarkastelee ainoastaan Windows 10:n haavoittuvuuksia ja niiden hyväksikäyttöä. Näitä kahta tutkimusta tarkastellaan lähemmin Windows 10:n tietoturvaa käsittelevässä alaluvussa.

2.2 Haavoittuvuustietokannat ja haavoittuvuuksien luokittelu

Haavoittuvuuksia kerätään suuriin tietokantoihin, joista tunnettuja esimerkkejä ovat kansallinen haavoittuvuustietokanta, NVD (eng. National Vulnerabilities Database) ja CVE (eng. Common Vulnerabilities and Exposures system). CVE on voittoa tavoittelemattoman organisaation, MITRE:n ylläpitämä tietokanta, joka listaa tunnetut haavoittuvuudet ja antaa niille yksilöllisen CVE-ID:n. NVD on taas NIST:in (eng. National Institute of Standards and technology) ylläpitämä. NVD luokittelee haavoittuvuuden vakavuuden ja tyypin, jotta haavoittuvuuksia voidaan vertailla keskenään. Haavoittuvuuden vakavuuden luokitteleminen tapahtuu käyttämällä CVSS (eng. Common Vulnerability Scorin System) -pisteytystä. Haavoittuvuuden pisteytykseen vaikuttavat haavoittuvuuden hyödyntämisen helppous, hyökkäysvektori, vaaditut oikeudet, vaatiiko haavoittuvuus käyttäjältä toimenpiteitä, haavoittuvuuden vaikutuksen laajuus, sekä se, kuinka haavoittuvuus vaikuttaa CIA-turvallisuusmalliin. Pisteytyksen perusteella haavoittuvuuden vakavuus luokitellaan tasolle kriittinen, korkea, keskitaso, matala, tai olematon. Myös yritykset ylläpitävät omia haavoittuvuustietokantojaan, joita heidän kehittämänsä haavoittuvuusskannerit hyödyntävät.

Muita pisteytys- tai luokittelujärjestelmiä ovat CPE (eng. Common Platdorm Enumeration Dictionary), sekä CWE (eng. Common Weakness Enumeration Specification). CPE:tä käytetään määrittelemään, mitä osaa, valmistajaa, tuotetta ja versiota haavoittuvuus koskee. CWE:n avulla voidaan luokitella käyttöjärjestelmähaavoittuvuuksia yleisellä tasolla. Esimerkkinä voidaan käyttää CWE-312:ta, joka tarkoittaa sensitiivisen tiedon tallentamista selkokielisenä tekstinä. (Softić & Vejzović, 2022.)

2.3 Kyberturvallisuuden hyökkäysmallit

Tässä alaluvussa esitellään vaiheet, joista kyberhyökkäykset muodostuvat. Eri toimijat ovat luoneet erilaisia hyökkäysketjukuvauksia poikkeavin termein. Näissä kuvauksissa on kuitenkin yhteisiä piirteitä, eri nimeämis- ja kuvauskäytänteistä huolimatta. Kaksi yleisesti tunnettua hyökkäysmallia tai viitekehystä ovat Lockheed Martinin kehittämä Cyber Kill Chain, sekä MITRE:n kehittämä ATT&CK viitekehys. Näitä kahta eri toimijan kehittämää mallia esitellään seuraavissa alaluvuissa tarkemmin.

2.3.1 Lockeed Martin Cyber Kill Chain

Lockheed Martin (2023) on kehittänyt hyökkäysmallin "Cyber Kill Chain" (CKC). Malli koostuu hyökkäyksen seitsemästä eri vaiheesta, jotka ovat:

1. Tiedustelu (engl. Reconnaissance): hyökkääjä määrittää kohteet ja käytetyt taktiikat hyökkäykselle ja suorittaa kattavan tutkinnan siitä, mitä haavoittuvuuksia hyökkäyksessä voidaan hyväksikäyttää.
2. Aseistaminen (engl. Weaponization): hyökkääjä luo tiedustelutiedon pohjalta hyökkäyksen, kuten haittaohjelman, joka hyödyntää löydettyjä haavoittuvuuksia.
3. Toimittaminen (engl. Delivery): hyökkääjä siirtää rakentamansa hyökkäyksen kohteeseen tarkoituksenaan hyväksikäyttää löytämiään haavoittuvuuksia. Toimittaminen voi tapahtua esimerkiksi sähköpostitse tai verkkosivun kautta.
4. Hyväksikäyttö (engl. Exploitation): hyökkääjä hyväksikäyttää järjestelmän haavoittuvuuksia hyökkäyksensä avulla tavoitteenaan saada laajempi pääsy kohdejärjestelmään.
5. Asentaminen (engl. Installation): hyökkääjä asentaa muita ohjelmia järjestelmään, kuten takaovia (engl. Backdoor) tai etäkäyttöisiä komentorivejä järjestelmään. Tämä mahdollistaa hyökkääjän jatkotoimenpiteet, kuten oikeuksien laajentamisen ja turvallisuuskontrollien ohittamisen.

6. Komentaminen ja hallinta (engl. Command and control): hyökkääjä muodostaa kanavan, jonka avulla hänen on mahdollista suorittaa komentoja järjestelmässä ja valvoa sitä etänä.
7. Toimet tavoitteiden saavuttamiseksi (engl. Actions on Objectives): hyökkääjä suorittaa loput vaaditut toimenpiteet saavuttaakseen hyökkäyksen tavoitteen. Tavoitteena voi olla muun muassa datan varastaminen, muokkaaminen, tuhoaminen, tai salaaminen.

Mallin tarkoituksena on havainnollistaa, että hyökkääjä suorittaa samat hyökkäyksen vaiheet, vaikka hyökkäys itsessään olisi erilainen. Tämä antaa hyökkäyksiltä puolustautuville tiimeille mahdollisuuden havaita hyökkääjä hyökkäyksen eri vaiheissa ja auttaa vastatoimijoita ymmärtämään hyökkääjän käyttämiä taktiikoita ja tekniikoita (Lockheed Martin, 2023). Esimerkiksi toimittamisvaiheessa hyökkäys on mahdollista havaita, mikäli uhrin lataama tiedosto skannataan virussuojan toimesta.

2.3.2 MITRE ATT&CK

MITRE ATT&CK on MITRE:n kehittämä viitekehys, jossa kuvataan hyökkääjien hyödyntämiä taktiikoita, proseduureja ja tekniikoita. Nämä kuvaukset perustuvat oikeisiin havaintoihin ja ne kuvaavat hyökkäyksen elinkaaren eri vaiheita (MITRE, 2023). ATT&CK viitekehyksen sisältämät sisäänpääsytekniikat (engl. Intrusion techniques) on jaettu neljääntoista eri taktiikkaan, jotka ovat:

1. Tiedustelu (engl. Reconnaissance): hyökkääjä kerää tietoa, jonka pohjalta suunnittelee hyökkäyksen toteuttamisen.
2. Resurssien kehittäminen (engl. Resource Development): hyökkääjä pyrkii keräämään resursseja, jotka tukevat hyökkäystä.
3. Ensimmäinen pääsy järjestelmään (engl. Initial Access): hyökkääjä koittaa saada pääsyn kohdejärjestelmään.
4. Suorittaminen (engl. Execution): hyökkääjä yrittää suorittaa haitallista koodia kohdejärjestelmässä.
5. Pysyvyys (engl. Persistence): hyökkääjä koittaa säilyttää pysyvän pääsyn järjestelmään ilman, että pääsy katkeaa, vaikka järjestelmä käynnistettäisiin uudestaan, tai tunnuksia päivitettäisiin.
6. Oikeuksien nostaminen (engl. Privilege escalation): hyökkääjä koittaa saada vahvemmat oikeudet kohdejärjestelmässä.
7. Puolustuksen väistäminen (engl. Defence Evasion): hyökkääjä pyrkii välttämään havaituksi tulemista vaarantuneessa järjestelmässä.
8. Pääsy tunnuksiin (engl. Credential Access): hyökkääjä pyrkii varastamaan tunnuksia päästäkseen kohdejärjestelmään laajemmin.
9. Uusien kohteiden löytäminen (engl. Discovery): hyökkääjä koittaa saada lisää tietoa kohdejärjestelmästä ja sen sisäisestä infrastruktuurista.
10. Poikittainen liikkuminen (engl. Lateral Movement): hyökkääjän tavoitteena on päästä sisään sisäisen järjestelmän eri laitteisiin ja järjestelmiin.

11. Tiedon kerääminen (engl. Collection): hyökkääjä kerää tietoa, joka on hyökkäyksen kannalta relevanttia.
12. Komentaminen ja hallinta (engl. Command and Control): hyökkääjä ohjaa kohdejärjestelmää jonkin ulkoisen kanavan kautta.
13. Tiedon varastaminen (engl. Exfiltration): hyökkääjä varastaa dataa vaarantuneesta järjestelmästä.
14. Vaikuttaminen (engl. Impact): hyökkääjä pyrkii vaikuttamaan kohdejärjestelmään manipuloimalla, tuhoamalla, tai häiritsemällä sitä.

Hyökkääjät pyrkivät pääsemään järjestelmään sisään hyödyntämällä järjestelmässä olevia haavoittuvuuksia ja heikkouksia. Hyökkäysmallien, kuten Cyber Kill Chain:in ja ATT&CK:n avulla voidaan identifioida eri vaiheita, joissa hyökkäys voidaan huomata. Tämän lisäksi mallit tarjoavat tietoa hyökkäyksissä käytetyistä taktiikoista. Myös tämän tiedon pohjalta hyökkäyksiä voidaan havaita ja estää. Kuten hyökkäysmalleista voidaan havaita, haavoittuvuuksia on mahdollista hyödyntää useissa hyökkäyksen eri vaiheissa. Tämän vuoksi myös järjestelmien puolustusta kehittävät tahot käyttävät usein hyökkääjien suosimia metodeja testatessaan järjestelmän puolustuskykyä. Järjestelmää koventaessa sitä voidaan skannata esimerkiksi Nmapilla puolustajien toimesta, jotta he näkevät, mitä hyökkääjät näkevät järjestelmästä sen ulkopuolelta. Vastaavasti haavoittuvuusskannereilla pyritään havaitsemaan järjestelmän heikkoja lenkkejä, jotta ne voidaan korjata ennen kuin hyökkääjä ehtii hyväksikäyttämään niitä. Hyökkäysmalleissa otetaan huomioon, että vaikka jokin vaihe hyökkäyksestä jäisi huomaamatta, hyökkäys voidaan havaita myös sen myöhäisemmässä vaiheessa. Näin ollen hyökkäysmallit ovat äärimmäisen hyödyllisiä järjestelmän koventamisen ja puolustamisen kannalta, sillä ne vakioivat koventamisprosessia ja tarjoavat näkyvyyttä siihen, mitä kaikkea koventamisessa tulee ottaa huomioon.

2.4 Windows 10:n tietoturva

Käyttöjärjestelmiä laajennetaan lisäämällä erilaisia ominaisuuksia, jotta käyttöjärjestelmä vastaisi paremmin käyttäjien toiveita ja vaatimuksia. Tämän kääntöpuolena on kuitenkin se, että käyttöjärjestelmän hyökkäyspinta-ala, sekä haavoittuvuuksien määrä kasvaa lisättävien ominaisuuksien mukana. Haasteena käyttöjärjestelmien kehittäjillä onkin, kuinka luoda erinomainen käyttäjäkokemus, samalla suojaten käyttöjärjestelmä tehokkaasti (Softic & Vejzovic, 2022). Käyttöjärjestelmää pyritään kehittämään päivitysten avulla.

Windows 10-käyttöjärjestelmän osalta päivityksistä vastaa Microsoft Update-palvelu. Se tarjoaa listauksen Microsoftin ohjelmien päivityksistä, ajureista (eng. Drivers), sekä nopeista korjauksista (eng. Hotfix). Päivitykset jaetaan kolmeen luokkaan: tärkeät, suositellut ja valinnaiset päivitykset. Tärkeät päivitykset tarjoavat parempaa turvallisuutta ja luotettavuutta, suositellut päivitykset parantavat käyttöjärjestelmän suorituskykyä, kun taas valinnaiset päivitykset voivat

sisältää esimerkiksi uusia tai päivitettyjä ajureita tietylle laitteelle. Microsoft tarjoaa myös palvelupakettiin (eng. Service pack, SP) liittyviä päivityksiä. Nämä testatut palvelupaketit koostuvat nopeista korjauksista, turvallisuuspäivityksistä, sekä kriittisistä päivityksistä. Yksittäinen turvallisuuspäivitys kattaa usein useita tunnettuja haavoittuvuuksia ja päivityksen korjaamat haavoittuvuudet listataan Microsoftin päivityskohtaisessa turvallisuustiedotteessa (eng. Security bulletin). (Badawy ym., 2013)

Käyttöjärjestelmän koventamisen ohella voidaan käyttää myös muita tapoja suojautua hyökkäyksiltä, esimerkiksi säännöllisiä haavoittuvuusarviointeja, sekä haavoittuvuus- ja penetraatiotestauksia (Softic & Vejzovic, 2022). Microsoft on pyrkinyt parantamaan Windows 10:n turvallisuutta kehittämällä Windows-puolustuskeskuksen (eng. Windows Defender Centre, WDC), joka parantaa järjestelmän turvallisuutta havaitsemalla turvallisuusuhat tapahtumalokien optimoinnilla (Baráth, 2017). Windows-puolustuskeskus tarjoaa käyttäjälle näkyvyyttä ja mahdollisuuden hallita laitteen turvallisuutta viiden eri osa-alueen kautta: virukset ja uhkien torjunta, laitteen suorituskyky ja kunto, palomuuri ja verkon suojaaminen, sovellus- ja selainhallinta, sekä perheasetukset (Softic & Vejzovic, 2022). Sovellushallinnan työkaluihin lukeutuu muun muassa Windowsiin sisäänrakennettu AppLocker. AppLockerin avulla voidaan määrittää, mitä sovelluksia työasemalla saa suorittaa. Sallittuja sovelluksia voidaan lisätä järjestelmänvalvojan toimesta. Näin voidaan estää vieraiden ja mahdollisesti haitallisten sovellusten suorittaminen työasemalla. Palomuurin ja verkon suojaamisella voidaan määrittää muun muassa, mitä portteja työasemalla on auki. Tämä on olennainen osa hyökkäyspinta-alan pienentämistä, ja täten työaseman koventamista. Windows 10 tarjoaa kattavasti erilaisia asetusvaihtoehtoja, joiden avulla turvallisuutta voidaan parantaa.

Softic ja Vejzovic (2022) tutkivat Windows 10:n haavoittuvuuksia hyödyntämällä erilaisia hyökkääjien käyttämiä työkaluja, kuten Metasploit-viitekehystä, Nmapia, msfvenomia, sekä Netcat-työkalua. Tavoitteena tutkijoilla oli löytää heikkoja kohtia vasta asennetusta Windows 10-käyttöjärjestelmästä. Tutkijat halusivat myös todistaa, että käyttöjärjestelmän oletusasetukset eivät pysty täysin suojaamaan järjestelmää kyberhyökkäyksiltä. Tämän lisäksi tutkijat halusivat ymmärtää todennäköisiä hyökkäysskenaarioita paremmin. Tutkimuksessa käyttöjärjestelmään ei asennettu mitään ulkopuolisia sovelluksia, joten tutkimus antaa hyvän kuvan siitä, mitä haavoittuvuuksia itse käyttöjärjestelmästä löytyy. Tutkimuksessa testattiin Windows 10:n kolmea eri versiota: Pro, Education ja Enterprise. Virtuaalikoneille, joilla käyttöjärjestelmät toimivat, ei asennettu kolmannen osapuolen sovelluksia, tai päivityksiä. Tutkijat skannasivat käyttöjärjestelmät Nmapilla ja Nessus Pro:lla. Nmap ei havainnut käyttöjärjestelmissä haavoittuvuuksia, kun taas Nessus havaitsi yhden keskitason haavoittuvuuden liittyen SMB-kirjautumiseen (Server Message Block), sekä 37 informaatiotason haavoittuvuutta.

Tutkimuksessa havaittiin, ettei Windows 10:stä löydy paljonkaan käyttöjärjestelmätason haavoittuvuuksia, kun siihen ei asenneta kolmannen osapuolen sovelluksia. Tutkijat suorittivat käyttöjärjestelmillä haitallisia tiedostoja, joissa kaksi hyökkäystä kahdeksasta onnistuivat vaarantamaan käyttöjärjestelmän. Huomionarvoista on se, että käyttöjärjestelmäversioista Pro-versio ei vaarantunut millään hyökkäyksellä, toisin kuin Education- ja Enterprise-versiot. Tutkijat olivat sammuttaneet palomuurin sekä virussuojan käyttöjärjestelmistä. Tutkimus osoittaa, että käyttöjärjestelmästä itsestään ei löydy suurta määrää haavoittuvuuksia. On kuitenkin hyvä huomata, että tutkimuksessa käytettiin ainoastaan kahta haavoittuvuusskanneria. Myös tutkijat kehottavat laajentamaan tutkimusta hyödyntämällä erilaisia työkaluja. Käyttöjärjestelmään voi kohdistua myös nollapäivähaavoittuvuuksia (eng. Zero-day vulnerability), joita skannerit eivät pysty löytämään. Tutkijoiden lupaavat tulokset eivät siis tarkoita, että käyttöjärjestelmä on täysin turvallinen, vaan että käyttöjärjestelmästä ei löydy montaa tunnettua haavoittuvuutta.

Toisessa tutkimuksessaan Softic ja & Vejzovic (2021) tutkivat Windows 10:n, MacOS:n ja Ubuntu:n käyttöjärjestelmien haavoittuvuuksia vuodesta 2015 aina vuoteen 2021 asti. Tarkoituksena tutkijoilla oli selvittää, kuinka käyttöjärjestelmien haavoittuvuuksien määrä on kehittynyt samalla kun kyberhyökkäykset ja kybersodat ovat yleistyneet ennen ja jälkeen COVID-19-pandemian. Softic ja & Vejzovic (2021) hyödynsivät kahta tietokantaa, MITREn ylläpitämää CVE-tietokantaa, sekä NIST:in (eng. National Institure of Standards and Technology) ylläpitämää NVD-tietokantaa. Tutkimuksessa käytettiin myös CWE-määritelmää (eng. Common Weakness Enumeration), jota käytetään käyttöjärjestelmien haavoittuvuuksien luokittelemiseen, esittelyyn, ja käsittelyyn. Vuosien 2015 ja 2021 välillä Windows 10:stä on löydetty yhteensä 2 489 haavoittuvuutta. Ubuntusta löytyi 2 660 haavoittuvuutta ja MacOS:sta 1 972. Viisi yleisintä haavoittuvuutta olivat koodin suorittaminen (eng. Code execution), palvelunestohyökkäys (eng. DoS attack), ylivuotohaavoittuvuus (eng. Overflow), muistin korruptoiminen (eng. Memory corruption) ja tiedon hankkiminen (eng. Gaining information). Näistä Windows 10:ssä kolme yleisintä haavoittuvuutta olivat koodin suorittaminen, tiedon hankkiminen, sekä oikeuksien saaminen (eng. Gaining privileges). Windows 10:ssä haavoittuvuudet ovat lisääntyneet vuodesta 2015 lähtien. Haavoittuvuuksia julkaistiin eniten vuonna 2020, jolloin julkaistiin 807 Windows 10:tä koskevaa haavoittuvuutta. Vuonna 2021 haavoittuvuuksien määrä laski tästä noin puoleen, ollen 485. Kun haavoittuvuuksille laskettiin painotettu keskiarvo jokaista vuotta kohden, tutkijat havaitsivat, että Windows 10:llä oli korkein painotettu keskiarvo koko tarkastelujaksolta, kun taas Ubuntulla oli matalin.

Käyttöjärjestelmien haavoittuvuudet luokitellaan hyödyntämällä CWE:tä. Windows 10:lle yleisimmät CWE:t olivat CWE-119, CWE-19, CWE-20 ja CWE-281. CWE-119 tarkoittaa, että ohjelma pääsee lukemaan myös muille ohjelmille varattuja muistialueita, eli kyseessä on epäonnistunut muistin rajoittaminen sovellukselle. CWE-19 tarkoittaa haavoittuvuutta datan prosessoimisessa. CWE-20 on

epäonnistunut syötteen tarkastaminen (eng. Improper input validation), joka voi johtaa esimerkiksi pahantahtoisen koodin suorittamiseen. CWE-281 on sovelluksen oikeuksien säilyttämisen epäonnistuminen, joka voi johtaa korotettuihin oikeuksiin käyttöjärjestelmässä ilman perusteita. (Softic ja & Vejzovic, 2021)

Tutkijat toteavat kyberhyökkäysten kasvaneen määrän näkyvän myös käyttöjärjestelmähaavoittuvuuksien määrässä. Käyttöjärjestelmiä käytetään hyökkäysten kohteena niiden sisältämien haavoittuvuuksien takia. Windows 10 oli yleisin kohde kolmesta käyttöjärjestelmästä. (Softic & Vejzovic, 2021.) Ottaen huomioon käyttöjärjestelmän mittavan käyttäjäkunnan, tämä ei ole yllättävää, mutta perustelee sitä, miksi Windows 10 -käyttöjärjestelmään kohdistuvaa tutkimusta on syytä tehdä. Näin ollen tässäkin tutkimuksessa pyritään selvittämään juuri Windows 10:n koventamisen vaikutuksia.

2.5 Windows 10:n koventaminen

Kuten todettu, Windows 10 on toiminut tutkimusten kohteena jo pitkään, sillä se on käyttöjärjestelmänä suosittu. Myös Windows 10:n koventamiseen liittyvää tutkimusta on tehty. Durve ja Bouridane (2017) tutkivat Windows 10:n koventamista käyttämällä Device Guardia ja AppLockeria. Device Guard:in avulla voidaan sallia tiettyjen ohjelmistojulkaisijoiden sovellusten ajaminen, sekä varmistaa ajurien ja kernel-koodin integriteetti. Device Guard on myöhemmin korvattu Microsoftin toimesta WDAC:illa (Windows Defender Application Control). AppLocker keskittyy sallittujen sovellusten määrittämiseen. Molempia työkaluja voidaan konfiguroida ryhmäpolitiikan avulla.

Tutkijat pyrkivät hyödyntämään sekä sallittujen sovellusten listaa (eng. Whitelist), sekä kiellettyjen sovellusten listaa (eng. Blacklist). Näiden listojen avulla on mahdollista pyrkiä estämään niin haitalliset sovellukset, kuin sallittujen sovellusten väärinkäyttäminen hyödyntämällä esimerkiksi sovelluksen käyttämää ohjelmakoodinkääntäjää (eng. Interpreter). Joissakin tapauksissa ohjelmakoodinkääntäjää ei voida kieltää sen kriittisyyden takia. Esimerkiksi Excelin tarvitsee usein suorittaa makroja. Näissä tapauksissa makrojen tulisi olla allekirjoitettuja luotettujen julkaisijoiden toimesta. Näin voidaan estää vieraiden makrojen suorittaminen. (Durve & Bouridane, 2017.)

Tutkijat suorittivat työasemalla .com- ja .exe-päätteisiä ajettavia tiedostoja, jotka matkivat haittaohjelmien käytöstä. Tiedostoja myös muutettiin, jotta niiden havaitseminen vaikeutuisi. Tutkijat halusivat selvittää, kuinka hyvin AppLocker ja DeviceGuard pystyvät estämään sovellusten suorittamisen.
Tutkimustulokset osoittivat, että konfiguroimalla AppLocker ja DeviceGuard huolellisesti, haitalliset tiedostot oli mahdollista estää kokonaan. Tämä johtui siitä, että haitallisia sovelluksia ei allekirjoitettu hyväksyttyjen toimittajien toimesta, eikä niitä sallittu erikseen. Hyvistä tuloksista huolimatta tutkijat

huomauttavat, että heidän lähestymistapansa ei estä ohjelmistohaavoittuvuuksien hyödyntämistä, eikä ainoastaan muistissa sijaitsevien haitallisten ohjelmien suorittamista. Mikäli hyökkääjän onnistuu kaapata sertifikaatteja myöntävä auktoriteetti (eng. Certificate signing authority), hyökkääjä pystyy ohittamaan tämän suojaustavan kokonaan. (Durve & Bouridane, 2017.)

Tutkimus antaa rohkaisevia tuloksia siitä, että työkaluilla, kuten AppLocker ja DeviceGuard, pystytään tehokkaasti estämään vieraiden ja mahdollisesti haitallisten ohjelmien ja koodin suorittaminen. Vastaava sovellusten ja ohjelmakoodin erikseen salliminen ja allekirjoittaminen on kuitenkin hyvin työläs ja kallis prosessi, kuten tutkijat Durve ja Bouridanekin (2017) toteavat. Erityisesti suurissa organisaatioissa voi olla hyvin haastavaa käydä läpi kaikki sovellukset ja sallittavat ohjelmakoodit, jotta tämä lähestymistapa olisi täysin kattava. Toinen heikkous, jonka tutkijatkin nostivat esiin, ovat sovellusten sisältämät haavoittuvuudet. Näitä ei tietenkään voida koskaan täysin estää, mutta hyökkäyspinta-alaa voidaan pienentää tehokkaalla päivittämisellä. AppLocker ja DeviceGuard eivät tarjoa yksinään täydellistä suojaa, mutta tarjoavat parannettua turvallisuutta omalla osa-alueellaan, täten parantaen käyttöjärjestelmän ja työaseman turvallisuutta.

Ryhmäpolitiikalla voidaan määrittää Windows-käyttöjärjestelmän asetuksia hyvin kattavasti. Työkalujen, kuten AppLockerin ja DeviceGuardin, ohella voidaan määrittää muun muassa työaseman lokittamisesta. Työaseman koventaminen on usein preventatiivista ja proaktiivista, mutta myös reaktiivista ja havaitsevaa suojaamista tarvitaan. Työasemalokien avulla voidaan havaita pahantahtoista toimintaa. Baráth (2017) tutki Windows 10:n lokituksen, eli työasematapahtumien tallentamisen optimointia Microsoft Security Compliance Manager (SCM) -työkalun avulla. SCM sisältää valmiita turvallisuuspohjia (eng. security templates), joiden avulla työaseman asetuksia voidaan muuttaa. Tutkimus keskittyi turvallisuusuhkien havaitsemiseen verkkokommunikoinnissa, vaikkakaan käyttöjärjestelmän lokittaminen ei rajoitu pelkästään tähän, vaan työasemalta voidaan lokittaa tapahtumia hyvin monipuolisesti. Turvallisuuden kannalta hyödyllisiä ja kiinnostavia tapahtumia ovat esimerkiksi kirjautumistapahtumat ja etäyhteystapahtumat.

Baráth (2017) toteaa tutkimuksessaan, että käyttöjärjestelmän lokittamisen oletusasetukset eivät ole riittävän kattavat turvallisuusuhkien havaitsemiseen. Hyödyntämällä SCM:n turvallisuuspohjia työasematapahtumien lokituksesta saatiin kattavampi. Tämä parantaa turvallisuusuhkien havaitsemisen mahdollisuutta. Baráth (2017) kuitenkin toteaa, että valmiin pohjan hyödyntäminen vaatii myös optimointia, sillä tallennettujen tapahtumien määrä kasvaa nopeasti, kun käytetään SCM:n turvallisuuspohjaa. Vaikka tässä tutkimuksessa ei varsinaisesti keskitytä työasematapahtumien lokittamisen kehittämiseen, on hyvä huomioida, että tapahtumien kerääminen ja tallentaminen on olennainen osa organisaation

ja työaseman turvallisuuden parantamista, sillä se mahdollistaa tietoturvauhkien havaitsemisen, kun muut työkalut eivät niitä havaitse.

2.5.1 Koventaminen ryhmäpolitiikalla

Kuten aiemmissa kappaleissa on mainittu, ryhmäpolitiikalla voidaan muuttaa lähes kaikkia työaseman asetuksia. Ryhmäpolitiikalla on tärkeä rooli myös Windows-palvelimien, sekä työasemien koventamisen suhteen. Moskowitz (2012, s. 447) luettelee kirjassaan koventamiskohteiksi muun muassa:

- Salasanapolitiikan (engl. Password policy). Salasanapolitiikka pitää sisällään useita sääntöjä, kuten esimerkiksi sen, kuinka pitkä salasanan tulee olla, montako aiempaa salasanaa järjestelmä muistaa käyttäjältä, sekä sen, vanheneeko salasana.
- Sovellusten rajoituspolitiikan, sekä AppLockerin. Tämä tarkoittaa, että ryhmäpolitiikalla määritellään mitä sovelluksia työasemalla tai palvelimella saadaan suorittaa.
- Windows-palomuurin, sekä edistyneemmät turvallisuusasetukset. Näihin lukeutuvat palomuurin asetukset, sekä muita edistyneempiä ja yksityiskohtaisempia asetuksia, joita voidaan muuttaa ryhmäpolitiikan avulla.

Koska ryhmäpolitiikalla koventaminen on hyvin työläs prosessi, organisaatiot ovat kehittäneet valmiita ryhmäpolitiikkapohjia, joiden avulla prosessia voidaan helpottaa. Nämä valmiit pohjat eivät useimmiten kuitenkaan sovi suoraan organisaatiolle käyttöönotettavaksi, vaan ne tulee testata ja muokata omiin tarpeisiin sopivaksi.

Center for Internet Security (CIS) on yksi organisaatioita, jotka kehittävät kyberturvallisuutta parantavia ratkaisuja. CIS tuottaa ohjeita siitä, kuinka konfiguroida järjestelmä turvalliseksi noudattaen parhaita käytänteitä. Jokainen suositus viittaa yhteen tai useampaan CIS:in kehittämään kontrolliin (eng. Control), jotka kehitettiin parantamaan organisaatioiden kyberturvallisuuskyvykkyyksiä. Nämä kontrollit linkittyvät luotuihin standardeihin ja säännöksiin, mukaan lukien NIST:in Cybersecurity Framework, NIST SP 800-53, sekä ISO 27000 standardisarja. (Microsoft, 2023.)

Vastaavasti Australian kyberturvallisuuskeskus ACSC (eng. Australian Cyber Security Centre, 2017) ylläpitää sisällöltään samankaltaisia ohjeita erilaisten järjestelmien koventamiseen. Ohjeisiin sisältyy myös Windows 10:n koventaminen hyödyntämällä sovellusten hallinnointia, hyökkäyspinta-alan vähentämistä, sekä ryhmäpolitiikkaa. Myös Yhdysvaltojen puolustusjärjestelmävirasto (eng. US Department of Defense Systems Agency, DISA) tarjoaa teknisen turvallisuuden implementointioppaita (engl. Security Technical Implementation Guides, STIG). Kaikkia edellä mainittuja lähteitä voidaan hyödyntää järjestelmän

koventamisessa, mutta jokainen näistä vaatii myös räätälöintiä. Koventaminen valmiin pohjan avulla on helpompaa, kuin aloittaminen alusta, mutta prosessi on silti työläs, sillä se vaatii testaamista ja asetusten muuttamista kohdejärjestelmän käyttötarkoituksen mukaan.

2.6 Haavoittuvuusskannaus

2.6.1 Haavoittuvuusskannereista yleisesti

Haavoittuvuusskannerit ovat ohjelmia, joilla skannataan verkon arkkitehtuuria, raportoidaan havaittuja haavoittuvuuksia, sekä mahdollisesti tarjotaan ohjeita, kuinka korjata havaitut haavoittuvuudet (Holm, 2011). Haavoittuvuusskannereilla on yleensä kaksi tehtävää: verkon järjestelmien skannaaminen ja tulosten raportoiminen (Harrel ym., 2018). Haavoittuvuusskannausprosessi aloitetaan skannaamalla määritelty osa verkosta, tavoitteena on löytää aktiiviset laitteet verkosta. Kun aktiiviset laitteet on havaittu, skanneri pyrkii löytämään jokaiselta laitteelta avoimet portit, sekä päättelemään, mikä palvelu portissa toimii. Tämän jälkeen skanneri identifioi haavoittuvuudet kohdejärjestelmässä. (Badawy ym., 2013) Tämä tapahtuu vertailemalla kohdejärjestelmän käyttöjärjestelmää ja asennettuja ohjelmia tunnettuihin haavoittuvuuksiin, jotka sijaitsevat haavoittuvuusskannerin tietokannassa (Wack ym., 2003). Skannerit hyödyntävät Yhdysvaltain Kansallisen haavoittuvuustietokannan (National Vulnerability Database, NVD) määritelmiä ja haavoittuvuuksia. Tietokanta sisältää tunnistettuja haavoittuvuuksia ja aukkoja (Common Vulnerabilities and Exposures, CVE), kuten esimerkiksi EternalBlue-haavoittuvuuden (CVE-2017-0144).

Haavoittuvuusskannereiden välillä on eroja, ja jotkin tuotteet havaitsevat haavoittuvuuksia paremmin kuin toiset. Mahdollisena haittapuolena tästä seuraa se, että skanneri saattaa löytää myös haavoittuvuuksia, joita järjestelmässä ei ole. Nämä ovat niin sanottuja vääriä positiivia (Eng. False positive). Väärien positiivisten tulosten saaminen voi johtua esimerkiksi siitä, että skanneri tulkitsee skannattavan järjestelmän version tai sovelluksen väärin. Osa skannereista etsii käyttöjärjestelmän rekisteristä merkkejä päivityksen asentamisesta, kun taas osa skannereista tarkastaa päivitykseen liittyviä DLL-tiedostoja ja muita tiedostoja (Badawy ym., 2013). Näistä tarkempi, joskin myös hitaampi, metodi on jälkimmäinen (Beale ym., 2004). Haavoittuvuusskannerin tuloksiin voivat vaikuttaa myös häiriöt, kuten skannauspakettien putoaminen verkossa, tai skannattavan palvelun tilapäinen toimintahäiriö. Myös aika haavoittuvuuden julkaisemisen ja skannerin tietokannan päivittymisen välillä voi johtaa haavoittuvuuden huomaamatta jäämiseen. (Badawy ym., 2013)

Haavoittuvuusskannerin toimintaan voidaan vaikuttaa myös skannerin asetuksia ja konfiguraatiota muuttamalla. Mikäli skannaus suoritetaan sekä

autentikoituneena, että ilman tunnusten syöttämistä skannerille, skannausten tulokset voivat poiketa toisistaan. Myös skannerin konfiguroiminen voi vaikuttaa skannaustuloksiin. Haavoittuvuusskannaus voidaan ajaa myös sekä järjestelmän ulkopuolelta, että järjestelmän sisäpuolelta. Mikäli skannaus suoritetaan järjestelmän ulkopuolelta, tulee tämä huomioida palomuuriasetuksissa, jotta skannerin yhteysyritykset pääsevät kohdejärjestelmään perille. Skannauskategoriat voidaan jakaa käyttötapausten mukaan intrusiivisiin ja ei-intrusiivisiin skannauksiin. Intrusiivinen skannaus pyrkii hyödyntämään löytynyttä haavoittuvuutta, täten varmistaen haavoittuvuuden olemassaolon (Badawy ym., 2013). Ei-intrusiivinen skannaus ainoastaan identifioi ja raportoi löytyneen haavoittuvuuden. Haavoittuvuusskannauksen kohdealueet vaihtelevat paljon. Haavoittuvuusskannauksia voidaan kohdistaa IoT-verkkoihin ja laitteisiin, pilvipalveluihin, web-sivustoihin, käyttöjärjestelmiin, sekä mobiililaitteisiin.

Haavoittuvuusskanneri saattaa myös epäonnistua haavoittuvuuden havaitsemisessa, mutta silti tarjota haavoittuvuudelle korjausehdotuksen. Näin tapahtuu usein, jos haavoittuvuus on korjattu isommassa päivityksessä, jossa paikataan useita haavoittuvuuksia kerralla. On myös mahdollista, että havaittuun haavoittuvuuteen ei ole korjausehdotuksia, vaikkakin tämä on harvinaista. Useimmiten tarjolla on joko päivitys, tai toimintapa, jolla kierretään haavoittuvuus (eng. Workaround). (Holm, 2011.)

Harrel ym. (2018) arvioivat tutkimuksessaan haavoittuvuusskannereiden heikkouksia ja puutteita, sekä ehdottivat näihin parannuksia. Haavoittuvuusskannereiden ongelmana on informaation runsas määrä ilman, että informaatio sisältää toteutettavia ratkaisuja tai korjaavia toimenpiteitä. Tämän lisäksi haavoittuvuusskannereiden luomat korjausehdotukset vaativat usein resursseja muilta alustoilta. Tämä voi johtaa siihen, että raportti ohjaa sivustolle, jota ei enää löydy. Toinen ongelma on, että raportin antama ohje voi olla monta sataa sivua pitkä, jolloin korjaavan toimenpiteen löytäminen raportista on haastavaa. Harrel ym. (2018) toteavatkin, että haavoittuvuusskannereiden korjaus- ja raportointiominaisuudet ovat jääneet aiemmissa tutkimuksissa vähemmälle huomiolle kuin haavoittuvuusskannereiden vertaileminen ja niiden ajaminen erilaisia laitteita vasten. Tutkijat pyrkivät kehittämään haavoittuvuusskannerin tuottamaa raporttia täydentämällä sitä muun muassa vaihtoehtoisilla korjaustoimenpiteillä, sekä tarjoamalla haavoittuvuuden varmistamismahdollisuuden. Raporttien parantaminen vaatii kuitenkin runsaasti manuaalista työtä, mikä tekee prosessista hitaan ja työlään.

Myös Holm (2011) on tehnyt kattavan tutkimuksen, jossa vertailtiin useita haavoittuvuusskannereita niiden ominaisuuksien ja toiminnan perusteella. Tutkimuksessa vertailtiin skannerien havaitsemiskykyä, korjaavien toimenpiteiden ehdottamista, sekä väärien hälytysten määrää. Tämän lisäksi haavoittuvuusskannaukset ajettiin niin tunnusten kanssa, kuin ilman tunnuksia. Tutkimuksen tuloksina havaittiin, että haavoittuvuusskannerit eivät löydä kaikkia

haavoittuvuuksia, mutta tarjoavat valtaosaan havaituista haavoittuvuuksista korjaavia toimenpiteitä. Myös Holm toteaa, että haavoittuvuusskannereiden tuottamien raporttien läpikäyminen on työläs prosessi.

2.6.2 Haavoittuvuusskannereiden vertaileminen ja käyttäminen koventamisen toimivuuden mittaamiseen

Kuten aiemmin todettiin, haavoittuvuusskannereiden välillä on eroja. Osa eroista selittyy jo pelkästään sillä, että haavoittuvuusskannerin käyttötarkoitus poikkeaa toisesta. Myös skannerin toteuttamistapa selittää osan näistä eroista. Haavoittuvuusskannereita voidaan ajaa toiselta koneelta kohti skannattavaa konetta, tai skanneri voidaan asentaa skannattavaan järjestelmään. Jälkimmäisessä tavassa on kyse agenttipohjaisesta skannerista. On olemassa myös skannereita, jotka hyödyntävät molempia näistä, jolloin kyse on hybridimallista.

Myös sillä voi olla vaikutusta, onko kyseessä maksullinen vai ilmainen haavoittuvuusskanneri. Maksulliset, eli kaupalliset haavoittuvuusskannerit edustavat uusinta tekniikkaa ja ovat todistetusti tehokkaita identifioimaan ja analysoimaan turvallisuushaavoittuvuuksia (Amankwah ym., 2020). Kaupalliset haavoittuvuusskannerit ovat kuitenkin kalliita, mikä rajoittaa kaupallisten skannereiden käytön niille, jotka kykenevät kustantamaan tällaisen skannerin. Osa kaupallisista skannereista tarjoaa myös kattavammat ominaisuudet kuin ilmaiset skannerit. Kohteen kattavaan skannaamiseen ilmaisilla työkaluilla joutuu mahdollisesti hyödyntämään useampaa skanneria, kun taas kaupallisessa skannerissa nämä testit löytyvät yhdestä skannerista. On kuitenkin myös todistettu, että useamman haavoittuvuusskannerin käyttäminen on joka tapauksessa hyödyllistä kattavamman skannauksen ja tarkempien tuloksien saamiseksi (Esposito ym., 2018). Maksullisissa skannereissa on usein kattavampi haavoittuvuustietokanta, sekä mahdollisesti parempi havaitsemiskyky. Nämä kuitenkin vaihtelevat jokaisen haavoittuvuusskannerin kohdalla.

Chalvatzis ym. (2019) arvioivat tutkimuksessaan, miten haavoittuvuusskannerit Nessus, OpenVAS ja Nmap NSE vertautuvat toisiinsa ja kuinka niitä voidaan käyttää riskiarvioinnin automatisointiin organisaatiossa. Nmapin NSE:tä (eng. Nmap Scripting Engine) voidaan käyttää haavoittuvuusskannaukseen hyödyntämällä työkalun skriptausominaisuuksia (eng. Scripting engine). Tutkijat Chalvatzis ym. (2019) esittävät oman arviointiviitekehyksensä, jossa hyödynnetään virtuaalikoneita haavoittuvuusskannereiden alustoina ja kohdejärjestelminä. Tutkimuksessa pystytettiin verkko, jossa skannattavina järjestelminä toimivat Windows 10, Windows 7, Windows Server 2012 ja Ubuntu palvelin. Verkon skannaaminen kesti lyhyimmän aikaa NSE:llä ja pisimmän aikaa OpenVASilla. NSE oli valmis kymmenessä minuutissa, Nessus kahdessakymmenessä minuutissa, kun taas OpenVASilla kesti tunti ja kaksikymmentä minuuttia. Nessus

havaitsi kolme kriittistä haavoittuvuutta, kun taas OpenVAS ja NSE löysivät molemmat kaksi samaa kriittistä haavoittuvuutta.

Tutkijat argumentoivat, että haavoittuvuusskannerin tärkein aspekti on skannerin tietämyskannan (eng. Knowledge Base) laajuus. Vaikkakin laaja haavoittuvuustietokanta on olennainen osa haavoittuvuusskanneria, myös hyvä havaitsemistarkkuus on kriittinen ominaisuus haavoittuvuusskannerissa. Liiallinen määrä vääriä havaintoja, tai huono havaitsemiskyky heikentävät skannerin luotettavuutta. Tutkimuksessaan Chalvatzis ym. (2019) vertailivat skannereiden haavoittuvuustietokantojen laajuutta tarkastamalla, kuinka monta CVE:tä kunkin skannerin tietokannasta löytyy. Tutkijat latasivat skannereiden hyödyntämät tietokannat ja etsivät näistä uniikit CVE:t. Kävi ilmi, että Nessuksen tietokannasta löytyi 41 030 haavoittuvuutta, OpenVASin tietokannasta 35 464 haavoittuvuutta ja NSE:n tietokannasta kuusikymmentäseitsemän haavoittuvuutta. Tutkijat myös selvittivät, löytyikö tietokannoista sellaisia haavoittuvuuksia, joita ei löydy muista haavoittuvuustietokannoista. Kävi ilmi, että Nessus kattaa yli kymmenen tuhatta haavoittuvuutta, joita OpenVAS ei kata. OpenVAS puolestaan kattaa yli viisi tuhatta haavoittuuvuutta, joita Nessus ei kata. Myös NSE:stä löytyi muutama haavoittuvuus, joita ei löytynyt Nessuksesta, eikä OpenVAS:ista.

Haavoittuvuusskannereiden välillä on myös muita eroja. Haavoittuvuuksien vakavuuden luokittelussa Nessus käyttää luokittelua: kriittinen, korkea, keskitaso, matala. OpenVAS taas luokittelee haavoittuvuudet asteikolle: korkea, keskitaso ja matala. Myös yhdistäviä ominaisuuksia löytyy, OpenVAS ja Nessus molemmat hyödyntävät NASL-kieltä (eng. Nessus Attack Scripting Language), sillä haavoittuvuusskannerit haarautuvat samasta ohjelmakoodista. Chalvatzis ym. (2019) päättävätkin tutkimuksensa johtopäätöksellä, että kaikki kolme haavoittuvuusskanneria soveltuvat riskiarviointityökalun luomiseen. Nessuksella on kattavin haavoittuvuustietokanta, vahva markkinaosuus, sekä hyvä dokumentaatio. OpenVAS taas kykenee pitkälti samaan kuin Nessus, mutta haavoittuvuustietokanta on selvästi pienempi. Tutkimuksen tulokset vahvistavat, että useamman haavoittuvuusskannerin käyttäminen on kannattavaa. Haavoittuvuusskannerit sisältävät usein eri haavoittuvuuksia ja löytävät erilaisia haavoittuvuuksia vaihtelevasti. Tästä johtuen yhden haavoittuvuusskannerin käyttäminen saattaa johtaa tilanteeseen, jossa haavoittuvuuksia jää löytämättä. On kuitenkin epäselvää, kuinka montaa haavoittuvuusskanneria tulisi käyttää, jotta kaikki tunnetut haavoittuvuudet löydettäisiin.

Badawy ym. (2013) selvittivät tutkimuksessaan, kuinka luotettavasti haavoittuvuusskannereita voidaan käyttää asentamattomien päivityksien havaitsemiseen tai päivityksen asentamisen vahvistamiseen. Tutkijat keskittyivät Microsoft Windowsin päivityksiin ja kohdejärjestelmänä tutkimuksessa toimi Windows Server 2008. Tutkimuksessa käytettävät haavoittuvuusskannerit olivat McAfee Vulnerability Manager, Retina Network Security Scanner, Nexpose ja Nessus. Tutkimus toteutettiin kahdessa vaiheessa. Ensimmäisessä vaiheessa kohdepalvelin

skannattiin ennen ensimmäisenkään päivityksen asentamista, ja päivitysten asentamisen jälkeen. Toisessa vaiheessa skannaukset toistettiin, ennen toisen palvelupaketin (eng. Service Pack 2) asentamista, ja palvelupaketin asennuksen jälkeen. Ensimmäisen vaiheen tulokset osoittivat, että haavoittuvuusskannerit havaitsivat paremmin vanhat päivitykset, kuin uudemmat päivitykset. Myös väärien positiivisten määrä kaikilla skannereilla, paitsi Nexposella, oli melko korkea. Nessuksella ja Nexposella oli myös korkeampi väärien negatiivisten (eng. False negative) havaintojen määrä kuin McAfeella ja Retinalla. Väärällä negatiivisella tuloksella tarkoitetaan tapausta, jossa haavoittuvuusskanneri epäonnistuu havaitsemaan olemassa olevan haavoittuvuuden, tai tässä tapauksessa päivityksen. Toisessa vaiheessa palvelin skannattiin ennen ja jälkeen toisen palvelupaketin asentamisen. Toisen vaiheen tulokset osoittivat, että skannerit kykenivät havaitsemaan paremmin puuttuvat päivitykset toisen palvelupaketin asentamisen jälkeen. Tutkimuksesta käy ilmi, että haavoittuvuusskannereiden hyödyntäminen puuttuvien päivitysten havaitsemiseen toimii paremmin, jos kohdejärjestelmään on asennettu ensimmäinen palvelupaketti. Tutkijat päättävät tutkimuksensa johtopäätöksellä, että haavoittuvuusskannereiden ei tule olla ainoa tapa tarkastaa puuttuvia päivityksiä kohdejärjestelmästä. (Badawy, 2013.) Näin ollen voidaankin todeta, että haavoittuvuusskannereiden erot tulisi ottaa huomioon tarkasteltaessa niiden toimintamahdollisuuksia, ja tyypillisesti on kannattavinta hyödyntää useampaa kuin yhtä haavoittuvuusskanneria.

2.7 Haavoittuvuuksien hyödyntäminen

Aiemmissa luvuissa on esitelty, mitä haavoittuvuudet ovat, kuinka niitä luokitellaan, sekä missä eri hyökkäyksen vaiheissa niitä voidaan hyödyntää. Tässä luvussa käsitellään, miten haavoittuvuuksia hyödynnetään ja miksi haavoittuvuuksia pyritään korjaamaan.

Haavoittuvuudet ovat kyberturvallisuudessa keskiössä, koska ne mahdollistavat ulkopuolisten toimijoiden päästä käsiksi luottamukselliseen tietoon, pysäyttää järjestelmän toiminta, muuttaa tai poistaa dataa, sekä aiheuttaa jopa fyysistä vahinkoa. On kuitenkin tärkeää huomioida, ettei haavoittuvuus itsessään aiheuta järjestelmälle haittaa, vaan haavoittuvuuksien vahingollisuus syntyy siitä, kun haavoittuvuuksia hyväksikäytetään rakentamalla hyökkäys, joka hyödyntää haavoittuvuutta, täten antaen hyökkääjälle jalansijan järjestelmään. Haavoittuvuuden hyödyntämisen tarkoituksena voi olla rahallisen edun tavoitteleminen, palvelun pysäyttäminen tai häiritseminen, haktivismi, tai tiedon kerääminen. Hyökkäys voi aiheuttaa niin taloudellista kuin fyysistä haittaa, mutta myös mainehaittaa hyökkäyksen kohteeksi joutuneelle organisaatiolle.

Vuonna 2022 löydetty haavoittuvuus ProxyLogon (CVE-2021-26855) antoi hyökkääjälle mahdollisuuden ohittaa sisäänkirjautumisen ja tekeytyä järjestelmänvalvojaksi Microsoftin Exchange-palvelimella. Kyseistä haavoittuvuutta

hyödyntämällä hyökkääjän on mahdollista suorittaa komentoja palvelimella. (Infosec, 2022.) Järjestelmänvalvojan oikeuksilla hyökkääjän on mahdollista poistaa, muuttaa tai lisätä tietoja palvelimelle. Palvelimelta on mahdollista poistaa tai lähettää sähköposteja, mutta myös havaita verkon muita laitteita ja hyökätä myös niitä kohti. Toinen laajaa tuhoa aiheuttanut haavoittuvuus on vuonna 2017 julkaistu CVE-2017-0144. Kyseessä on haavoittuvuus, joka johtui Microsoftin puutteellisesta toteutuksesta koskien Server Message Block (SMB) -protokollaa (NIST, 2017). Tämä haavoittuvuus antoi hyökkääjälle mahdollisuuden suorittaa koodia kohdekoneella. Haavoittuvuutta varten kehitettyä hyväksikäyttökoodia EternalBlue:a on käytetty kiristyshaittaohjelmien WannaCry:n, sekä NotPetya:n levittämiseen. WannaCry kiristyshaittaohjelma levisi vuonna 2017 yhden päivän aikana yli 230 000:een Windows-koneeseen. Kohteisiin lukeutui suuria organisaatioita kuten FedEx, sekä Deutsche Bahn. (Avast, 2020.) WannaCry:n arvioidaan aiheuttaneen vahinkoa yli neljän miljardin dollarin edestä (CBC News, 2017).

Kuten aiempi tutkimuskirjallisuus osoittaa, haavoittuvuuksista, niiden toimintaperiaatteista, sekä haavoittuvuusskannereiden merkityksestä haavoittuvuuksien tunnistamisessa on merkittävä rooli. Kyberuhkien lisääntyessä yhteiskunnassa on kuitenkin tärkeää rakentaa mahdollisimman ajankohtaista ja kattavaa ymmärrystä siitä, millaisin keinoin kyberturvallisuutta työasemilla voi parantaa käyttäen olemassa olevia työkaluja ja haavoittuvuuksien tunnistamismenetelmiä. Näin ollen seuraavissa luvuissa esitellään tämän tutkimuksen menetelmä ja tulokset, sekä niistä tehdyt johtopäätökset.

3 TUTKIMUSMENETELMÄ

Tässä luvussa esitellään tutkimuksessa käytetty tapaustutkimus tutkimusmenetelmänä. Lisäksi luvussa kuvataan, kuinka tutkimus on toteutettu, millaista aineistoa tutkimus on tuottanut, ja kuinka tutkimusprosessi on rakentunut. Tälle tutkimukselle on asetettu kolme tutkimuskysymystä:

1. Kuinka haavoittuvainen Windows 10 käyttöjärjestelmä on?
2. Kuinka hyvin valmiilla ryhmäpolitiikkapohjalla voidaan koventaa Windows 10 käyttöjärjestelmä?
3. Kuinka haavoittuvuusskannerit soveltuvat mittaamaan koventamisen onnistumista?

Ensimmäisen tutkimuskysymyksen avulla pyritään selvittämään tunnettuja haavoittuvuuksia Windows 10 käyttöjärjestelmässä.
Kysymyksiin vastataan tutkimuksessa saatujen tulosten perusteella, sekä arvioidaan tulosten luotettavuutta.

3.1 Tapaustutkimus tutkimusmenetelmänä

Tämän tutkimuksen tutkimusmenetelmänä toimii tapaustutkimus. Tapaustutkimuksen tavoitteena on prosessien ja muiden muuttujien syvällinen ymmärtäminen. Tapaustutkimus keskittyy kuvaamaan, ymmärtämään, ennakoimaan, sekä kontrolloimaan tutkimuksen kohdetta. Kohteena voi toimia esimerkiksi prosessi, henkilö, ryhmä, tai organisaatio. (Woodside, 2017.) Tapaustutkimukselle on ominaista, että tutkimuksessa keskitytään yksittäiseen kohteeseen pitkän aikaa sen sijaan, että tutkittaisiin useita kohteita samanaikaisesti lyhyemmän aikaa (Skinner, 1963). Yin (2014) määrittelee tapaustutkimuksen olevan empiirinen tutkimus, jossa käytetään kattavaa ja useilla eri tavoilla hankittua tietoa, tapahtuman tutkimiseksi rajatussa ympäristössä. Anttila (1998) listaa tapaustutkimukselle ominaisia piirteitä seuraavasti:

- Tapaustutkimukset ovat syvätutkimuksia jostakin kohteesta, antaen siitä tarkan, kattavan ja hyvin organisoidun kuvan.
- Tutkimuksella pyritään selvittämään suppeaa kohdetta suurella määrällä muuttujia.
- Koska tutkimusmenetelmä on luonteeltaan intensiivinen, sen avulla voidaan havaita tekijöitä, joita voidaan tutkia lisää muilla menetelmillä.
- Tapaustutkimusta voidaan käyttää jatkotutkimuksen valmistelemiseen samasta aiheesta.

Anttila (1998) myös mainitsee, että tapaustutkimuksen perustapahtumat ovat toistettavissa, vaikka kahta täysin samanlaista tilannetta ei voida saada aikaan.

Tämä johtuu siitä, että tutkijan raportti tutkitusta tapauksesta on tutkijan oma tulkintansa tilanteesta.

3.2 Tutkimuksen toteutus

Tässä alaluvussa perehdytään tutkimuksen toteuttamiseen, tutkimuksessa käytettyyn aineistoon, sekä tutkimusprosessiin. Tutkimus toteutetaan pystyttämällä virtuaaliympäristö, jossa tutkimus toteutetaan. Tutkimuksessa käytetään toimialueeseen kuuluvaa Windows 10 -käyttöjärjestelmää, toimialueohjainpalvelinta (engl. Domain Controller, DC), sekä useampaa haavoittuvuusskanneria. Skannausten kohteena oleva Windows 10 -käyttöjärjestelmä kovennetaan valmiin ryhmäpolitiikan avulla, ja koventaminen arvioidaan haavoittuvuusskannereiden tuottamien raporttien perusteella.

Työasema skannataan haavoittuvuusskannereilla ennen ja jälkeen työaseman koventamisen. Skannaukseen käytettiin seuraavia haavoittuvuusskannereita:

1. Rapid7:n Nexpose
2. ManageEngine:n Vulnerability Manager Plus
3. Greenbone:n Enterprise TRIAL
4. Tenable:n Nessus
5. Qualys:in VMDR (Vulnerability Management, Detection and Response)

Käytetyistä haavoittuvuusskannereista Nexpose, Vulnerability Manager Plus ja Nessus asennettiin paikallisesti työasemalle. Greenbonen Enterprise TRIAL on puolestaan oma virtuaalikoneensa, eli skannaaminen tapahtuu työaseman ulkopuolelta. Vastaavasti Qualys toimii pilvestä ja työasemalle asennetaan agentti, joka suorittaa toimenpiteet ja lähettää tiedot pilveen. Käytetyistä haavoittuvuusskannereista on tarjolla ilmainen kokeiluversio, jonka kokeilujakso vaihtelee tuotteen mukaan. Rapid7 tarjoaa Nexpose-haavoittuvuusskannerilleen kolmenkymmen päivän kokeilujakson. Vastaavasti myös ManageEngine ja Qualys tarjoavat skannereilleen yhtä pitkän kokeilujakson. Tenable tarjoaa Nessus-tuotteelleen viikon ilmaisen kokeilujakson. Greenbone tarjoaa Enterprise-skanneristaan kokeiluversion, jolle ei ole määritelty kokeilujaksoa. Haavoittuvuusskanneria voi käyttää määräämättömän ajan. On kuitenkin hyvä huomata, että kokeiluversio käyttää ainoastaan yhteisön syöttämää uhkatietokantaa, kun taas maksullinen versio hyödyntää yrityksen syöttämää uhkatietokantaa.

3.2.1 Tutkimuksen aineisto

Tutkimuksessa aineistona käytetään haavoittuvuusskannereiden tuottamia raportteja. Haavoittuvuusskannerit tuottavat raportin kohteen skannauksen jälkeen, ja nämä raportit tallennettiin analysointia varten. Raportti sisältää havaitut

haavoittuvuudet, niiden vakavuudet, sekä mahdollisesti korjausehdotuksia. Tutkimusaineistona toimii käytettyjen viiden haavoittuvuusskannerin raportit, jotka on kerätty ennen koventamista, sekä koventamisen jälkeen. Aineiston tukena on käytetty kirjallisuuskatsauksessa esiteltyä kirjallisuutta. Kirjallisuuskatsauksella on luotu teoreettinen pohja tutkimusaiheelle keräämällä tietoa käyttöjärjestelmien haavoittuvuuksista, koventamismetodeista, haavoittuvuusskannereista, sekä keräämällä tietoa siitä, kuinka koventamisen onnistumista voidaan varmentaa.

3.2.2 Tutkimusprosessi

Tutkimusta varten pystytettiin virtuaaliympäristö, johon kuuluvat Windows Server 2019-palvelimella toimiva toimialueohjain, sekä Windows 10-käyttöjärjestelmällä toimiva virtuaalikone. Windows 10-työasemalle asennettiin Nessus, Nexpose ja Vulnerability Management Plus haavoittuvuusskannerit. Tämän lisäksi Greenbonen haavoittuvuusskanneri asennettiin omaksi virtuaalikoneekseen. Tutkimuksessa Windows 10 skannattiin haavoittuvuusskannereilla heti kun virtuaalikone oli asennettu ja päivitetty, ja haavoittuvuusskannerit oli asennettu työasemalle. Haavoittuvuusskannereiden antamat raportit tallennettiin, minkä jälkeen käyttöjärjestelmä kovennettiin valmiilla ryhmäpolitiikkapohjalla. Ryhmäpolitiikka asennettiin toimialueohjaimelle, josta asetukset päivittyvät työasemalle. Tämän jälkeen käyttöjärjestelmä skannattiin uudestaan haavoittuvuusskannereilla. Haavoittuvuusskannereiden tuottamia tuloksia vertailtiin ja analysoitiin sen perusteella, onko koventaminen toiminut, sekä miten haavoittuvuusskannerit toimivat koventamisen onnistumisen mittarina. Tämän lisäksi arvioitiin havaittujen haavoittuvuuksien vakavuutta, sekä haavoittuvuusskannereiden havaintojen välisiä eroja. Nämä analyysin tulokset esitellään seuraavissa luvuissa.

3.3 Tutkimuksen luotettavuus

Tutkimuksen luotettavuuden arvioinnissa on huomioitava, etteivät tutkimuksessa havaitut haavoittuvuudet välttämättä ole täysin kattava listaus siitä, mitä haavoittuvuuksia Windows 10 -käyttöjärjestelmästä löytyy ennen ja jälkeen koventamisen. Tämä johtuu siitä, että haavoittuvuusskannerit eivät pysty tunnistamaan kaikkia haavoittuvuuksia täysin luotettavasti. Tämä selittää myös sitä, miksi eri haavoittuvuusskannereilla saadaan erilaisia tuloksia. Haavoittuvuusskannerit käyttävät eri laajuisia tietokantoja, eivätkä nämä tietokannat välttämättä kata kaikkia haavoittuvuuksia. On myös hyvä huomata, että haavoittuvuusskannerit eivät havaitse nollapäivähaavoittuvuuksia, sillä näitä ei ole mahdollista lisätä tietokantaan ennen haavoittuvuuden julkaisemista. Havaitut haavoittuvuudet antavat kuitenkin alustavan suunnan siitä, kuinka paljon haavoittuvuuksia on ja kuinka vakavista haavoittuvuuksista on kyse.

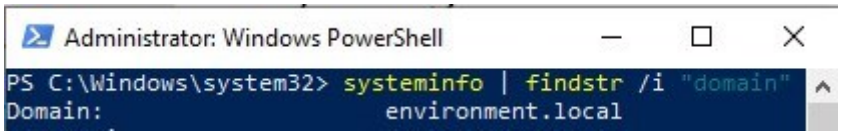
Haavoittuvuuksien havaitsemisen luotettavuutta on pyritty parantamaan tutkimuksessa valitsemalla useampi haavoittuvuusskanneri, jolla skannaus suoritetaan. Tutkimustuloksien luotettavuutta parantaa myös se, että tutkimuksessa käytetään haavoittuvuuksien pisteyttämiseen skannereiden pisteytyksestä riippumatonta CVSS-pisteytystä. Kaikki haavoittuvuusskannerit eivät ilmoita havaitsemiaan haavoittuvuuksia näin, vaan käyttävät omia metriikoitaan. Koska tulokset on yhdenmukaistettu käyttämällä CVSS-pisteytystä, haavoittuvuuksien tulkinnat eivät perustu haavoittuvuusskannereiden omiin metriikoihin, mikä mahdollistaa yhdenmukaisemman tulosten vertailun.

Tutkimuksen tulosten tarkastelussa on tärkeää huomioida, että palomuuri on sammutettu skannausten ajaksi, jotta skannaukset pääsevät perille ongelmitta. Tämä johti tutkimuksessa myös sellaisten haavoittuvuuksien havainnointiin, jotka poistuvat, kun palomuuri laitetaan takaisin päälle. Tutkimuksessa kohteena käytettiin virtuaalityöasemaa tutkimuksen toteutettavuuden helpottamiseksi. Jatkotutkimuksessa olisi mahdollista tehdä vastaava tutkimus, jossa verrattaisiin skannausten tuloksia virtuaalialustalla pyörivän käyttöjärjestelmän ja normaalilla työasemalla pyörivän käyttöjärjestelmän välillä. Näin voitaisiin selvittää, vaikuttaako virtuaalialusta tutkimuksen tuloksiin.

4 TUTKIMUKSEN KOEYMPÄRISTÖ

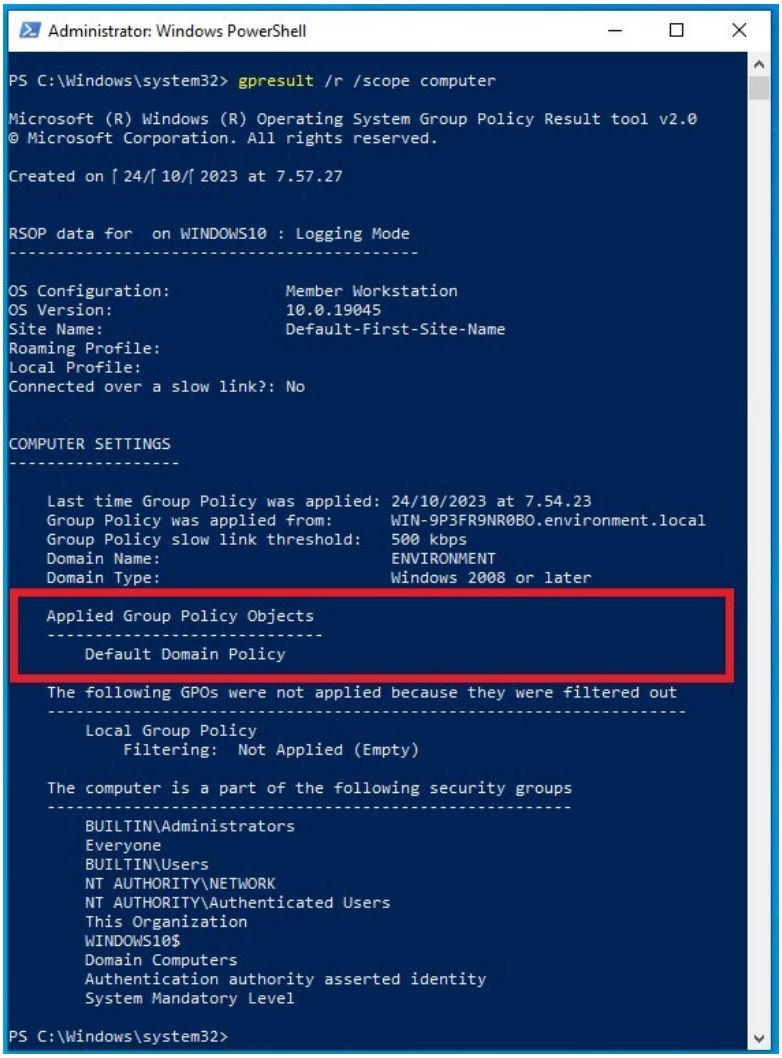
Tässä luvussa käydään läpi, mistä laitteista koeympäristö koostuu, mitä haavoittuvuusskannereita ympäristössä käytetään, ja mitä ryhmäpolitiikkapohjia koventamisessa käytetään. Koeympäristön, sekä ryhmäpolitiikan toimintaa havainnollistetaan kuvin. Kuten aiemmin mainittiin, virtuaalisessa ympäristössä käytetään yhtä toimialueohjainta, sekä yhtä työasemaa. Toimialueohjaimena toimii Windows Server 2019, jolla hallinnoidaan tutkimuksen kannalta relevantteja ryhmäpolitiikkoja. Palvelimesta on käytössä ilmainen Windows Server 2019 Standard Evaluation -editio, jonka versio on 1809 ja OS build 17763.4851. Palvelimelle on asennettu viimeisimmät päivitykset ennen tutkimuksen suorittamista. Palvelimelle on asennettu myös Active Directory Directory Services, tämän lisäksi palvelin toimii myös DNS-palvelimena. Palvelimelle luotu toimialue on nimeltään environment.local, johon myös virtuaalinen työasema on liitetty. Työaseman käyttöjärjestelmä on Windows 10 Enterprise Evaluation, versio 22H2, OS build 19045.2006, Windows Feature Experience Pack 120.2212.4190.0.

Ensimmäisenä työaseman liittäminen toimialueeseen tulee varmistaa. Tämä voidaan todentaa suorittamalla työasemalla powershellissä komento “systeminfo | findstr /i “domain””, joka tulostaa komentoriville kuvan 1 mukaisen tulosteen:



Kuva 1 Toimialueen varmistaminen

Tästä voidaan päätellä liittämisen onnistuneen. Seuraavaksi tulee tarkastaa, että työasemalla on voimassa oletusryhmäpolitiikka (eng. Default Group Policy). Ryhmäpolitiikka voidaan selvittää työasemalta suorittamalla komento “gpresult /r /scope computer”, jonka tulosteesta voidaan varmistaa sovellettu ryhmäpolitiikka kuvan 2 mukaisesti:



Kuva 2 Ryhmäpolitiikan selvittäminen

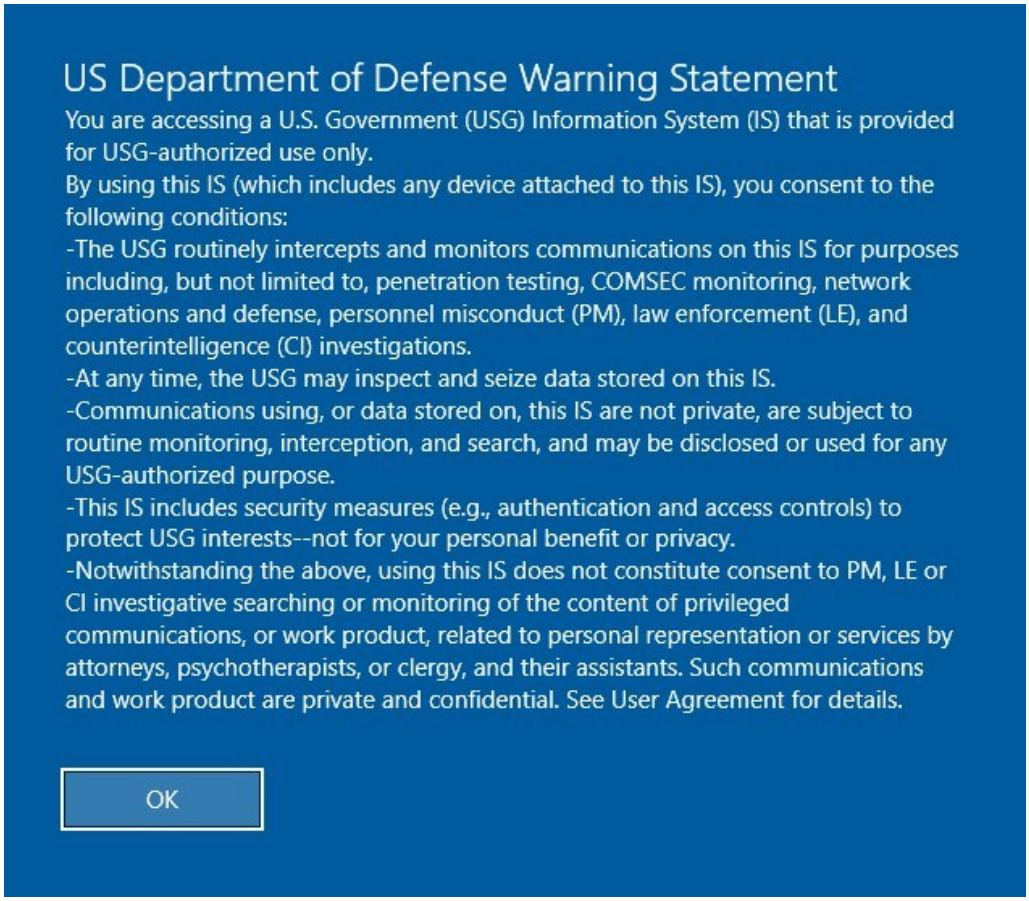
Oletusryhmäpolitiikka tulee työasemalle voimaan samalla, kun työasema liitetään toimialueeseen. Oletusryhmäpolitiikka ei ole konfiguraatioltaan kovin laaja, siinä määritellään ainoastaan tietokoneen konfiguraatio (eng. Computer Configuration), eikä lainkaan käyttäjäkonfiguraatiota (eng. User Configuration). Oletusryhmäpolitiikka määrittelee tietokoneen konfiguraatioon salasanapolitiikan, tilin lukitsemispolitiikan, käyttäjä- ja kerberospolitiikan, sekä kolme asetusta liittyen verkkoasetuksiin. Sen lisäksi, että konfiguraatio on suppea, se on myös turvallisuuden kannalta melko salliva. Esimerkiksi salasanan minimipituus on seitsemän merkkiä. Positiivisia havaintoja oletuspolitiikasta on kuitenkin muutamia: salasanan tulee olla tarpeeksi monimutkainen (eng. Password must meet complexity requirements), minkä lisäksi salasanahistoria kattaa kaksikymmentäneljä salasanaa, mikä estää salasanojen kierrättämisen. Seitsemänmerkkinen salasana, vaikka sen pitääkin sisältää erikoismerkkejä ja numeroita, on kuitenkin liian lyhyt ja voidaan murtaa.

DoD Cyber Exchange on DISA:n (Defense Information Systems Agency) sponsoroima alusta, joka tarjoaa kyberturvallisuuteen liittyvää tietoa, ohjausta, turvallisuuskäytäntöjä, sekä koulutusta. Näiden ohjeiden avulla pyritään siihen, että käyttäjä noudattaa sovittuja sääntöjä, hyviä käytänteitä, sääntelyä, sekä lakia. (DoD Cyber Exchange, 2023) Alusta tarjoaa myös valmiita ryhmäpolitiikkamalleja, joiden avulla organisaatiot ja yksilöt voivat koventaa järjestelmiään. Tässä tutkimuksessa hyödynnetään lokakuussa 2023 julkaistua ryhmäpolitiikkapohjaa. Tiedosto ladataan toimialueohjaimelle ja puretaan. Tiedosto sisältää selkeät ohjeet siitä, kuinka ryhmäpolitiikkaobjektit tuodaan tietokoneelle. Tässä hyödynnetään tiedoston mukana tulevaa skriptiä "DISA_GPO_Baseline_Import.ps1", sekä .csv-tiedostoa, joka määrittelee tuotavat ryhmäpolitiikat. DoD:n tuottama paketti sisältää ryhmäpolitiikkoja myös Adobe:n tuotteille, yleisimmille selaimille kuten Google Chrome, Mozilla Firefox ja Microsoft Edge. Tämän lisäksi paketista löytyy kovennuspohjat Microsoft Office-tuotteille. Tässä tutkimuksessa tuodaan valmiista pohjista seuraavat:

- DoD Internet Explorer 11 STIG Computer v2r4
- DoD Internet Explorer 11 STIG User v2r4
- DoD Microsoft Defender Antivirus STIG Computer v2r4
- DoD Microsoft Edge STIG Computer v1r7
- DoD Windows 10 STIG Computer v2r6
- DoD Windows 10 STIG User v2r6

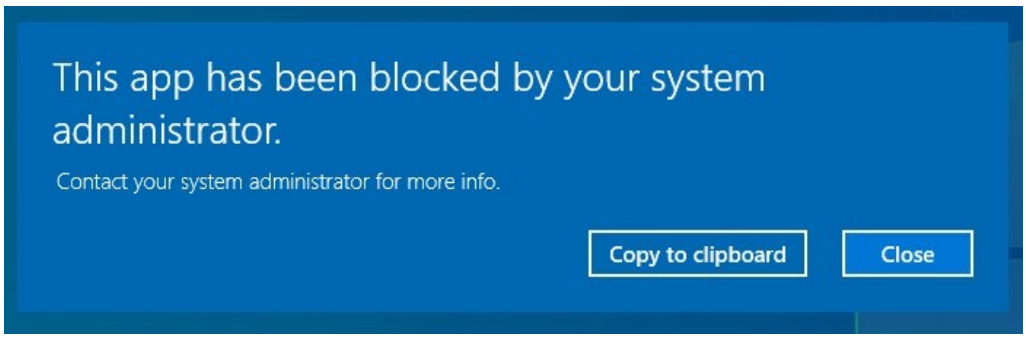
Myös "DoD Windows Firewall v1r7"-ryhmäpolitiikkapohja olisi äärimmäisen hyödyllinen, mutta tässä tutkimuksessa palomuuri on disabloitu skannereiden toimivuuden takaamiseksi. Tästä syystä kyseistä politiikkaa ei tuotu käyttöön.
DoD:n ryhmäpolitiikkapohja on selvästi laajemmin määritelty ja määrityksiltään tiukempi, kuin Windows oletuksellinen ryhmäpolitiikka. Salasanojen minimipituus on neljätoista merkkiä, ja tunnus lukkiutuu kolmen epäonnistuneen kirjautumisyrityksen jälkeen. Järjestelmänvalvojatunnus on uudelleennimetty muotoon "X_Admin" tyypillisen "Administrator"-nimen sijaan. Käyttäjäpuolen ryhmäpolitiikassa on poistettu käytöstä esikatseluominaisuus (eng. Preview), joka itsessään sisältää haavoittuvuuksia. Esimerkkinä voidaan käyttää Follina-nollapäivähaavoittuvuutta (CVE-2022-30190), jonka hyväksikäyttöön riittää .rtf-tiedoston klikkaaminen, jolloin esikatseluikkuna aukeaa ja haavoittuvuutta voidaan hyödyntää (Help Net Security, 2023).

Koventamisen onnistuminen on helppo todentaa, sillä työasemalle kirjautuessa tulee kuvan 3 mukainen ilmoitus:



Kuva 3 Kirjautumisilmoitus

Tämän lisäksi esimerkiksi powershellin käynnistäminen järjestelmänvalvojan oikeuksilla ei onnistu, koska politiikka estää sen, kuten kuva 4 osoittaa:



Kuva 4 Sovelluksen suorittamisen estämisilmoitus

Näiden vaiheiden avulla tässä tutkimuksessa toteutettiin Windows 10-käyttöjärjestelmän koventaminen. Tutkimuksen tulokset esitellään seuraavassa luvussa hyödyntäen haavoittuvuusskannereiden antamia raportteja.

5 TULOKSET

Haavoittuvuusskannerit havaitsivat koventamattomalta työasemalta haavoittuvuuksia seuraavasti:
Greenbone Enterprise TRIAL havaitsi yhteensä kolme haavoittuvuutta, joista kaksi oli Medium-tason haavoittuvuuksia, ja yksi Low-tason haavoittuvuus. Vulnerability Manager Plus puolestaan havaitsi kaksi haavoittuvuutta, joista toinen oli tasoa Moderate, ja toinen Important. Vulnerability Manager Plus havaitsi tämän lisäksi myös lukuisia konfiguraatiovirheitä, kuten TLSv1.1-protokollan sallimisen. Nessus havaitsi neljä haavoittuvuutta, joista yksi oli High-tason havainto ja kolme Medium-tason havaintoa. Tämän lisäksi Nessus havaitsi kolmekymmentäseitsemän Informational-tason havaintoa, jotka ovat rinnastettavissa konfiguraatiovirheisiin. Nexpose havaitsi yhdeksän haavoittuvuutta, joista viisi oli Severe-tason haavoittuvuuksia ja neljä Moderate-tason haavoittuvuuksia. VMDR havaitsi seitsemän haavoittuvuutta, joista yksi oli Critical-tason havainto, kolme Serious-tason havaintoa, ja kolme Medium-tason havaintoa. Koska jokainen haavoittuvuusskanneri käyttää hieman erilaista tapaa luokitella havainnot, on havainnot koottu alle taulukkoon. Taulukon luokittelussa noudatetaan CVSS haavoittuvuuksien vakavuuden luokittelumetriikkaa, jossa pistemäärät jaetaan kategorioihin kriittinen (eng. Critical), korkea (eng. High), keskitaso (eng. Medium), matala (eng. Low) ja tyhjä (eng. None). Taulukkoon on kirjattu jokaisen haavoittuvuusskannerin havainnot, mutta luokittelu on tehty CVSS-pisteytyksen mukaan. Täten esimerkiksi Vulnerability Manager Plus:in määrittelemä Moderate-tason havainto kuuluu taulukossa luokkaan High, koska itse haavoittuvuuden CVSS-pisteytys on 7.6. Ilmoitetuista haavoittuvuuksista on suodatettu muista haavoittuvuusskannereista johtuvat havainnot. Esimerkiksi Vulnerability Manager Plus avasi http-protokollalla toimivan käyttöliittymän, joka näkyi haavoittuvuusskannauksissa suojaamattomana protokollana, sekä itsekirjoitettuina sertifikaatti-ilmoituksina. Taulukko 1 esittelee haavoittuvuusskannereilla havaitut haavoittuvuudet ennen työaseman koventamista. Taulukossa raportoidaan jokaisen haavoittuvuusskannerin osalta tiettyyn CVSS-luokkaan sijoittuvien havaittujen haavoittuvuuksien määrä, sekä sulkeissa kyseisten haavoittuvuuksien CVSS-pisteet.

Taulukko 1 Havaitut haavoittuvuudet ennen koventamista

	Critical CVSS 9.0–10.0	High CVSS 7.0–8.9	Medium CVSS 4.0–6.9	Low CVSS 0.1–3.9	None CVSS 0.0
Enterprise TRIAL	-	-	2 (5.0 & 4.3)	1 (2.1)	-
Vulnerability Manager Plus	-	2 (7.6 & 7.5)	-	-	37 (-)
Nessus	-	1 (7.5)	3 (6.5, 6.5, 5.3)	-	37 (-)
Nexpose	-	-	5 (6.2, 5.0, 4.3, 4.3, 4.0)	2 (2.6 & 2.6)	2 (0.0)
VMDR	-	2 (7.6 & 7.3)	4 (5.0, 5.0, 5.0, 5.0)	1 (2.6)	-

Kun toimialueohjaimelle asennettiin uudet ryhmäpolitiikat, saatiin haavoittuvuusskannereilla seuraavat tulokset, jotka on esitelty alla olevassa taulukossa (Taulukko 2) vastaavasti CVSS-pisteytyksen mukaan.

Taulukko 2 Havaitut haavoittuvuudet koventamisen jälkeen

	Critical CVSS 9.0–10.0	High CVSS 7.0–8.9	Medium CVSS 4.0–6.9	Low CVSS 0.1–3.9	None CVSS 0.0
Enterprise TRIAL	-	-	2 (5.0 & 4.3)	1 (2.1)	-
Vulnerability Manager Plus	-	2 (7.6 & 7.5)	-	-	24 (-)
Nessus	-	1 (7.5)	2 (6.5 & 6.5)	-	29 (-)
Nexpose	-	-	4 (5.0, 4.3, 4.3, 4.0)	2 (2.6 & 2.6)	2 (0.0)
VMDR	-	1 (7.6)	1 (5.0)	-	-

Huomattavia muutoksia tapahtui Vulnerability Manager Plus -haavoittuvuusskannerin osalta, sillä konfiguraatiovirheiden määrä laski kolmellatoista. Vastaavasti myös Nessuksen havainnoissa tapahtui laskua, yksi Medium-tason haavoittuvuus korjaantui, samoin kahdeksan konfiguraatiovirhettä. Nexposen havaitsemista haavoittuvuuksista korjaantui yksi. VMDR:n havainnoissa tapahtui suurin muutos haavoittuvuuksien osalta, sillä havaittujen haavoittuvuuksien määrä tippui seitsemästä kahteen. Haavoittuvuusskannereiden havaitsemia haavoittuvuuksia ja niiden merkityksellisyyttä tarkastellaan tarkemmin seuraavassa luvussa.

6 POHDINTA

Haavoittuvuusskannereiden tekemien havaintojen välillä havaittiin suuria eroja niin havaintojen määrässä, kuin laadussa. Koventamisen jälkeen Greenbone Enterprise TRIAL havaitsi kolme haavoittuvuutta, joista kaikki liittyivät verkkoliikenteeseen ja verkkoprotokolliin. Korkein havainto (CVSS: 5.0) liittyy tcp-porttiin 135: haavoittuvuus mahdollistaa RPC:tä (eng. Remote Procedure Call) käyttävien palveluiden listaamisen laitteella. Toinen Medium-tason havainto (CVSS: 4.3) on vanhentuneiden TLS-protokollaversioiden käyttäminen työasemalla. Tämä mahdollistaa verkkoliikenteen purkamisen ja seuraamisen ulkopuolisen tahon toimesta. Microsoft on ilmoittanut aikovansa estää vanhentuneiden TLS-versioiden tukemisen lähitulevaisuudessa (Microsoft, 2023). Matalin havainto (CVSS: 2.1) liittyy puolestaan ICMP-kontrolliprotokollaan: työasema vastaa ICMP aikaleimapyyntöön. Näistä haavoittuvuuksista voidaan suodattaa palomuurin aktivoimisella pois RPC:tä käyttävien palveluiden listaaminen, sekä ICMP:hen liittyvä haavoittuvuus. Täten ainoa haavoittuvuus, joka laitteelta löytyy sekä ennen että jälkeen koventamisen, on vanhentuneen TLS-protokollaversion käyttäminen. Haavoittuvuus on CVSS-pisteytykseltään 4.3, joten kyseessä ei ole kriittinen haavoittuvuus, joka vaarantaisi työaseman.

Vulnerability Manager Plus havaitsi työasemalta hyvin erilaisia haavoittuvuuksia, kuin Greenbonen haavoittuvuusskanneri. Haavoittuvuusskanneri havaitsi työasemalla kaksi CVSS-pisteytykseltään melko korkeaa havaintoa, sekä suuren määrän konfiguraatiovirheitä. Ennen työaseman koventamista työasemalta havaittiin kolmekymmentäseitsemän konfiguraatiovirhettä, kun taas koventamisen jälkeen virheitä löytyi kaksikymmentäneljä. Haavoittuvuuksien määrä pysyi samana. Havaituista haavoittuvuuksista pisteytykseltään korkeampi (CVE-2013-3900, CVSS: 7.6) liittyy haavoittuvuutta hyödyntävän sovelluksen suorittamiseen, joka mahdollistaa haitallisen ohjelmakoodin ja komentojen suorittamisen työasemalla. Haavoittuvuuden hyödyntäminen vaatii siis haitallisen sovelluksen suorittamisen työasemalla, sekä sovelluksen päätymisen työasemalle, tai työasemalla sijaitsevan sovelluksen muokkaamisen. Toinen havaittu haavoittuvuus, CVE-2023-38039 (CVSS: 7.5) johtaa curl-sovelluksen palvelunestohyökkäykseen. Haavoittuvuuden hyväksikäyttäminen vaatii curl:in suorittamisen komentoriviltä hyökkääjän haluamaan osoitteeseen. Hyökkäys johtaa sovelluksen tukkiutumiseen, joka kuluttaa työaseman resursseja, kuten prosessointikykyä ja muistia. Kyseessä ei kuitenkaan ole kovin kriittinen haavoittuvuus, sillä se ei vaaranna työasemaa muilla tavoin. Työaseman koventaminen ryhmäpolitiikalla vähensi huomattavasti konfiguraatiovirheiden määrää työasemalla, sillä havainnot tippuivat kahteenkymmeneenneljään kolmestakymmenestäseitsemästä. Tämä viittaisi siihen, että koventamiseen käytetty ryhmäpolitiikka muuttaa työaseman konfiguraatioasetuksia turvallisemmaksi, sisältäen vähemmän konfiguraatiovirheitä.

Vastaavia havaintoja saatiin myös Nessuksen haavoittuvuusskannerilla, joka havaitsi ennen koventamista kolmekymmentäseitsemän konfiguraatiovirhettä. Virheiden määrä koventamisen jälkeen laski kahteenkymmeneenyhdeksään. Havaittujen konfiguraatiovirheiden määrä koventamisen jälkeen poikkeaa viidellä Vulnerability Manager Plus:in tekemistä havainnoista. Haavoittuvuuksien määrä laski neljästä kolmeen koventamisen jälkeen, kun haavoittuvuus "SMB signing not required" korjautui ryhmäpolitiikan ajamisen jälkeen. Myös Greenbonen haavoittuvuusskanneri havaitsi kyseisen SMB-haavoittuvuuden, sekä haavoittuvuuden korjaantumisen työaseman koventamisen jälkeen. Nessuksen korkein havainto (CVSS: 7.5) liittyy siihen, että työasema tukee keskivahvaa salausta liittyen SSL:n salaamiseen. Nessus luokittelee keskivahvaksi salaukseksi kaikki salaukset, joissa käytetään alle 112 bittiä pitkää salausavainta, tai 3DES-salausmuotoa. Haavoittuvuutta hyväksikäyttämällä hyökkääjän on mahdollista varastaa esimerkiksi istuntoon liittyviä evästeitä (eng. Session cookies), joita voidaan käyttää kirjautumisessa. Nessuksen kaksi muuta havaintoa liittyvät vanhentuneiden TLS-versioiden tukemiseen työasemalla. Nessuksen ohella myös Nexpose ja Greenbonen haavoittuvuusskanneri havaitsivat nämä TLS-haavoittuvuudet.

Nexpose oli Nessuksen lisäksi ainoa haavoittuvuusskanneri, joka havaitsi haavoittuvuuden liittyen SSL:n salaamiseen keskivahvalla algoritmilla. Nexpose nosti kyseisen haavoittuvuuden Severe-tason havainnoksi, kun taas Nessus luokitteli havainnon tasolle High. Nexpose luokittelee havainnot kolmeen kategoriaan: Moderate, Severe ja Critical kun taas Nessus käyttää viisiosaista jaottelua: Info, Low, Medium, High, Critical. Haavoittuvuuden luokittelu vaikuttaa melko yhtenäiseltä, ottaen huomioon, että haavoittuvuus luokitellaan molempien skannereiden toimesta vakavammalle puolelle luokitteluasteikkoa. Nexpose havaitsi ennen koventamista yhdeksän havaintoa ja koventamisen jälkeen kahdeksan. Havaittujen haavoittuvuuksien määrässä ei tapahtunut suurta muutosta, ainoastaan haavoittuvuus liittyen SMB-protokollaan korjaantui koventamisen myötä, kuten myös Nessus havaitsi. Loput havaitut haavoittuvuudet liittyvät kaikki SSL- ja TLS-protokolliin. TLS-protokollasta tuetaan haavoittuvia versioita ja SSL tukee heikkoja salausalgoritmeja ja lyhyitä salausavaimia. Nexposen havainnot keskittyvätkin vahvasti näihin protokolliin. Nexposen havainnot olivat luokittelultaan matalia, korkein havainto ennen koventamista oli CVSS-pisteytykseltään 6.2 ja koventamisen jälkeen 5.0.

Qualysin VMDR poikkesi havainnoiltaan suuresti muista haavoittuvuusskannereista. VMDR havaitsi ennen koventamista seitsemän haavoittuvuutta, joista yksi oli luokitukseltaan Critical, kolme luokitukseltaan Serious ja kolme luokitukseltaan Medium-tason havaintoja. VMDR havaitsi työasemalta seuraavat haavoittuvuudet:

1. Microsoft WinVerifyTrus Signature Validation Vulnerability
2. Enabled Cached Logon Credential
3. Microsoft Windows Explorer AutoPlay Not Disabled

4. Windows Explorer Autoplay Not Disabled for Default User
5. Allowed Null Session
6. SMB Signin Disabled or SMB Signin Not Required
7. Built-in Guest Account Not Renamed at Windows Target System

Haavoittuvuusskannerin havainnot ovat myös erilaisia kuin muiden skannereiden, sillä haavoittuvuudet keskittyvät työaseman asetuksiin. Kuten Vulnerability Manager Plus, myös VMDR havaitsi "Microsoft WinVerifyTrust Signature Validation"-haavoittuvuuden (CVE-2013-3900). VMDR luokitteli haavoittuvuuden kriittiseksi, eli korkeimman tason havainnoksi. Vulnerability Manager Plus puolestaan arvioi haavoittuvuuden tasolle Important, joka on skannerin luokitteluasteikolta toiseksi korkein taso. Työaseman koventamisen jälkeen VMDR havaitsi työasemalla enää kaksi haavoittuvuutta: "Microsoft WinVerifyTrus Signature Validation Vulnerability"-haavoittuvuuden, sekä "Enabled Cached Logon Credential"-haavoittuvuuden. Työaseman koventamisen myötä viisi haavoittuvuuksista korjaantui. Tätä selittää se, että ryhmäpolitiikka keskittyy vahvasti muuttamaan työaseman asetuksia turvallisemmaksi ja VMDR-haavoittuvuusskanneri selvästi pyrkii havaitsemaan näitä työaseman asetuksista johtuvia haavoittuvuuksia.

Kun tarkastellaan haavoittuvuusskannereiden skannauksien tuloksia ennen työaseman koventamista, havaitaan että keskenään erilaisia haavoittuvuuksia havaittiin yhteensä kaksitoista. Näistä on suodatettu pois duplikaatit, sekä palomuurin sammuttamisesta aiheutuvat havainnot. Havaitut haavoittuvuudet on esitelty alla taulukossa 3:

Taulukko 3 Havaitut haavoittuvuudet ennen koventamista, suodatettu väärät havainnot pois

Haavoittuvuus	**CVSS-pisteytys**	**Haavoittuvuuden havainneet haavoittuvuusskannerit**
Microsoft WinVerifyTrus Signature Validation Vulnerability	7.6	Qualys Vulnerability Management Plus
Vulnerability CVE-2023-38039 are affected in Curl For Windows 8.2.1	7.5	Vulnerability Management Plus
SSL Medium Strength Cipher Suites Supported	7.5	Nessus Nexpose
SMB Signin Disabled or SMB Signin Not Required	7.3	Qualys Nessus Nexpose
TLS Version 1.0 Protocol Detection	6.5	Nessus Greenbone Enterprise Trial Nexpose
TLS Version 1.1 Protocol Detection	6.5	Nessus Greenbone Enterprise Trial Nexpose
Enabled Cached Logon Credential	5.0	Qualys
Microsoft Windows Explorer AutoPlay Not Disabled	5.0	Qualys
Windows Explorer Autoplay Not Disabled for Default User	5.0	Qualys
Allowed Null Session	5.0	Qualys
ssl-cve-2011-3389-beast	4.3	Nexpose
Built-in Guest Account Not Renamed at Windows Target System	2.6	Qualys

Työaseman koventamisen jälkeen havaittujen haavoittuvuuksien määrä laskee seitsemään, kuten taulukossa 4 näkyy:

Taulukko 4 Havaitut haavoittuvuudet koventamisen jälkeen, suodatettu väärät havainnot pois

Haavoittuvuus	**CVSS-pisteytys**	**Haavoittuvuuden havainneet haavoittuvuusskannerit**
Microsoft WinVerifyTrus Signature Validation Vulnerability	7.6	Qualys Vulnerability Management Plus
SSL Medium Strength Cipher Suites Supported	7.5	Nessus Nexpose
Vulnerability CVE-2023-38039 are affected in Curl For Windows 8.2.1	7.5	Vulnerability Management Plus
TLS Version 1.0 Protocol Detection	6.5	Nessus Greenbone Nexpose
TLS Version 1.1 Protocol Detection	6.5	Nessus Greenbone Nexpose
Enabled Cached Logon Credential	5.0	Qualys
ssl-cve-2011-3389-beast	4.3	Nexpose

Kun tarkastellaan tuloksia tutkimuskysymyksien kannalta, voidaan todeta, että tämän tutkimuksen perusteella Windows 10 -käyttöjärjestelmä itsessään on turvallinen. Yksikään haavoittuvuusskanneri ei havainnut käyttöjärjestelmässä kriittisiä haavoittuvuuksia. Korkein haavoittuvuus on CVSS-pisteytykseltään 7.6. Tämäkin haavoittuvuus vaatii haitallisen tiedoston suorittamisen työasemalla haavoittuvuuden hyödyntämiseksi, mitä vastaan voidaan suojautua hyödyntämällä esimerkiksi Windowsin omaa virussuojaa, joka tarkastaa sovelluksen ennen sovelluksen suorittamista. Käyttöjärjestelmän curl-sovelluksesta löydettiin myös High-luokan haavoittuvuus, joka hyödynnettynä johtaa sovelluksen toiminnan estymiseen, sekä resurssien kulutukseen työasemalla. Nämä haavoittuvuudet eivät kuitenkaan suoraan vaaranna työaseman tai käyttäjän tietojen turvallisuutta. Käyttöjärjestelmä oletuksellisesti sallii vanhentuneiden TLS-protokollien käyttämisen, sekä keskivahvojen salausalgoritmien hyödyntämisen. Tämä on ongelmallista ja voi johtaa tietojen vuotamiseen ulkopuoliselle taholle, mikäli hyökkääjä suorittaa MiTM-hyökkäyksen, tai saa käyttäjän vierailemaan haitallisella sivustolla. Organisaatiot voivat kuitenkin päättää, sallitaanko

vanhentuneiden TLS-protokollien käyttämistä. Tämän lisäksi voidaan myös määrittää, mitä salausalgoritmejä käytetään.

Työaseman koventaminen ryhmäpolitiikalla vähensi havaittuja haavoittuvuuksia ja konfiguraatiovirheitä. Haavoittuvuuksista korjaantui yksi High-luokan haavoittuvuus, kolme Medium-tason haavoittuvuutta, sekä yksi Low-luokan haavoittuvuus. Koventamisen voidaan todeta toimivan, sillä koventaminen vähentää haavoittuvuuksien määrää työasemalla ja täten pienentää hyökkäyspintaalaa. Ryhmäpolitiikalla työaseman koventaminen ei kuitenkaan korjaa kaikkia haavoittuvuuksia, vaan työaseman suojaamiseen tulee käyttää myös muita työkaluja, kuten palomuuria ja virussuojaa. Työaseman koventamisesta ryhmäpolitiikalla ei ole tehty aikaisemmin vastaavia tutkimuksia. Näin ollen tämän tutkimuksen tuloksia tältä osin ei voida verrata samaa aihetta käsitteleviin tutkimuksiin.

Tässä tutkimuksessa ei käyty tarkemmin läpi haavoittuvuusskannereiden havaitsemia konfiguraatiovirheitä, mutta havaittiin, että myös niiden määrä laski koventamisen myötä. Tämä indikoi, että oletusryhmäpolitiikkaa käyttävä työasema on alttiimpi haavoittuvuuksille, kuin kovennettua ryhmäpolitiikkaa käyttävä työasema. Haavoittuvuusskannereiden konfiguraatiovirheiden havaitsemiskyvyn selvittäminen on mielenkiintoinen tutkimusaihe tulevaisuuden kannalta. Valmiilla ryhmäpolitiikalla voidaan koventaa käyttöjärjestelmä hyvin, mutta tulee huomata, että työaseman käyttömahdollisuudet pienevät selvästi, jos ryhmäpolitiikkaa ei konfiguroida organisaatiokohtaisesti. Myös DoD Cyber Exchange huomauttaa tästä laatimissaan ryhmäpolitiikkaohjeissa.

Haavoittuvuusskannereiden soveltuminen työaseman koventamisen onnistumisen mittaamiseen jää tutkimustulosten valossa epäselväksi. Skannerit havaitsevat onnistuneesti käyttöjärjestelmästä haavoittuvuuksia. Havainnot kuitenkin vaihtelevat merkittävästi eri haavoittuvuusskannereiden välillä. Esimerkiksi Greenbonen haavoittuvuusskanneri havaitsi kaksi merkityksellistä haavoittuvuutta, jotka molemmat liittyivät TLS-protokollan versioihin. Qualys:in VMDR taasen havaitsi ennen koventamista seitsemän relevanttia haavoittuvuutta ja koventamisen jälkeen kaksi relevanttia haavoittuvuutta. Myös Vulnerability Management Plus havaitsi sellaisen haavoittuvuuden, jota yksikään muu skanneri ei havainnut. Nämä havainnot puoltavat sitä, että useamman haavoittuvuusskannerin käyttäminen kattavampien tuloksien saamiseksi on välttämätöntä. Jotkin haavoittuvuusskannerit soveltuvat paremmin koventamisen onnistumisen mittaamiseen, kuin toiset. Tämän tutkimuksen perusteella haavoittuvuuksien tunnistamiseen soveltuivat parhaiten valituista haavoittuvuusskannereista Qualysin VMDR, ManageEnginen Vulnerability Management Plus, sekä Tenablen Nessus. Jatkotutkimus voisi täydentää tässä tutkimuksessa saatuja tuloksia vertailemalla myös muita haavoittuvuusskannereita. Haavoittuvuusskannerit eivät ratkaise kokonaan koventamisen onnistumisen arvioimista, vaan tässä

vaaditaan myös manuaalista työtä, kun tarkastellaan mitä asetuksia koventamisella on muutettu ja mitä asetuksia koventamisella ei ole muutettu. Työaseman koventamisessa on kannattavaa hyödyntää valmiita ryhmäpolitiikkapohjia, mutta nämä valmiit pohjat tulee räätälöidä jokaiseen kontekstiin eri tavalla.

Tämä tutkimus tuo lisää tietoa siitä, mitä konkreettisia haavoittuvuuksia Windows 10 -käyttöjärjestelmässä havaitaan sen asentamisen ja päivittämisen jälkeen. Windows 10 -käyttöjärjestelmään liittyy suuri määrä erilaisia haavoittuvuuksia, mutta tämän tutkimuksen perusteella vain pieni osa näistä haavoittuvuuksista on relevantteja päivitysten asentamisen jälkeen. Vastaavia tutkimuksia, joissa tutkitaan ryhmäpolitiikalla koventamisen toimivuutta, ei kirjallisuuskatsauksen perusteella ole tehty aiemmin. Tutkimuksessa havaittiin myös, että haavoittuvuusskannereiden valmistajat luokittelevat haavoittuvuuksia eri tavoilla. Luokittelu itsessään voi olla numeerista tai sanallista, minkä lisäksi käytetyt mittaasteikot vaihtelevat ja haavoittuvuuden konteksti vaikuttaa sen luokitteluun. Tämä johtaa siihen, että eri haavoittuvuusskannerit saattavat antaa samalle haavoittuvuudelle eri pisteytyksen. Haavoittuvuusraporteista ei myöskään käy aina suoraan ilmi, käyttääkö skanneri luokittelussa CVSS-pisteytystä, vai omaa luokitteluaan. Haavoittuvuusskannereiden valmistajien voisi olla hyödyllistä miettiä haavoittuvuuksien raportoinnin ja luokittelun yhtenäistämistä ja selkeyttämistä. Tämä helpottaisi eri haavoittuvuusskannereiden vertailemista.

7 Johtopäätökset

Tämän tutkimuksena tavoitteena oli selvittää, kuinka haavoittuvainen käyttöjärjestelmä Windows 10 on, ja kuinka hyvin käyttöjärjestelmän koventaminen onnistuu hyödyntämällä valmista ryhmäpolitiikkaa. Tämän lisäksi tutkimus pyrki selvittämään, kuinka haavoittuvuusskannerit soveltuvat mittaamaan koventumisen onnistumista. Tutkimuksessa havaittiin, että vasta asennetussa ja päivitetyssä Windows 10 -käyttöjärjestelmässä ei ole kriittisiä haavoittuvuuksia haavoittuvuusskannereiden tulosten perusteella. Korkein haavoittuvuus oli CVSS-pisteytykseltään 7.6, joka on High-tason havainto. Haavoittuvuuden hyväksikäyttäminen vaatii haitallisen tiedoston suorittamisen työasemalla, ja tämän haavoittuvuuden hyväksikäyttöä voidaan estää käyttämällä työasemalla Windowsin omaa virussuojaa.

Koventaminen valmiilla ryhmäpolitiikalla vaikuttaa tulosten perusteella toimivan. Haavoittuvuuksien määrä laski koventamisen jälkeen kahdestatoista haavoittuvuudesta seitsemään haavoittuvuuteen. Tämän lisäksi konfiguraatiovirheiden määrä käyttöjärjestelmässä laski koventamisen jälkeen. Tulosten perusteella voidaan todeta, että haavoittuvuusskannerit antavat osviittaa käyttöjärjestelmän haavoittuvuuksien määrästä ja vakavuudesta, mutta tulee muistaa, että haavoittuvuusskannerit eivät välttämättä huomaa kaikkia haavoittuvuuksia. Tämän vuoksi tulisikin käyttää useampaa haavoittuvuusskanneria, kun pyritään mittaamaan haavoittuvuuksien määrää tai varmistamaan koventamisen onnistuminen. Haavoittuvuusskannerit ovat hyvä työkalu, jota voidaan hyödyntää koventamisen onnistumisen mittaamisessa, mutta niiden ei tulisi olla ainoa työkalu. On myös hyvä huomata, että haavoittuvuusskannereiden raporttien läpikäyminen vaatii manuaalista työtä, asiantuntemusta ja aikaa.

Tämä tutkimus tarjoaa lisätietoa Windows 10 -käyttöjärjestelmän turvallisuudesta, kun käyttöjärjestelmä on päivitetty uusimpaan versioon, eikä siihen ole asennettu ylimääräisiä toimintoja. Tutkimuksen tulokset perustelevat sitä, miksi ryhmäpolitiikka on toimiva tapa koventaa työasema, sillä koventaminen vähensi haavoittuvuuksien ja konfiguraatiovirheiden määrää käyttöjärjestelmässä. Ryhmäpolitiikan avulla koventaminen on helppo skaalata koskettamaan koko organisaatiota. Tutkimus tukee aikaisempia tutkimustuloksia sen suhteen, että useamman haavoittuvuusskannerin käyttäminen on hyödyllistä kattavampien tulosten saamiseksi. Tutkimus tarjoaa myös käytännönläheistä tutkimustietoa eri haavoittuvuusskannereista sekä siitä, kuinka nämä haavoittuvuusskannerit vertautuvat toisiinsa. Tutkimuksessa havaittiin myös, että haavoittuvuusskannereita voidaan käyttää koventamisen onnistumisen mittaamisessa, mutta skannereiden ei tulisi olla ainoa käytetty työkalu tässä prosessissa. Tutkimus tarjoaa myös uusia jatkotutkimusideoita, kuten haavoittuvuusskannerien hyödyntäminen konfiguraatiovirheiden havaitsemisessa ja muiden haavoittuvuusskannereiden testaaminen käyttöjärjestelmän haavoittuvuuksien havaitsemisessa.

LÄHTEET

Amankwah, R., Kudjo, P., Chen, J. & Towey, D. (2020). An empirical comparison of commercial and open-source web vulnerability scanners.

Anttila, P. (1998) Tutkimisen taito ja tiedonhankinta. Metodix Oy. Noudettu osoitteesta: https://metodix.fi/2014/05/17/anttila-pirkko-tutkimisen-taito-ja-tiedon-hankinta/

Australian Cyber Security Centre. (2017). Hardening Microsoft Windows 10 version 21H1 Workstations. Tarkastettu 5.6.2023 https://www.cyber.gov.au/resources-business-and-government/maintaining-devices-and-systems/system-hardening-and-administration/system-hardening/hardening-microsoft-windows-10-version-21h1-workstations

Avast. (2020). What Is EternalBlue and Why Is the MS17-010 Exploit Still Relevant? Tarkastettu 6.6.2023 https://www.avast.com/c-eternalblue

Badawy, M., El-Fishawy, N. & Elshakankiry, O. (2013). Vulnerability Scanners Capabilities for Detecting Windows Missed Patches: Comparative Study.

Baráth, J. (2017). Optimizing windows 10 logging to detect network security threats. 2017 Communication and Information Technologies (KIT).

Beale, J., Deraison, R., Meer, H., Temmingh, R., Walt, C.V.D. (2004) Nessus Network Auditing. Syngress Publishing (2004)

CBC News. (2017). "WannaCry" Ransomware attack losses could reach $4 billion. Tarkastettu 6.6.2023 https://www.cbsnews.com/news/wannacry-ransomware-attacks-wannacry-virus-losses/

Chalvatzis, I., Karras, D. A. & Papademetriou, R. C. (2019) Evaluation of Security Vulnerability Scanners for Small and Medium Enterprises Business Networks Resilience towards Risk Assessment. IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 2019, pp. 52-58, doi: 10.1109/ICAICA.2019.8873438.

DoD Cyber Exchange (2023). About the DoD Cyber Exchange. Tarkastettu 6.11.2023 https://public.cyber.mil/about-cyber/

Durve, R. & Bouridane, A. (2017). Windows 10 Security Hardening using Device Guard Whitelisting and AppLocker Blacklisting.

Esposito, D., Rennhardh, M., Ruf, L., & Wagner, A. (2018). Exploiting the potential of web application vulnerability scanning.

Gorbenko, A., Romanovsky A., Tarasyuk, O. & Biloborodov, O. (2020). From Analyzing Operating System Vulnerabilities to Designing Multiversion Intrusion-Tolerant Architectures. IEEE transactions on reliability, 69(1), 22-39. https://doi.org/10.1109/TR.2019.2897248

Help Net Security (2023). Zero-day bug exploited by attackers via macro-less Office documents (CVE-2022-30190). Tarkastettu 6.11.2023 https://www.helpnetsecurity.com/2022/05/31/cve-2022-30190-follina/

Holm, H. (2011) Performance of automated network vulnerability scanning at remediating security issues. Computers & Security, Vol. 31, Issue 2, 2012, p.164-175, https://doi.org/10.1016/j.cose.2011.12.014.

Infosec (2022). The most dangerous vulnerabilities exploited in 2022. Tarkastettu 5.6.2023 https://resources.infosecinstitute.com/topic/most-dangerous-vulnerabilities-exploited/

Lockheed Martin (2023). The Cyber Kill Chain. Tarkastettu 4.6.2023. https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html#:~:text=Developed%20by%20Lockheed%20Martin%2C%20the%20Cyber%20Kill%20Chain%C2%AE,must%20complete%20in%20order%20to%20achieve%20their%20objective.

Microsoft (2022). What Is a Zero-Day Vulnerability Exploit? Tarkastettu 19.3.2023. https://www.microsoft.com/en-us/microsoft-365-life-hacks/privacy-and-safety/zero-day-vulnerability-exploit

Microsoft (2023). Center for Internet Security (CIS) Benchmarks. Tarkastettu 4.6.2023. https://learn.microsoft.com/en-us/compliance/regulatory/offering-CIS-Benchmark

Microsoft (2023). TLS 1.0 and TLS 1.1 soon to be disabled in Windows. Tarkastettu 6.11.2023. https://techcommunity.microsoft.com/t5/windows-it-pro-blog/tls-1-0-and-tls-1-1-soon-to-be-disabled-in-windows/ba-p/3887947

Mitre (2023). MITRE ATT&CK Matrix for Enterprise. Tarkastettu 4.6.2023. https://attack.mitre.org/

Moskowitz, J. (2013). Group Policy: Fundamentals, Security and the Managed Desktop. John Wiley & Sons, Incorporated.

NIST (2017). CVE-2017-0144 Detail. Tarkastettu 6.6.2023. https://nvd.nist.gov/vuln/detail/CVE-2017-0144

Ruohonen, J., Hyrynsalmi, S. & Leppänen, V. (2016). Software Vulnerability Life Cycles and the Age of Software Products: An Empirical Assertion with Operating System Products. Advanced Information Systems Engineering Workshops. p. 207–218, 2016, doi:10.1007/978-3-319-39564-7_20.

Skinner, B. F. (1963). Operant behavior. The American psychologist, 18(8), 503-515. https://doi.org/10.1037/h0045185

Softić, J., & Vejzović, Z. (2022). Windows 10 Operating System: Vulnerability Assessment and Exploitation. https://doi.org/10.1109/INFOTEH53737.2022.9751274

Softić, J., & Vejzović, Z. (2021). Operating Systems Vulnerability - An Examination of Windows 10, macOS, and Ubuntu from 2015 to 2021.

Wack, J., Tracy, M. & Souppaya, M. (2003). Guideline on network security testing. National Institute of Standards and Technology, NIST SP 800-42 (October 2003)

Woodside, A. G. (2017) Case Study Research: Core Skills in Using 15 Genres. Vol. Second edition. Emerald Group Publishing Limited.

Yin, R. K. (2014). Case study research: Design and methods (5th edition.). SAGE