Tampereen yliopisto

Tomi Salmi

# HAAVOITTUVUUSSKANNAUKSET OSANA ORGANISAATION TIETOTURVALLISUUDEN KEHITTÄMISTÄ

Informaatioteknologian ja viestinnän tiedekunta
Diplomityö
Toukokuu 2021

# TIIVISTELMÄ

Tomi Salmi: Haavoittuvuusskannaukset osana organisaation tietoturvallisuuden kehittämistä
Tampereen yliopisto
Tietotekniikan tutkinto-ohjelma
Diplomityö
Toukokuu 2021

---

Tietoverkkorikosten määrä on ollut viime vuosina voimakkaassa kasvussa. Tietoverkkorikosten keskeinen mahdollistaja on eri tekijöistä johtuva haavoittuvuus. Tietojärjestelmien teknisten haavoittuvuuksien etsimiseen luotuja automatisoituja työkaluja kutsutaan haavoittuvuusskannereiksi. Tässä diplomityössä selvitetään, mitä haavoittuvuusskannaukset ovat, miten niitä suoritetaan ja mitä hyötyä niistä voi olla organisaatiolle. Pääpaino on IP-verkon TCP- ja UDP-pohjaisten palveluiden haavoittuvuusskannauksissa.

Työ on luonteeltaan kirjallisuusselvitys, joka jakautuu tietoteknisen haavoittuvuuden tarkasteluun sekä haavoittuvuusskannauksiin ja niistä tehtäviin havaintoihin. Työssä esitellään haavoittuvuusskannausprosessi suunnittelusta skannauksiin ja tulosten analysointiin, sekä nostetaan esimerkkien avulla esiin asioita, joihin haavoittuvuusskannauksissa tulisi kiinnittää huomiota. Työssä otetaan kantaa myös skannaustyön ulkoistamisessa huomioitaviin seikkoihin. Skannerituotteista esitellään tarkemmin Nessus Professional -haavoittuvuusskanneri.

Johtopäätöksenä voidaan todeta, että haavoittuvuusskannaukset ovat kustannustehokas keino löytää tietojärjestelmissä olevia haavoittuvuuksia. Aukkojen ja heikkouksien varhainen löytäminen ja korjaaminen torjuvat tehokkaasti tietojärjestelmiin kohdistuvia hyökkäyksiä, jotka muuten voisivat johtaa esimerkiksi tietomurtoihin ja olla riski liiketoiminnalle. Haavoittuvuusskannausten on oltava säännöllisiä, ja havaittujen puutteiden korjaustoimenpiteiden toteutumista täytyy valvoa. Lisäksi organisaatiossa on huomioitava sellaisetkin verkkomaailman uhkat, joihin tietojärjestelmiä skannaamalla ei voi vaikuttaa. Tällaisia ovat esimerkiksi sosiaalisen vaikuttamisen kautta etenevät hyökkäykset.

Avainsanat: Haavoittuvuusskannaus, haavoittuvuus, porttiskannaus, tietoturvatestaaminen, tietoverkkorikos, Nessus

# ABSTRACT

Tomi Salmi: Vulnerability scanning as a tool of developing the organisation's information security
Tampere University
Degree Programme in Information Technology
Master's Thesis
May 2021

---

The number of cybercrimes has been growing strongly in recent years. A key enabler of cybercrime is vulnerability. Automated tools created to search for technical vulnerabilities in information systems are called vulnerability scanners. This thesis examines what vulnerability scanning is, how it is utilized and what benefits it can bring to an organisation. The focus is on vulnerability scans for TCP and UDP-based services on the IP network.

This thesis is a literature review, which is divided into an examination of IT vulnerabilities and vulnerability scans and observations made from them. This thesis explores the vulnerability scanning workflow from planning to the scanning process and analysing results. Examples are used to highlight issues that should be addressed in vulnerability scans. Nessus Professional vulnerability scanner product is described in more detail.

In conclusion, vulnerability scans are a cost-effective way to find vulnerabilities in information systems. Finding and fixing vulnerabilities is an effective way to prevent and mitigate attacks that could otherwise lead to, for example, data breaches and pose a risk to business. Vulnerability scanning must be regular, and the implementation of remedial actions need to be monitored. In addition, organisation must bear in mind that not all kind of online threats can be detected by vulnerability scanners. These include, for example, social engineering attacks.

Keywords: Vulnerability scan, vulnerability, port scanning, information security testing, cybercrime, Nessus

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

# ALKUSANAT

Työelämä vei minut mukanaan jo hyvän aikaa ennen tutkintoni viimeistelemistä. Siitä huolimatta diplomityö ei missään vaiheessa päässyt kokonaan unohtumaan, vaikka aloituskynnys tuntuikin vuosien varrella nousevan ja nousevan. Lopulta sain tehtyä päätöksen työhön ryhtymisestä, ja sopivan aiheen löydyttyä korona-ajan illat pääsivät hyötykäyttöön.

Kiitokset Jukka Koskiselle työn ohjaamisesta oikeaan suuntaan, sekä kollegalleni Timo Porjamolle hyvistä kommenteista ja keskusteluista aiheen ympärillä. Lopuksi vielä erityiskiitokset vanhemmille, ystäville ja työkavereille tsemppauksesta sekä väsymättömästä kyselemisestä diplomityöni valmistumisen perään.

Helsingissä, 19.5.2021

Tomi Salmi

# SISÄLLYSLUETTELO

# KUVALUETTELO

# TAULUKKOLUETTELO

# LYHENTEET JA MERKINNÄT

| | |
|---|---|
| AWS | Amazon Web Services, Amazonin pilvipalvelukokonaisuus |
| CIA-malli | Tietoturvan käsitemalli, joka perustuu luottamuksellisuuteen, eheyteen ja saatavuuteen |
| CVE | Common Vulnerabilities and Exposures, tietoturvahaavoittuvuuksien hallintajärjestelmä |
| CWE | Common Weakness Enumeration, haavoittuvuuksien luokittelujärjestelmä |
| CVSS | Common Vulnerability Scoring System, tietoturvahaavoittuvuuksien vakavuuksien pisteytysjärjestelmä |
| DNS | Domain Name System, nimipalvelujärjestelmä |
| DoS | Denial of Service, palvelunestohyökkäys |
| ENISA | European Union Agency for Cybersecurity, Euroopan unionin verkko- ja tietoturvaviranomainen |
| FIRST | Forum of Incident Response and Security Teams, kansainvälinen tietoturvatiimien järjestö |
| FTP | File Transfer Protocol, tiedonsiirtoprotokolla |
| HTML | Hypertext Markup Language, hypertekstin merkintäkieli esimerkiksi internetsivujen kirjoittamiseen |
| HTTP | Hypertext Transfer Protocol, tiedonsiirtomenetelmä |
| IaaS | Infrastructure as a Service, infrastruktuuri palveluna, pilvipalvelun toteutustapa |
| ICMP | Internet Control Message Protocol, tietoliikenteen kontrolliprotokolla |
| IoT | Internet of Things, esineiden internet |
| IP | Internet Protocol, internetin keskeinen tietoliikenneprotokolla |
| IPv4 | IP-protokollan versio 4 |
| IPv6 | IP-protokollan versio 6 |
| ISMS | Information Security Management System, tietoturvallisuuden hallintajärjestelmä |
| ISO | International Organization for Standardization, kansainvälinen standardisoimisjärjestö |
| MFA | Multi-Factor Authentication, monivaiheinen tunnistautuminen |
| NASL | Nessus Attack Scripting Language, Nessus-skannerin skriptikieli |
| NCSC | National Cyber Security Centre, tässä työssä Britannian kyberturvallisuuskeskus |
| NIST | National Institute of Standards and Technology, yhdysvaltalainen standardivirasto |
| NTP | Network Time Protocol, protokolla aikatiedon välittämiseen |
| PCI DSS | Payment Card Industry Data Security Standard, korttimaksamisen standardi |
| PoC | Proof of Concept, uuden menetelmän tai prototyypin esittely |
| RFC | Request for Comments, IETF-järjestön julkaisemia internet-standardeja ja -suosituksia |
| SCADA | Supervisory Control and Data Acquistion, etähallittavat automaatiolaitteet |
| SIEM | Security Information and Event Management, tietoturvatiedon havainnointijärjestelmä |
| SMTP | Simple Mail Transfer Protocol, sähköpostiprotokolla |
| SSH | Secure Shell Protocol, salatun tietoliikenteen protokolla |
| TCP | Transmission Control Protocol, yhteydellinen tietoliikenneprotokolla |
| UDP | User Datagram Protocol, yhteydetön tietoliikenneprotokolla |
| VPR | Vulnerability Priority Rating, Tenablen käyttämä tietoturvahaavoittuvuuksien vakavuuksien pisteytysjärjestelmä |

# 1. JOHDANTO

Nykyaikaisessa tietoteknisessä ympäristössä kaikki verkkoon kytketyt laitteet ovat potentiaalisia kohteita verkosta tuleville hyökkäyksille. Niin sanottu hyökkäyspinta-ala on laajentunut nopeasti, kun internetiin on liitetty tavanomaisten kotitietokoneiden ja palvelinten lisäksi viime vuosina muun muassa runsaasti kodintekniikkaa ja kiinteistöautomaatiota. Tämä tarkoittaa, että hyökkääjillä on yhä enemmän kohteita, ja toisaalta laitteiden omistajilla, ylläpitäjillä ja käyttäjillä on yhä enemmän suojattavaa.

Järjestelmissä olevat haavoittuvuudet altistavat ne hyökkäyksille. Haavoittuvuus voi olla seurausta esimerkiksi ohjelmointivirheestä tai järjestelmän väärin tai puutteellisesti suoritetusta käyttöönotosta. Haavoittuvuudet ovat riski liiketoiminnalle, sillä niiden hyväksikäyttö voi johtaa tietomurtoihin tai muihin häiriöihin tai keskeytyksiin yrityksen jokapäiväisessä toiminnassa. Ongelmien välttämiseksi on tärkeää löytää ja korjata aukot ennen kuin rikolliset löytävät ne ja hyödyntävät niitä. Tämän vuoksi tietoturvatestaaminen, kuten haavoittuvuuksien etsiminen, on tärkeää.

Jotta haavoittuvuudet olisi mahdollista korjata, ne täytyy ensin löytää. Hyväntahtoista haavoittuvuuksien etsimistä kutsutaan valkohattuhakkeroinniksi, ja haavoittuvuuksien etsimiseen tehtyjä automatisoituja työkaluja haavoittuvuusskannereiksi. Tässä kirjallisuusselvityksenä tehdyssä diplomityössä selvitetään, mitä haavoittuvuusskannaukset ovat, miten niitä suoritetaan ja mitä hyötyä niistä voi olla organisaatiolle. Tämän työn pääpaino on IP-verkon TCP- ja UDP-pohjaisten palveluiden haavoittuvuusskannauksessa.

Aluksi työssä selvitetään, keitä hyökkääjät tietoverkoissa ovat, ja mitkä ovat heidän motiivinsa. Suojautuminen on helpompaa, kun tuntee hyökkääjien toimintatapoja ja tietää, miltä suojautua. Lisäksi käydään läpi haavoittuvuuden käsitettä sekä siihen läheisesti liittyviä termejä ja mittareita, joihin myöhemmissä luvuissa viitataan.

Kolmannessa luvussa on porttiskannausten yleiskuvaus sekä yleisimpiä skannausmenetelmiä ja järjestelmien tunnistuskeinoja. Haavoittuvuusskannausprosessin läpivienti esitellään alkuvalmisteluista skannauksiin ja tulosten analysointiin. Skannausprosessin eri vaiheista nostetaan esiin asioita, joihin tulisi kiinnittää tarkempaa huomiota. Viime vuosina verkoissa voimakkaasti yleistyneiden IoT- ja automaatiolaitteiden erityispiirteet huomioidaan omassa luvussaan. Skanneriohjelmistoista tutustutaan tarkimmin Nessus Professional -haavoittuvuusskanneriin.

Neljännessä luvussa arvioidaan haavoittuvuusskannausten merkitystä ja hyötyjä organisaatiolle. Mietitään myös, miten muilla tavoin voisi saada tietoa oman verkon haavoittuvuuksista. Lisäksi pohditaan muun muassa, mitä tulisi ottaa huomioon, jos skannausten suorittaminen ulkoistetaan, ja minkä kaltaisia uhkia voi jäädä huomaamatta huolellisesti suoritetuista skannauksista huolimatta. Viimeisenä lukuna ennen lähdeluetteloa on tiivis yhteenveto työstä.

# 2. TIETOTURVA JA HAAVOITTUVUUS

Kansainvälisen tietoturvastandardin ISO/IEC 27000 (2018) määritelmän mukaan tietoturvallisuus tarkoittaa luottamuksellisuutta, eheyttä ja saatavuutta. Tässä luvussa käydään läpi näitä sekä muita tietoturvaan ja tietoteknisten järjestelmien haavoittuvuuksiin läheisesti liittyviä termejä, käsitteitä ja mittareita. Luku luo pohjaa tämän diplomityön tutkimusaiheelle, eli verkossa suoritettaville haavoittuvuusskannauksille.

Tunnettu tietoturva-asiantuntija Gene Spafford sanoi jo 1990-luvulla: "*The only truly secure system is one that is powered off, cast in a block of concrete and sealed in a lead-lined room with armed guards - and even then I have my doubts.*" (Andress 2014)

Kuten Spafford toteaa, tietyssä pisteessä järjestelmän sisältämien tietojen voidaan katsoa olevan turvassa ulkopuolisilta. Tietoturva on kuitenkin aina kompromissi käytettävyydelle; kun turvallisuusvaatimuksia nostetaan, käytettävyys yleensä kärsii. Suojautumisessa voidaan aina ottaa edistysaskelia. Sen sijaan on vaikea kuvitella tilannetta, jossa voitaisiin todeta, että oma ympäristö on nyt varmuudella täysin turvallinen. On myös tärkeää muistaa, että elämme muuttuvassa maailmassa, jossa kerran turvalliseksi todettu ympäristö ei välttämättä ole sitä enää huomenna, eikä varsinkaan kuukauden päästä tai ensi vuonna.

## 2.1 Haavoittuvuus mahdollistaa tietoverkkorikoksen

Tietoverkoissa tapahtuu koko ajan erilaisia rikoksia, kuten tietomurtoja, palvelunestohyökkäyksiä ja haittaohjelmien levittämistä. Tietoverkoissa tapahtuvaa ja tietoverkkoja hyödyntävää rikollisuutta kutsutaan tietoverkkorikollisuudeksi tai kyberrikollisuudeksi (Poliisi 2021). Tietoverkkorikoksen suurin mahdollistaja on eri tekijöistä johtuva haavoittuvuus.

Tässä luvussa selvitetään, mitä ovat tyypilliset tietoverkkoihin kohdistuvat rikokset, ketkä niitä toteuttavat ja mitkä ovat tekijöiden motiivit. Näiden asioiden tuntemus auttaa ymmärtämään haavoittuvuuksien löytämisen ja korjaamisen tärkeyttä.

Suomessa termi "kyberrikos" on vakiintunut yleiseen käyttöön niin median kuin viranomaistenkin keskuudessa. Poliisi käyttää kyberrikoksissa jakoa tietoverkkosidonnaisiin ja tietoverkkoavusteisiin rikoksiin (Poliisi 2021). Tietoverkkosidonnaisia rikoksia ovat rikokset, jotka kohdistuvat tietoverkkoihin ja tietojärjestelmiin, eli ne ovat rikoksia, joita ei voisi tehdä ilman tietoteknologiaa. Tietoverkkoavusteisia rikoksia sen sijaan ovat rikokset, joissa tietoverkkoa käytetään apuvälineenä rikokseen, kuten esimerkiksi

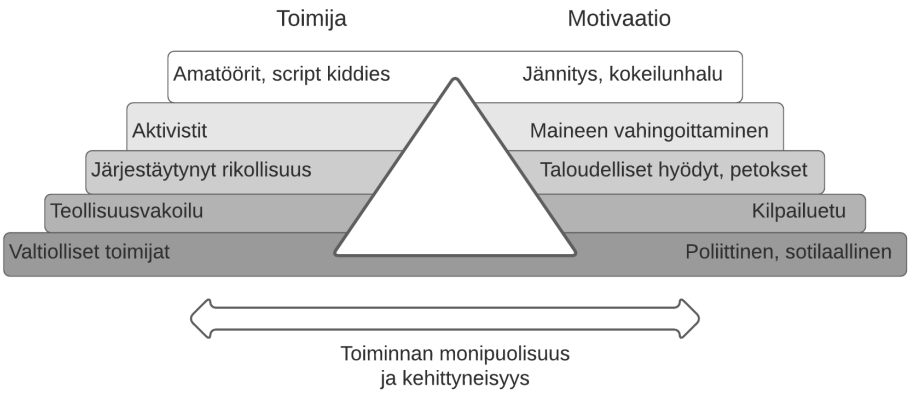
erilaisiin petoksiin, rahanpesuun tai huumausainekauppaan. Tässä diplomityössä sivuttavat tietoturvaloukkaukset, kuten tietojärjestelmässä olevan haavoittuvuuden hyväksikäyttäminen rikolliseen toimintaan, kuuluvat ensiksi mainittuun tietoverkkosidonnaisten rikosten ryhmään.

Verizonin vuoden 2020 Data Breach Investigations Reportin (2021) ja Trend Micron vuoden 2020 vuosiraportin (2021) mukaan suurin osa organisaatioihin kohdistuneista tietoturvaloukkauksista vuonna 2020 oli lähtöisin kohdeorganisaatioiden ulkopuolelta. Sekä Verizon että Trend Micro analysoivat ja tilastoivat vuosittain maailmanlaajuisesti kymmeniätuhansia tietoturvaloukkauksia.

Organisaation sisältä tulevat kyberuhkat ovat lähteen Limnéll et al. (2014) mukaan lähtöisin pääasiassa tyytymättömistä entisistä ja nykyisistä työntekijöistä tai kumppaneista, alihankkijoista ja näiden työntekijöistä. Syvällistä tuntemusta tietojärjestelmiin tunkeutumisesta ei tällöin yleensä vaadita, vaan henkilöllä on jo valmiiksi riittävät käyttöoikeudet tietojärjestelmiin tai hän pystyy hankkimaan sellaiset. Tekninen tietoturva ei tunnista tällaista tapahtumaa hyökkäykseksi, vaan normaaliksi järjestelmäresurssien käytöksi. Sisäisen uhkan muodostavat myös työntekijät, joiden mukana järjestelmiin voi päästä muun muassa haittaohjelmia esimerkiksi huolimattomuuden tai tietämättömyyden vuoksi.

Kyberuhkien takana on erilaisia tekijöitä ja motiiveja. Tekijät vaihtelevat harrastelijamaisista amatööreistä aktivisteihin sekä yritysten ja valtioiden tukemaan tai suorittamaan vakoiluun sekä järjestäytyneeseen rikolliseen toimintaan (Limnéll et al. 2014). Yleinen mielipide on, että tänä päivänä kyberrikollisten suurin motiivi on raha. Valtiotason toiminnassa taustalla voi olla myös poliittisia ja sotilaallisia tavoitteita. (Limnéll et al. 2014; Verizon 2021)

Kuva 1 esittää Scullyn (2011) näkemyksen kyberrikosten tekijöistä sekä heidän tyypillisistä tavoitteistaan ja motivaatioistaan. Kolmiomalli esittää toiminnan monipuolisuutta ja laajuutta. Vähiten kehittyneitä ja yksinkertaisimpia hyökkäyksiä ovat amatöörien tai niin sanottujen script kiddien toimet, joilla lähinnä testaillaan omia kykyjä ja haetaan jännitystä. Toisessa ääripäässä ovat valtiolliset toimijat, joiden toiminta on hyvin päämäärätietoista, ja resurssit ovat suuret.




***Kuva 1.*** *Tietoverkkorikosten tekijät ja heidän motiivinsa (mukaillen lähdettä Scully 2011).*

Kyberrikoksiin liittyy usein kansainvälisyys. Tämä on luonnollista, koska tietoverkoissa ei ole samanlaisia maarajoja tai muita fyysisiä rajoitteita, kuin perinteisissä rikoksissa. Kansainvälisyys vaikeuttaa rikosten torjuntaa ja selvittämistä. Rikokset saavat koko ajan uusia ilmentymismuotoja, mutta toisaalta vanhat muodot säilyvät. Osaltaan tästä johtuen suojautumiseen joudutaan käyttämään koko ajan enemmän resursseja, ja kyberhyökkäysten kustannukset maailman taloudelle ovat olleet viime vuosina voimakkaassa kasvussa. Tietoturvayhtiö McAfee arvioi kyberrikollisuuden maailmanlaajuisiksi kustannuksiksi vuonna 2018 noin 600 miljardia dollaria, ja vuonna 2020 jo 945 miljardia dollaria. McAfeen mukaan osa lisääntyneistä kustannuksista selittyy aiempaa kattavammalla raportoinnilla, mutta ennen kaikkea syynä on rikollisten aiempaa kehittyneemmät ja tehokkaammat menetelmät. (McAfee 2021)

Suomen osalta ei löydy arvioita kokonaiskustannuksista, mutta Suomen Kyberturvallisuuskeskuksen tilannekeskuksen käsittelemien tapausten määrä yli kaksinkertaistui vuonna 2020 edelliseen vuoteen nähden (Kyberturvallisuuskeskus 2021). Suorien taloudellisten menetysten lisäksi kyberrikollisuus aiheuttaa vahinkoja ja menetyksiä muun muassa luottamuksellisen tiedon menetyksinä, tietojärjestelmien ja -verkkojen toipumisesta ja korjaamisesta aiheutuneina kuluina, mainehaittoina, menetettyinä tilauksina sekä asiakkaille ja yhteistyökumppaneille suoritettuina korvauksina (Limnéll et al. 2014). Vahingot voivat siis olla paitsi mittavat, myös kaikkine kerrannaisvaikutuksineen mahdottomat laskea.

Verkkorikollisten ansaintalogiikat vaihtelevat. Internetin niin sanotussa mustassa pörssissä ostetaan, myydään ja vaihdetaan muun muassa varastettua tietoa sekä verkkorikollisuudessa hyödynnettäviä työkaluja. Eniten tarjoavalle myydään esimerkiksi huonosti suojatusta verkosta varastettuja asiakastietokantoja tai salaisiksi määriteltyjä

dokumentteja. Haittaohjelmien tekijät myyvät saastuttamiensa koneiden resursseja muuhun käyttöön, esimerkiksi osaksi bottiverkkoa ja palvelunestohyökkäyksiä. On olemassa myös tahoja, joilta voi ostaa murtopalveluita tai palvelunestohyökkäyksiä haluamaansa kohteeseen (Makrushin 2017). Rikollisten kaupankäynnissä verkossa maksut suoritetaan usein kryptovaluutoilla.

Verkossa houkutus rikolliseen toimintaan kasvaa, koska tunnistamisen ja kiinnijäämisen riski pienenevät verrattuna fyysiseen maailmaan. Useasti toimijan IP-osoite pystytään jäljittämään, mutta yhteyden todellisen käyttäjän selvittäminen voi olla mahdotonta, jos tekijä osaa peitellä jälkiään. Nykyaikainen kybermaailma mahdollistaa useat identiteetit sekä tunnistamattomana toimimisen, ja yleensä tunnistamattomana pysyminen tukee rikollisten tavoitteita.

Kuka tahansa voi joutua kyberrikoksen uhriksi. Uhreja ovat niin yksityiset henkilöt kuin erilaiset organisaatiot ja valtiotkin. Kohteiksi kelpaavat niin pienet kuin suuretkin, pienyrityksistä monikansallisiin yhtiöihin. Kuluttajiin kohdistuvia rikoksia ovat esimerkiksi identiteettivarkaudet ja luottokorttipetokset. Yrityksiin kohdistuvissa rikoksissa onnistumisen tuotto voi olla kuluttajarikoksiin verrattuna merkittävästi suurempi. (Ponemon Institute 2020; Trend Micro 2021; Verizon 2021)

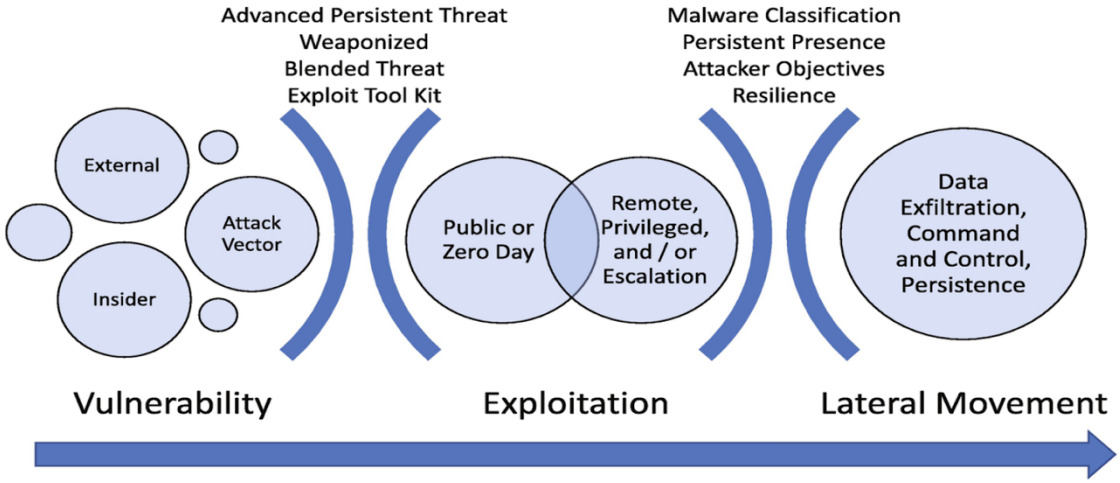
Tietoturva on jatkuvaa kilpajuoksua kyberrikollisia vastaan. Organisaatioiden on elettävä ajassa ja pystyttävä reagoimaan muuttuviin haasteisiin. Hyvä esimerkki nopeasti muuttuvasta tilanteesta on koronapandemian vuoksi vuonna 2020 voimakkaasti lisääntynyt etätyö, minkä seurauksena julkiseen internetiin ilmestyi runsaasti muun muassa yritysten tiedostopalvelimia ja muita kohteita, joiden turvallisuusratkaisut olivat puutteellisia (A-studio 2021). Suojelupoliisin (2020) mukaan koronapandemia siirsi kybervakoilun painopistettä perinteisestä henkilötiedustelusta verkkoon muun muassa vähentyneen matkustamisen seurauksena.

### 2.1.1 Tietomurron eteneminen

Tässä luvussa käydään lyhyesti läpi tyypillisen tietomurron eteneminen vaiheittain yleisellä tasolla Mitren ATT&CK-viitekehystä (Mitre 2021a) mukaillen. Tavoitteena on havainnollistaa, miksi haavoittuvuuksien paikantaminen ja korjaaminen on tärkeää, ja mitä hyökkääjä voi saada aikaan haavoja hyödyntämällä.

Vaikka eri kyberrikokset eroavat toisistaan merkittävästi, tietomurrot noudattavat tyypillisesti samaa kaavaa. Tyypillisen tietomurron eteneminen (Kuva 2) koostuu neljästä päävaiheesta:

1. Tiedustelu ja haavoittuvuuden löytäminen - *Vulnerability*
2. Tunkeutuminen haavaa hyväksikäyttämällä - *Exploitation*
3. Jalansijan laajentaminen kohdeympäristössä - *Lateral Movement*
4. Toteutus, esim. tietojen varastaminen (ei kuvassa)




***Kuva 2.*** *Tyypillisen tietomurron eteneminen (Haber & Hibbert 2018).*

Ensimmäisessä vaiheessa hyökkääjä etsii erilaisia tunkeutumisväyliä. Yksinkertaisimmillaan tämä tapahtuu esimerkiksi kohdistamalla verkkoon automatisoidusti porttiskannauksia, joilla etsitään avoimia palveluita suuresta kohdemassasta. Etsintä voi olla myös kohdistetumpaa, jolloin selvitetään yksityiskohtia juuri tietystä kohdeorganisaatiosta, sen asiakkaista, yhteistyökumppaneista, infrastruktuurista, henkilöstöstä ja ylipäänsä kaikesta, jolla voisi olla merkitystä hyökkäyksen toteuttamisessa. Hyökkääjää kiinnostavia teknisiä tietoja ovat esimerkiksi verkkoinfrastruktuurin verkkolaitteet, palvelimet ja käytössä olevat ohjelmistot. Tietoa on saatavilla muun muassa kohdeverkkoon kohdistetuilla skannauksilla, mutta myös internetin avoimista ja julkisista lähteistä kuten kohteen verkkosivuilta, hakukoneista ja nimipalvelusta. (Mitre 2021a) Eräs tiedonlähde on Shodan-hakukone (ks. luku 4.4.2), joka indeksoi jatkuvasti koko internetin IP-osoiteavaruutta.

Tunkeutumisväylä saatetaan löytää myös henkilöstön kautta. Monesti sanotaan, että ihminen on tietoturvan heikoin lenkki (Peltier 2006; Purushotham & Gowthamaraj 2019). Ihmiset ovat huijattavissa, ja kiireessä hyvätkin turvallisuusohjeet saattavat unohtua tai niitä ei löydetä. Tyypillinen tapa huijata käyttäjätunnuksia on tietojenkalastelu, eli *phishing*, jossa käyttäjä huijataan esimerkiksi sähköpostiviestin avulla väärennetylle kirjautumissivulle. Sivu muistuttaa aitoa, käyttäjälle entuudestaan tuttua kirjautumissivua, mutta onkin huijaussivu, jonka avulla hyökkääjä kerää esimerkiksi sähköpostipalveluiden tai VPN-etäyhteyksien käyttäjätunnuksia ja salasanoja.

Vaihtoehtoisesti sivulta voi asentua käyttäjän koneeseen haittaohjelma, jonka avulla hyökkääjä saa pääsyn kohdeverkkoon. Haittaohjelman ujuttaminen yrityksen sisäverkkoon voi onnistua niinkin helposti kuin lähettämällä sähköposti, jonka liitetiedoston joku henkilöstöstä ajaa.

Laajempi sosiaalisen vaikuttamisen käsite on *social engineering*, jolla tässä kontekstissa tarkoitetaan erilaisia sosiaalisen manipuloinnin vaikutusyrityksiä, joiden kohteena on tietojärjestelmän käyttäjä, ja tavoitteena luvaton pääsy tietojärjestelmään tai luottamukselliseen informaatioon. Peltierien mukaan (2006) tällaiset hyökkäykset perustuvat tyypillisesti ihmisten auttamisen haluun, laiskuuteen ja luontaiseen taipumukseen luottaa toisiin ihmisiin. Perinteinen hyökkäyskeino on salasanojen kerääminen IT-tuen työntekijäksi tekeytymällä. Social engineering -hyökkäys toteutetaan tavallisesti nelivaiheisena: taustalla tapahtuva tiedonkeruu, luottamuksen saavuttaminen, hyväksikäyttö ja jälkien peittäminen. (Purushotham & Gowthamaraj 2019)

Kun tunkeutuja on päässyt kohdeympäristöön, on usein tavoitteena hankkia ylläpitäjän oikeudet, tunkeutua syvemmälle sisäverkkoon ja ajaa siellä omaa ohjelmakoodia. Yleensä tietomurrot edellyttävät jonkin komentokanavan hyödyntämistä. Komentokanavaa käytetään tunkeutumisen laajentamiseen ja tiedon varastamiseen. Omien jälkien siivoaminen esimerkiksi lokitiedostoista sekä tietoturvamekanismien sulkeminen vaikeuttavat jäljitystä. Jos tavoitteena on saada aikaan pidempikestoinen pääsy verkkoon, asennetaan takaportteja ja komentokanavia. Lopulta hyökkääjä kerää ja siirtää varastettavat tiedot tai esimerkiksi saattaa kohdejärjestelmän toimintakyvyttömäksi. (Mitre 2021a)

Kiristyshuijaukset ovat yleistyneet voimakkaasti viime vuosina. Kun kiristyshaittaohjelma (*ransomware*) on saatu sisälle yrityksen sisäverkkoon, ohjelma käynnistyy ja salaa löytämänsä tiedostot, eli tekee niistä käyttökelvottomia. Tämän jälkeen rikollinen suostuu purkamaan salauksen ja palauttamaan tiedostot vain lunnaita vastaan. Tunnettuja lunnastroijalaisia ovat esimerkiksi WannaCry ja CryptoLocker. (Kaspersky 2021) Myös palvelunestohyökkäyksiä on käytetty pitkään osana kiristysrikoksia. Tyypillisesti kohteiksi valitaan sellaisia verkkopalveluita, joille keskeytykset ovat erityisen kalliita, kuten verkkokauppoja ja vedonlyöntipalveluita. Alas ajetuilla palveluilla pyritään aiheuttamaan myös mainehaittoja.

### 2.1.2 Suojattava tieto ja tiedon ominaisuudet

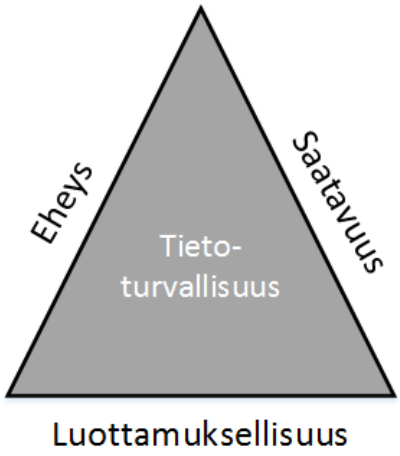
Suojattava tieto voi olla esimerkiksi yrityksen liiketoiminnallisia tietoja tai henkilötiedoiksi luokiteltavia tietoja työntekijöistä ja asiakkaista. Tiedolla on omistajalleen jokin merkitys

ja arvo, ja tiedon suojaaminen tukee tämän arvon säilymistä. Tieto voi olla arvokasta paitsi organisaatiolle itselleen, myös muille osapuolille, kuten asiakkaille. Mitä tietointensiivisemmästä toimialasta on kyse, sitä tärkeämpää liiketoiminnan kannalta merkittävän tiedon turvaaminen on. (Sipior & Ward 2008)

Seuraavassa on käyty läpi tietoturvan kolme keskeistä peruskäsitettä CIA-tietoturvamallin avulla. Tiedon ominaisuuksien pohtiminen auttaa ymmärtämään sen suojaustarpeita. Käsitteet on hyvä tuntea, koska niihin viitataan usein tiedon suojaustoimenpiteistä puhuttaessa, myös tämän työn myöhemmissä luvuissa.

### CIA-malli

Tietoturvan peruskäsitteistä puhuttaessa käytetään usein niin sanottua CIA-mallia. CIA-kolmion (Kuva 3), tietoturvallisuuden kolminaisuuden, kirjaimet tulevat sanoista *Confidentiality* (luottamuksellisuus), *Integrity* (eheys) ja *Availability* (saatavuus). Näitä ominaisuuksia voidaan pitää tietoturvan kolmena keskeisimpänä periaatteena. Joissain lähteissä samasta mallista käytetään nimeä CAI-malli luultavasti, jotta ei tapahtuisi sekoittumista yhdysvaltalaiseen tiedustelupalveluun. (Andress 2014) Seuraavaksi esitellään CIA-mallin muodostavat kolme tekijää.




***Kuva 3.*** *CIA-kolmio kuvaa tietoturvallisuuden kolminaisuutta.*

### Luottamuksellisuus

Tiedon luottamuksellisuus perustuu rajoituksiin tietoon pääsyssä. Luottamuksellisuus takaa, että tieto on vain niiden tahojen käsiteltävissä, joilla on siihen oikeus esimerkiksi työtehtäviensä hoitamisen vuoksi. Käyttöoikeudet määrittelee tiedon omistaja. Luottamuksellisuuden turvaaminen kattaa sekä tiedon tahallisen että tahattoman valtuudettoman käytön. Tahattomasti luottamuksellisuus voi kärsiä, jos esimerkiksi

sähköpostiviesti lähetetään vahingossa väärälle vastaanottajalle. (Andress 2014; Reid & Gilbert 2010)

Jotta luottamuksellisuus pystytään toteuttamaan, on tiedonkäsittely-ympäristön tarjottava riittävät rajoitusmekanismit. Luottamuksellisuutta vahvistetaan salauksella sekä pääsyrajoituksilla kuten salasanoilla, pääsylistoilla ja muilla todennusmenetelmillä. Lisäksi käytetään fyysisiä kontrolleja kuten lukkoja ja kulunvalvontaa, joilla rajoitetaan ihmisten pääsyä tiloihin ja laitteisiin, joissa tietoa säilytetään tai käsitellään. (Andress 2014)

**Eheys**

Eheys tarkoittaa, että tieto ei ole muutettavissa tai poistettavissa silloin kun käsittelijällä ei ole siihen oikeuksia. Ei-halutut, luvattomat muutokset tahdotaan estää. Käsittelijöiden pitää voida luottaa, että tieto ei ole muuttunut hallitsemattomasti tai valtuudettomasti. (Andress 2014; Reid & Gilbert 2010)

Eheyden varmistaminen on erityisen tärkeää aina, kun tietoa siirretään järjestelmästä tai verkosta toiseen. Eheys voi kärsiä paitsi tahallisen toiminnan seurauksena, myös esimerkiksi laiterikon tai huonolaatuisten verkkoyhteyksien vuoksi. Syystä riippumatta tiedon muuttuminen pitää pystyä havaitsemaan.

Tiedon eheys halutaan varmistaa ennen kaikkea estämällä vihamielisen tahon tekemät muutokset ja poistot. Lisäksi täytyy huomioida inhimillisen virheen mahdollisuus, eli mahdollistaa tiedon palautus siinäkin tapauksessa, että muutos oli sallittu, mutta tapahtui vahingossa. Palautukset tehdään varmuuskopioista, ja tiedon muuttumisen havaitsemiseen käytetään esimerkiksi tarkistussummia. (Andress 2014)

**Saatavuus**

Saatavuus tarkoittaa pääsyä tietoon silloin kun tietoa tarvitaan, ja kun siihen on oikeus. Puhutaan myös käytettävyyden käsitteestä. (Reid & Gilbert 2010) Saatavuus kärsii, jos sen mahdollistava polku, esimerkiksi verkkoyhteys kotikoneelta etäkäytettävään palvelimeen katkeaa tahallisesti tai tahattomasti. Saatavuus voi vaarantua tai estyä esimerkiksi ohjelmistovirheen, laiterikon tai sähkökatkon takia. Ulkopuolelta tulevia, tahallisia saatavuutta estäviä hyökkäyksiä kutsutaan usein palvelunestohyökkäyksiksi (*Denial of Service,* DoS). (Andress 2014)

CIA-mallia voi hyödyntää muistikaavana esimerkiksi uuden tietojärjestelmän suunnittelussa, toteutuksessa ja käyttöönotossa. Jokaisen järjestelmän tietoturvan kannalta on keskeistä, että kaikki kolme elementtiä tulevat huomioiduiksi. On hyvä tiedostaa myös

mallin yksinkertaisuudesta seuraavat puutteet. Malli keskittyy voimakkaasti dataan ja sen suojaamiseen, mutta teknologisena mallina se ei huomioi tietoturvallisuuden inhimillistä ulottuvuutta.

## 2.2 Tietotekninen haavoittuvuus

Tässä luvussa käydään läpi, mitä tietoteknisessä ympäristössä tarkoitetaan haavoittuvuudella sekä muutamalla siihen läheisesti liittyvällä käsitteellä ja termillä. Alaluvuissa esitellään haavoittuvuuksien hallintaan, luokitteluun ja vakavuusarviointiin luotuja järjestelmiä.

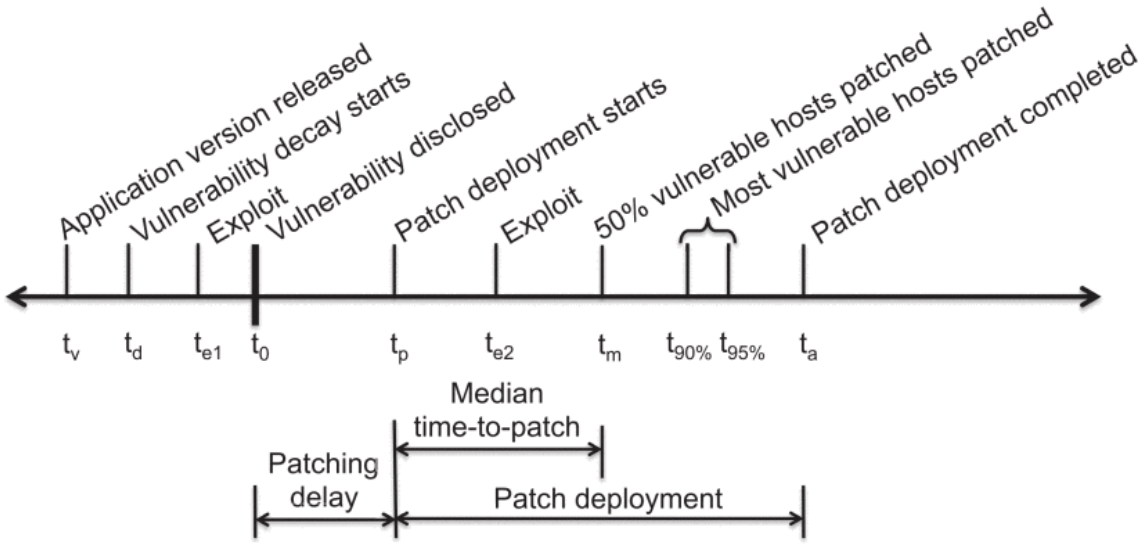
Haavoittuvuudella tässä diplomityössä tarkoitetaan ensisijaisesti teknistä, tietojärjestelmässä olevaa haavoittuvuutta. Yhdysvaltalainen NIST eli *National Institute of Standards and Technology* (2020) määrittelee haavoittuvuuden hyökkäyksen mahdollistavaksi heikkoudeksi tietojärjestelmässä, järjestelmän turvamenetelmissä, sisäisissä kontrolleissa tai näiden implementoinnissa.

Suomalainen Sanastokeskus (2020) määrittelee haavoittuvuuden seuraavasti: "*Haavoittuvuus voi olla mikä tahansa heikkous, joka mahdollistaa vahingon toteutumisen tai jota voidaan käyttää vahingon aiheuttamisessa. Haavoittuvuuksia voi olla tietojärjestelmissä, prosesseissa ja ihmisen toiminnassa. Esimerkiksi ohjelmistossa voi olla haavoittuvuus, joka mahdollistaa järjestelmän väärinkäytön.*"

Haavoittuvuus on jäänyt järjestelmään sen rakennusvaiheessa, tai se voi syntyä myöhemmin järjestelmää päivitettäessä tai muutettaessa. Tavallinen syy haavoittuvuudelle on ohjelmointivirhe tai virhe esimerkiksi tietoliikenneprotokollan implementoinnissa. Virhe on voitu rakentaa järjestelmään myös piilotettuna ominaisuutena tahallisesti, jolloin on kyse takaportista. Järjestelmässä oleva haavoittuvuus on hyökkääjien hyväksikäytettävissä, jos sille on tiedossa hyväksikäyttömenetelmä.

Kuva 4 esittää haavoittuvuuden elinkaarta sovelluksen tai sen uuden version julkaisun ajanhetkestä $t_v$ alkaen. Aikajanalta on poimittavissa muutama keskeinen ajankohta tai -jakso. Ajanhetket eivät välttämättä esiinny juuri kuvatussa järjestyksessä. Haavoittuvuuden julkisuuteen tuomisen ja korjauspäivityksen julkaisun välissä on mahdollisesti ajanjakso (kuvassa *patching delay*), jona aikana haavoittuvuuteen ei ole olemassa korjausta. Nollapäivähaavoittuvuudella tarkoitetaan sellaista haavoittuvuutta, joka on löydetty, mutta johon ei ole julkaistu korjausta (ENISA 2021). Uuden haavan löytänyt vastuullinen, hyväntahtoinen tietoturvatutkija tekee ennen haavan julkaisua yhteistyötä ohjelmistovalmistajan kanssa, ja antaa tälle aikaa kehittää ja julkaista virallinen korjaus.

Tyypillisten loppukäyttäjäsovellusten haavoittuvuuksia ja korjauspäivityksiä tutkineiden Nappa et al. (2015) mukaan 92 % julkaistuista haavoittuvuuksista saa virallisen korjauspäivityksen 30 vuorokauden sisällä haavan julkaisusta. Aikajanalla $t_m$ tarkoittaa ajanhetkeä, jolloin puolet haavoittuvista järjestelmistä on korjattu. Tutkimuksen mukaan hajonta on suurta, mutta mediaanikestoksi mitattiin 45 vuorokautta. Nopeimmin päivitetään ohjelmistot, joissa automaattipäivitys on laajasti käytössä. Merkille pantavaa on, että vain 28 % korjauspäivityksistä saavutti 95 prosentin kattavuuden viiden vuoden tarkastelujakson aikana.




***Kuva 4.*** *Haavoittuvuuden elinkaari esitettynä aikajanalla (Nappa et al. 2015).*

Paikkaamattomien haavoittuvuuksien suuri määrä tukee Weidmanin (2014) havaintoa, jonka mukaan suurimmassa osassa onnistuneita hyökkäyksiä on hyödynnetty muita kuin nollapäivähaavoittuvuuksia. Toisin sanoen tietoturva-aukko olisi ollut paikattavissa, ja hyökkäyksen toteutus estettävissä.

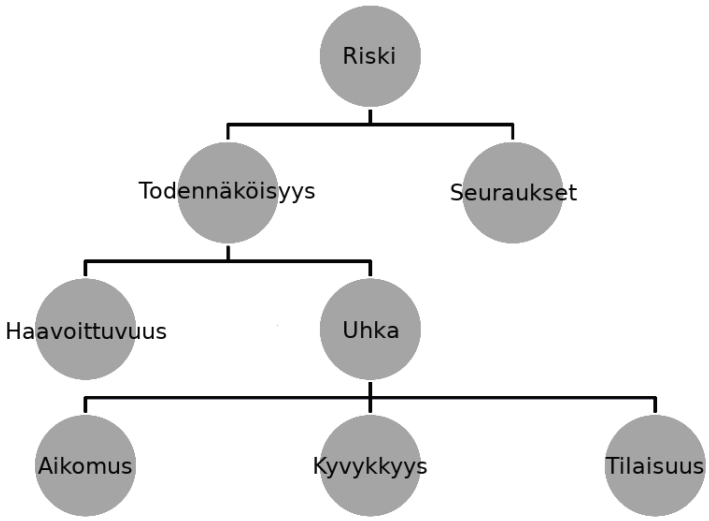
Uusien haavoittuvuuksien löytäminen on usein kallista. Euroopan unionin Kyberturvallisuusvirasto ENISA:n (*European Union Agency for Cybersecurity*) mukaan jotkut järjestelmätoimittajat, valtioiden hallinnot sekä rikollisorganisaatiot ovat valmiita maksamaan nollapäivähaavoittuvuuksien löytäjille suuria summia. Muun muassa tämä motivoi alan tutkijoita ja asiantuntijoita etsimään haavoittuvuuksia. Ohjelmiston tai järjestelmän käyttäjäorganisaatiolle sekä asiakkaalle paljastunut nollapäivähaavoittuvuus on merkittävä tietoturvariski. Vastaavasti ohjelmistotoimittaja kohtaa ainakin maine- ja liiketoimintariskejä. Järjestelmäylläpitäjältä nollapäivähaavoittuvuus edellyttää korostunutta tarkkaavaisuutta sekä harkintaa rajoitustoimenpiteisiin ryhtymiseksi. (ENISA 2021) Rikolliselle uuden haavoittuvuuden löytyminen on uusi ansaintamahdollisuus.

Kun haavoittuvuuden hyväksikäyttömenetelmä (*proof of concept,* PoC) julkaistaan, alkaa rikollisten kilpailu menetelmän automatisoinnista. Yleensä kohdennettu hyväksikäyttö alkaa hyvin nopeasti haavoittuvuuden julkaisun jälkeen, ja laajamittainen automatisoitu hyväksikäyttö usein viikossa tai kahdessa riippuen haavan monimutkaisuudesta.

Järjestelmän omistajalle haavoittuvuuksien vähentäminen on ensiarvoisen tärkeää. Jotta haavoittuvuudet voitaisiin korjata, ne täytyy ensin löytää, ja etsimisessä voidaan käyttää tämän työn aiheena olevia haavoittuvuusskannauksia. Laajemmin ajateltuna haavoittuvuusskannausprosessi sisältää skannausten lisäksi myös löydettyjen haavoittuvuuksien raportoinnin ja korjaamisen. Jotta järjestelmät voitaisiin jo lähtökohtaisesti tehdä turvallisiksi, tulisi suunnittelussa ja ohjelmoinnissa hyödyntää turvallisia suunnitteluperiaatteita ja turvallisuusarkkitehtuureja.

Haavoittuvuuden käsitteeseen liittyy läheisesti uhkan ja riskin käsitteet. Haavoittuvuus ja uhka muodostavat riskin, ja uhkista aiheutuu vahinkoa niiden hyödyntäessä järjestelmissä olevia haavoittuvuuksia. Littlen ja Rogovan mukaan (2006) uhka ja sen suuruus muodostuvat kolmesta tekijästä: aikomus (*intention*), kyvykkyys (*capability*) ja tilaisuus (*opportunity*). Aikomuksella tarkoitetaan hyökkääjän tahtotilaa tai suunnitelmaa toteuttaa ja päästä haluamiinsa tavoitteisiin. Aikomus edustaa uhkan psykologista osa-aluetta, jota kyvykkyys ja avautuva tilaisuus voimakkaasti ohjaavat. Kyvykkyys tai kapasiteetti viittaa hyökkääjän käytettävissä oleviin tietoihin ja taitoihin sekä välineisiin ja resursseihin. Tilaisuudella tarkoitetaan avautuvaa mahdollisuutta, sopivaa hetkeä tai pääsyä toteuttamaan hyökkäys. Ollakseen todellinen, uhka tarvitsee kaikki nämä kolme osa-aluetta. Jos yksikin osa-alue puuttuu, uhka ei voi konkretisoitua. (Little & Rogova 2006)

Puhekielessä riskiä käytetään usein synonyyminä uhkan todennäköisyydelle. Riskeihin voi suhtautua eri tavoin, kuten välttämällä, hyväksymällä, rajaamalla, lieventämällä tai oppimalla elämään niiden kanssa (Limnéll et al. 2014). Tätä kaikkea sanotaan riskienhallinnaksi, joka on oma laaja kokonaisuutensa, eikä kuulu tämän työn sisältöön. Kuva 5 vetää yhteen yllä esiteltyjä käsitteitä ja niiden suhteita toisiinsa.




***Kuva 5.*** *Kaaviokuva riskiin liittyvistä käsitteistä (mukaillen lähdettä United States Naval Academy 2021).*

Seuraavaksi esitellään CVE-haavoittuvuuksienhallintajärjestelmä, haavoittuvuuksien luokittelemiseksi kehitetty CWE-järjestelmä sekä haavoittuvuuksien vakavuutta mittaava CVSS-pisteytysjärjestelmä. Kaikki kolme ovat kansainvälisesti tunnettuja ja yleisesti käytettyjä järjestelmiä, ja ne esiintyvät säännöllisesti haavoittuvuustietojen yhteydessä.

### 2.2.1 CVE-hallintajärjestelmä

CVE (*Common Vulnerabilities and Exposures*) on yhdysvaltalaisen Mitren ylläpitämä tietoturvahaavoittuvuuksien ja -heikkouksien hallintaan kehitetty järjestelmä. Järjestelmä on ollut käytössä vuodesta 1999. (Mitre 2020)

Kun uusi haavoittuvuus tallennetaan CVE-järjestelmään, se saa yksilöivän tunnisteen, jolloin siihen on helppo viitata esimerkiksi tiedotteissa, haavoittuvuusskannausten raporteissa ja ohjelmistojen korjauspäivitysten yhteydessä. Lisäksi järjestelmäylläpitäjät ja muut kiinnostuneet voivat etsiä tunnisteen avulla lisätietoja haavoittuvuudesta. Järjestelmä on julkinen ja avoin, eli kuka tahansa voi hakea haavoittuvuuslöydölleen CVE-tunnistetta. (Mitre 2020)

CVE-tunniste muodostuu kolmesta osasta: CVE-alkuosasta, vuosiluvusta sekä juoksevasta numerosta. Esimerkiksi tunnetun Windows-palvelinten Zerologon-haavoittuvuuden yksilöllinen CVE-tunniste on CVE-2020-1472. Tunnisteessa oleva vuosiluku ei välttämättä ole sama kuin haavoittuvuuden julkaisuvuosi, sillä tunniste on voitu varata haavoittuvuudelle jo jonkin aikaa ennen haavoittuvuuden julkaisua.

CVE-järjestelmä ei ota kantaa siihen, minkä tyyppinen tai kuinka vakava haavoittuvuus on kyseessä. Vakavuusasteen mittaamiseen ja esittämiseen on olemassa yleisesti käytössä oleva CVSS-järjestelmä, josta kerrotaan alaluvussa 2.2.3.

### 2.2.2 CWE-luokittelujärjestelmä

CWE (*Common Weakness Enumeration*) on yhteisöllisesti ylläpidetty ohjelmistohaavoittuvuuksien luokittelujärjestelmä. Sen tavoite on kehittää ymmärrystä ja tietoisuutta erilaisista haavoittuvuustyypeistä sekä auttaa tunnistamaan, korjaamaan ja ehkäisemään ohjelmistovirheitä. CVE-järjestelmän tavoin myös CWE-järjestelmää operoi Mitre. (Mitre 2021b)

Mitren verkkosivuilla ylläpidetään CWE-luokitteluun pohjautuvaa listaa 25 vaarallisimmasta ohjelmistohaavoittuvuustyypistä. Keväällä 2021 listan kärjessä on *cross-site scripting*- eli XSS-haavoittuvuustyyppi, joka mahdollistaa hyökkääjän haittakoodin syöttämisen WWW-sivustoille. Listaus perustuu haavoittuvuuksien CVSS-pisteytykseen, joka esitellään seuraavassa luvussa.

Toinen samankaltainen vaarallisimpien haavoittuvuuksien listaus on OWASP:n (*The Open Web Application Security Project*) TOP 10 -lista. Se keskittyy nimensä mukaisesti web-sovellusten haavoittuvuuksiin. Listan kärjessä on injektiohaavoittuvuus. (OWASP 2021).

### 2.2.3 CVSS-pisteytysjärjestelmä

Kaikki haavoittuvuudet eivät ole yhtä merkittäviä tai vakavia. Eri haavoittuvuudet muodostavat erilaisia uhkia ja siksi niiden vakavuusasteetkin vaihtelevat. On myös hyvä huomata, että sama haavoittuvuus voi muodostaa eri ympäristöissä hyvin eritasoisen uhkan, minkä vuoksi haavoittuvuuden vaikutusten ymmärtäminen juuri omassa ympäristössä on tärkeää.

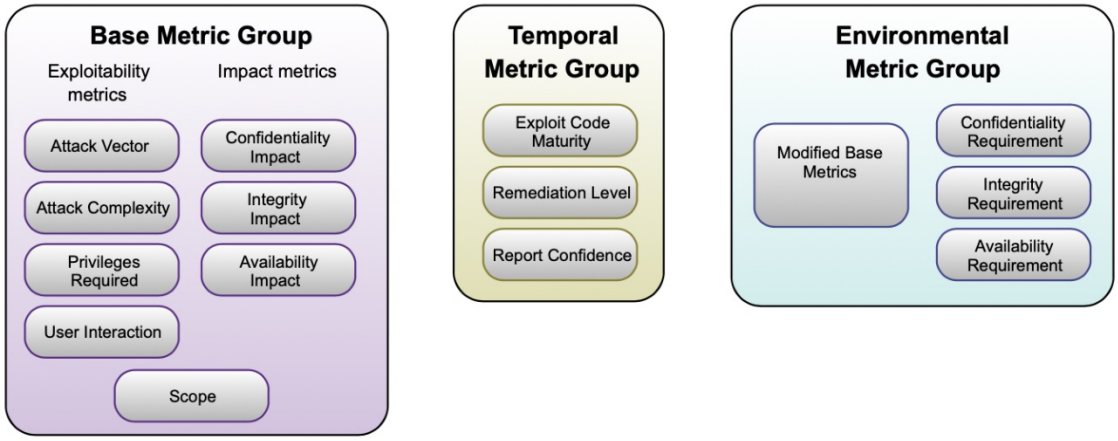
CVSS-pisteytysjärjestelmä (*Common Vulnerability Scoring System*) on tietoteknisten haavoittuvuuksien vakavuuden arviointiin kehitetty järjestelmä. Se on kansainvälisen FIRST-järjestön (*Forum of Incident Response and Security Teams*) luoma avoin luokittelujärjestelmä, jonka versio CVSSv1 julkaistiin vuonna 2005, ja viimeisin versio 3.1 kesällä 2019. (FIRST 2020) Tässä työssä CVSS-järjestelmästä puhuttaessa viitataan jatkossa, ellei toisin mainita, sen uusimpaan versioon 3.1.

CVSS-järjestelmää voidaan pitää merkittävänä, koska se on vakiinnuttanut paikkansa laajasti käytössä olevana valmistajariippumattomana mittaristona. Järjestelmä tuottaa haavoittuvuudelle sen vakavuutta kuvaavan pistearvon yhden desimaalin tarkkuudella

väliltä 0,0–10,0. Yksittäinen CVSS-lukuarvo on helposti tulkittavissa, joten haavoittuvuuksien vakavuuksista saa helposti ja nopeasti karkean tason ensikäsityksen. Yhteinen mittaristo tuo haavoittuvuuksien hallintaan myös yhdenmukaisuutta ja läpinäkyvyyttä. Haavoittuvuuksien pisteyttäminen helpottaa niiden vakavuuksien hahmottamista sekä keskinäistä vertailua, mikä auttaa myös esimerkiksi priorisoimaan korjaustoimenpiteitä organisaation sisällä.

On hyvä ymmärtää myös mihin CVSS-järjestelmä ei ota kantaa. Pisteytykseen eivät vaikuta esimerkiksi haavoittuvan tuotteen yleisyys tai mahdollisesta hyväksikäytöstä koituvat taloudelliset kustannukset.

CVSS-järjestelmässä haavoittuvuuden saama kokonaispistemäärä muodostuu useista eri tekijöistä. Järjestelmän viimeisimmässä versiossa ensimmäiset kahdeksan tekijää kuvaavat haavoittuvuuden ominaisuuksia ja muodostavat niin sanotun perusmittariston. Lisäksi käytetään ajallisia ja ympäristöön sidottuja mittareita. Mittarit on listattu ja ryhmitelty Kuvassa 6, ja ryhmien perusominaisuuksista kerrotaan tarkemmin seuraavissa alaluvuissa. Yksityiskohtaiset määritelmät, laskukaavat ja tulkintaohjeet löytyvät FIRST:n julkaisemasta CVSS-mittariston virallisesta dokumentaatiosta. (FIRST 2020)




***Kuva 6.*** *CVSS-järjestelmässä mittarit jaetaan kolmeen ryhmään (FIRST 2020).*

**Perusmittaristo (Base Metrics)**

Base-ryhmä kuvaa haavoittuvuuden luontaisia ominaisuuksia, jotka pysyvät samoina ajan kulusta ja käyttöympäristöstä riippumatta. Base-pisteet määrittelee tavallisesti haavoittuvasta tuotteesta vastuussa oleva taho, kuten ohjelmistoyhtiö. Base-ryhmään kuuluu kaikkiaan kahdeksan ominaisuutta, jotka on jaoteltu haavoittuvuuden hyödynnettävyyttä (*exploitability*) ja vaikuttavuutta (*impact*) kuvaaviin osatekijöihin.

Ensiksi mainittu tarkoittaa teknisiä menettelytapoja ja keinoja, joilla hyökkääjä voi hyödyntää kyseistä haavoittuvuutta. Jälkimmäinen ryhmä arvioi onnistuneesta hyökkäyksestä tai haavan hyväksikäytöstä kohteelle tai sen komponentille koituvia välittömiä haittavaikutuksia. (FIRST 2020)

Hyödynnettävyyttä kuvaavat osatekijät ja niiden lyhenteet ovat seuraavat:

- Hyökkäysvektori (Attack Vector, AV)
- Hyökkäyksen monimutkaisuus (Attack Complexity, AC)
- Vaaditut käyttöoikeudet (Privileges Required, PR)
- Käyttäjän vuorovaikutus (User Interaction, UI)

Hyödynnettävyyden osatekijät ottavat kantaa muun muassa siihen, missä kontekstissa ja mitä välinettä, eli hyökkäysvektoria käyttäen haavoittuvuuden hyödyntäminen on hyökkääjälle mahdollista. Hyökkäysvektorina voi toimia esimerkiksi julkinen verkko tai sähköpostin liitetiedostona leviävä haittaohjelma. Mittari saa sitä korkeammat pisteet, mitä kauempana, fyysisesti tai loogisesti tarkasteltuna, hyökkääjä saa sijaita haavoittuvasta komponentista ollakseen yhä riittävän lähellä hyväksikäyttääkseen haavoittuvuutta. Logiikka perustuu siihen olettamukseen, että potentiaalisten hyökkääjien määrä kasvaa jos esimerkiksi fyysistä kontaktia kohteeseen ei vaadita. (FIRST 2020)

Myös hyökkäyksen monimutkaisuus on oleellinen tekijä tarkasteltaessa haavoittuvuuden kokonaiskriittisyyttä. Mitä vähäisemmällä osaamisella ja niukemmilla resursseilla hyökkäys on toteutettavissa, sitä suurempi on jälleen potentiaalisten hyökkääjien joukko. Käyttäjän, siis jonkun muun henkilön kuin hyökkääjän, vuorovaikutus huomioidaan pisteytyksessä siten, että mitä vähemmän ulkopuolisten vuorovaikutusta onnistuneen hyökkäyksen toteuttamiseksi tarvitaan, sitä vakavammaksi uhka luokitellaan. Hyökkäyksen onnistumisen näkökulmasta tarpeellista vuorovaikutusta voi olla esimerkiksi vahingollisen verkkosivun avaaminen tai tietyn sovelluksen asentaminen kohdekoneeseen. (FIRST 2020)

Vaikuttavuusmittareilla arvioidaan suoria vaikutuksia, jotka onnistuneesta hyökkäyksestä hyökkäyksen kohteelle todennäköisesti koituisivat. Mittareina toimivat tietoturvallisuuden kolme perusominaisuutta luottamuksellisuus, eheys ja saatavuus. Kutakin turvaominaisuutta arvioidaan yksi kerrallaan. Jos ominaisuus ei vaarannu kyseisestä haavoittuvuudesta, se saa mittarissa arvon *None*. Muita vaihtoehtoja ovat *Low* (matala) ja *High* (korkea).

Viimeisenä, eli kahdeksantena ominaisuutena arvioidaan haavoittuvuuden vaikutuksen laajuutta (Scope, S). Pisteet nousevat, jos haavoittuvan komponentin, kuten ohjelmiston

tai ohjelmistomoduulin, onnistuneella hyväksikäytöllä voi olla vaikutusta haavoittuvan komponentin kanssa eri turva-alueella (*security scope*) toimivaan toiseen komponenttiin. FIRST määrittelee turva-alueen niin, että kaikki komponentit, jotka tarjoavat toimintojaan vain komponentille A, kuuluvat A:n kanssa samaan turva-alueeseen. Esimerkiksi sovellus, joka käyttää erillistä tietokantaa, on tietokannan kanssa samassa turva-alueessa, mikäli tietokantaa ei käytä mikään muu sovellus. (FIRST 2020)

**Ajallinen mittaristo (Temporal Metrics)**

Temporal-ryhmä kuvaa haavoittuvuuden sellaisia ominaisuuksia, joiden pisteytys voi muuttua ajan myötä:

- Hyökkäyskoodin kypsyys (Exploit Code Maturity, E)
- Haavoittuvuuden paikattavuus (Remediation Level, RL)
- Haavoittuvuustiedon luotettavuus (Report Confidence, RC)

Helppokäyttöisen hyväksikäyttömenetelmän julkaisu nostaa pistemäärää, kun taas virallinen korjauspäivitys laskee sitä. Kun uusi haavoittuvuus löydetään, virallisen korjauksen saapumisessa kestää oma aikansa. Usein riskiä on mahdollista lieventää jo ennen varsinaista lopullista korjausta erilaisilla tilapäisillä rajoitustoimenpiteillä. Epäviralliselkin korjausohjeen olemassaolo laskee pisteitä. Välillä tiedot haavoittuvuudesta, varsinkin jos haavoittuvuus on hyvin tuore ja vastikään julkaistu, ovat epämääräisiä tai tarkat tekniset yksityiskohdat puuttuvat. Ylipäänsä kaikki haavoittuvuustietoon liittyvä epävarmuus nostaa pistemäärää. (FIRST 2020)

Ajallinen mittaristo huomioi tunnettujen, käytettävissä olevien korjaus- ja rajoituskeinojen olemassaolon, mutta ei ota kantaa siihen, onko keinoja otettu ympäristössä käyttöön. Vasta seuraavaksi esiteltävä ympäristömittaristo huomioi tehdyt toimenpiteet.

**Ympäristömittaristo (Environmental Metrics)**

Environmental-ryhmän metriikat ovat toimintaympäristö- ja organisaatiokohtaisia. Tällä mittarilla voidaan kustomoida haavoittuvuuden vakavuusarviota juuri omaan kohteeseen sopivaksi ja sen prioriteettien mukaiseksi. Haavoittuvuutta arvioidaan luottamuksen, eheyden ja saatavuuden turvaamisen näkökulmista. Jos esimerkiksi luottamuksellisuuden säilyminen arvioidaan muita kahta arvoa tärkeämmäksi, sille voidaan antaa mittarissa muita suurempi painoarvo. Asteikkona on määrittämätön-matala-keskitaso-korkea. (FIRST 2020)

Työkalu sallii myös Base Metrics -perusmittariston ominaisuuksien kustomoinnin. Tämä antaa mahdollisuuden huomioida ympäristössä haavoittuvuuden hyväksikäytön estämiseksi jo tehdyt rajoitustoimenpiteet. Jos esimerkiksi ohjelmisto asennetaan oletusarvoisesti käyttämään ylläpitäjätason käyttäjätunnusta, mutta organisaatiossa ohjelmiston oikeuksia on rajoitettu, voi vaikuttavuutta ohjelmiston haavoittuvuuden vakavuusarvion kohdalla laskea matalammalle tasolle.

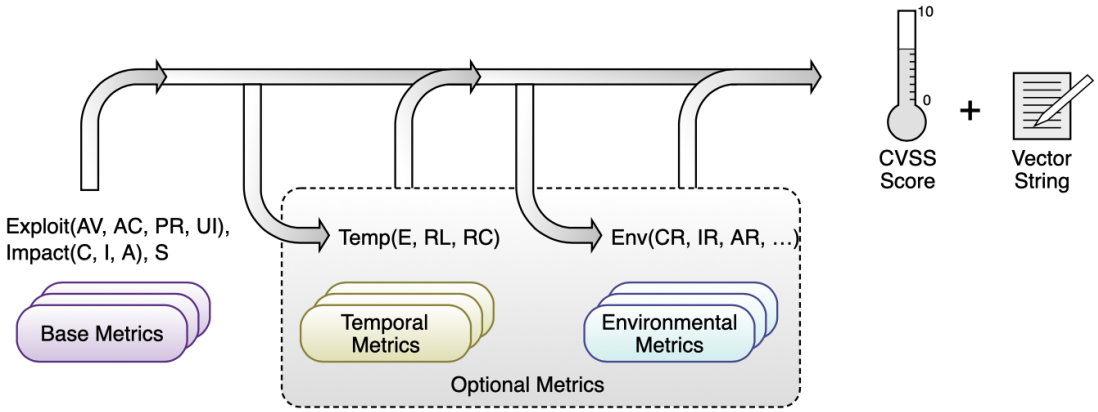
**Pisteyttäminen**

Haavoittuvuuden CVSS-kokonaispistearvo ilmaistaan yhden desimaalin tarkkuudella asteikolla 0,0–10,0. Mitä vakavammasta haavoittuvuudesta on kyse, sitä korkeammaksi pisteet nousevat. Pistearvo muodostuu peruspisteistä sekä valinnaisista pisteistä. Peruspisteet johdetaan haavoittuvuuden Base Metrics -mittariston ominaisuuksien pohjalta, ja ne määrittelee tavallisesti haavoittuvasta tuotteesta vastuussa oleva taho, kuten ohjelmistoyhtiö. Peruspisteet pysyvät yleensä muuttumattomina ajan kuluessa. Valinnaiset pisteet muodostuvat aikaan ja ympäristöön sidotuista arvoista, eivätkä ne ole CVSS-mallissa pakollisia, mutta ne täsmentävät kokonaisarviota haavoittuvuuden vakavuudesta. (FIRST 2020)

Ympäristöstä riippuvat pisteet (Environmental Metrics) määrittelee loppukäyttäjäorganisaatio, ja tämä arvio on tarkoitettu organisaatioiden sisäiseen käyttöön. Ohjelmiston käyttäjä on paras mahdollinen taho arvioimaan, miten haavoittuvuuden hyväksikäyttö vaikuttaisi juuri heidän omassa ympäristössään.

Kaikki arviot tehdään sillä olettamuksella, että hyökkääjä on jo havainnut haavoittuvuuden olemassaolon ja tunnistanut mistä haavoittuvuudesta on kyse. Tästä seuraa se, ettei arvioijan ole tarpeen pohtia, kuinka helposti haavoittuvuus voidaan löytää, eli sillä ei ole pisteytyksen kannalta merkitystä. (FIRST 2020)

Tarkat laskukaavat löytyvät CVSS-mittariston dokumentaatiosta, ja pisteiden laskemiseksi verkossa on olemassa myös CVSS-laskureita (FIRST 2021). Karkealla tasolla laskuprosessi etenee Kuvan 7 mukaisesti. Lopputuloksena (kuvassa oikealla) syntyy paitsi CVSS-pistearvo, myös kirjainmerkkijono (*Vector String*), josta on nähtävissä eri osa-alueiden vaikutus kokonaispistemäärään. Merkkijono on määrämuotoinen, ja se tulee aina esittää CVSS-pistelukeman yhteydessä. (FIRST 2020)



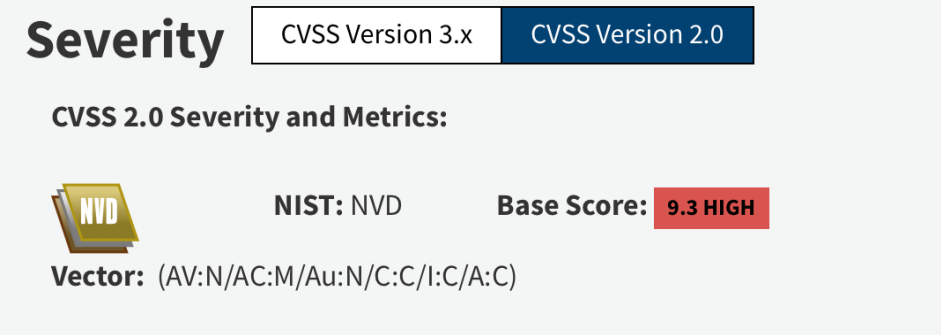

***Kuva 7.*** *Haavoittuvuuden CVSS-pistearvon muodostuminen (FIRST 2020).*

***Taulukko 1.*** *CVSS-pisteytyksen vakavuusluokat (FIRST 2020).*

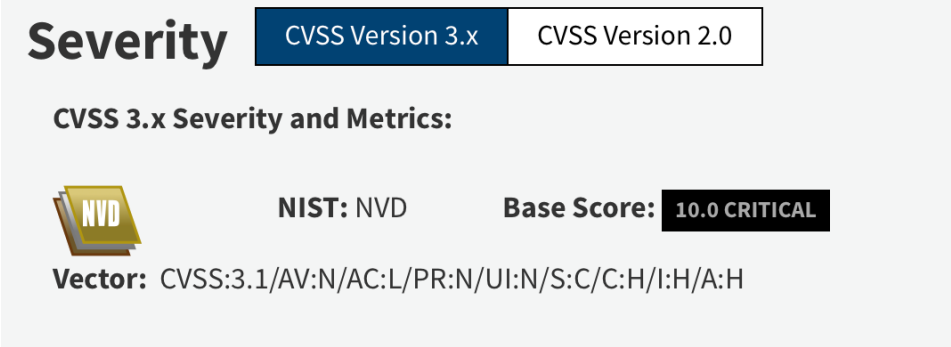
| CVSS-pisteet | Sanallinen vakavuusluokitus |
|---:|---|
| 0,0 | Tyhjä (none) |
| 0,1–3,9 | Matala (low) |
| 4,0–6,9 | Keskitaso (medium) |
| 7,0–8,9 | Korkea (high) |
| 9,0–10,0 | Kriittinen (critical) |

Pisterajat ja niitä vastaavat sanalliset vakavuusluokitukset on esitetty Taulukossa 1. Kuvat 8 ja 9 esittävät saman Zerologon-haavoittuvuuden tiedot vanhemmalla CVSS 2.0 -pisteytyksellä sekä uudemmalla CVSS 3.1 -pisteytyksellä. Kuvista huomataan, että pistemäärät eroavat hieman toisistaan. Molemmista kuvista käy ilmi myös merkkijonomuotoinen esitystapa. Esimerkiksi Kuvan 9 merkkijono AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H tarkoittaa seuraavaa:

- Attack Vector: Network (Hyökkäysvektori: Verkko)
- Attack Complexity: Low (Hyökkäyksen monimutkaisuus: Matala)
- Privileges Required: None (Vaaditut käyttöoikeudet: Ei mitään)
- User Interaction: None (Käyttäjän vuorovaikutus: Ei mitään)
- Scope: Changed (Laajuus: Muuttunut)
- Confidentiality: High (Luottamuksellisuus: Korkea)
- Integrity: High (Eheys: Korkea)
- Availability: High (Saatavuus: Korkea)

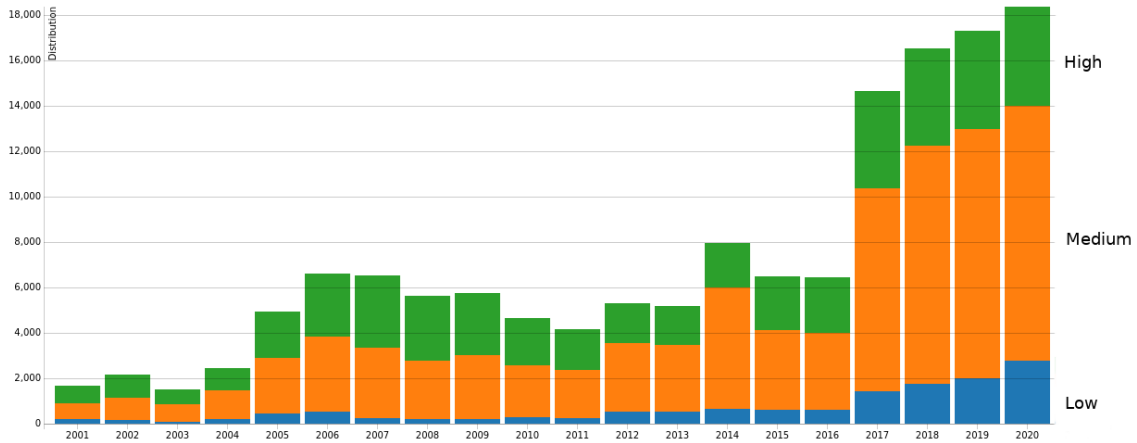



***Kuva 8.*** *Zerologon-haavoittuvuus CVSS 2.0 -pisteytyksellä (NIST 2020a).*




***Kuva 9.*** *Zerologon-haavoittuvuus CVSS 3.1 -pisteytyksellä (NIST 2020a).*

Kuva 10 esittää CVSSv2-pisteytettyjen haavoittuvuuksien pistejakaumaa vuosina 2001–2020. Aivan ensimmäisinä tarkasteluvuosina haavoittuvuuksia pisteytettiin noin 2000 kappaletta vuodessa. Tämän jälkeen vuosina 2005–2016 sekä CVE-tunnisteen saaneita haavoittuvuuslöytöjä että -pisteytyksiä tehtiin melko tasaisesti 4000–8000 kappaletta vuodessa, kunnes vuonna 2017 molempien määrät nousivat merkittävästi. Vuosina 2017–2020 haavoittuvuuksia löydettiin ja pisteytettiin keskimäärin noin 17 000 kappaletta vuosittain. Viime vuosina yli puolet haavoittuvuuksista on määritelty vakavuudeltaan Medium-luokkaan, eli keskitasoisiksi. Kuvaaja perustuu CVSSv2-laskentatapaan, minkä vuoksi siitä puuttuu vasta luokittelun kolmosversioon tuotu Critical-vakavuusluokka. (NIST 2021)




***Kuva 10.*** *Haavoittuvuuksien CVSSv2-pistejakauma vuosina 2001–2020 (NIST 2021).*

CVSS-järjestelmä on vakiinnuttanut paikkansa yleisenä haavoittuvuuksien vakavuuksien mittarina. Yksimielisyyttä sen hyvyydestä ei kuitenkaan ole, vaan se on saanut osakseen myös kritiikkiä. Spring et al. (2021) nostavat tutkimuksessaan esille monta CVSS:n ongelmaa. Tutkijat kritisoivat erityisesti läpinäkyvyyttä, kuten pisteytyksen osatekijöiden painoarvojen perusteluiden puuttumista. Tutkijat huomauttavat myös, että vaikka mittaristo antaa selkeät vakavuusarviot haavoittuvuuksille, niistä tehtävät johtopäätökset ovat yhä hankalia. Miten eri tavalla tulisi suhtautua esimerkiksi High- ja Critical-tason varoituksiin?

# 3. TIETOTURVAHAAVOITTUVUUKSIEN HAVAITSEMINEN HAAVOITTUVUUSSKANNAUSTEN AVULLA

Tässä luvussa selvitetään, mitä IP-verkkoon kohdistetut haavoittuvuusskannaukset ovat, miten niitä suoritetaan ja miten tuloksia voidaan tulkita. Ensiksi tutustutaan IP-verkkojen porttiskannausten perusasioihin, Nmap-skannaustyökaluun ja verkkoon liitettyjen laitteiden tunnistamiseen skannausten avulla. Lisäksi tutustutaan haavoittuvuusskannerin arkkitehtuuriin ja käydään läpi haavoittuvuusskannausprosessi kokonaisuudessaan suunnittelusta skannauksiin ja tulosten raportointiin. Lopuksi tarkastellaan automaatiojärjestelmien erityispiirteitä ja tutustutaan haavoittuvuusskanneriohjelmistoon Nessus Professional.

## 3.1 Tietoliikenneverkon porttiskannaustekniikat

Tämä luku valmistaa lukijaa haavoittuvuusskannauksille kuvailemalla erilaisia porttiskannausmenetelmiä tietoliikenneverkossa. Lisäksi luku auttaa lukijaa ymmärtämään tietoliikenneverkon toimintaa ennen haavoittuvuusskannaukseen tutustumista.

Tässä työssä porttiskannauksella tarkoitetaan RFC 4949:n (Shirey 2007) porttiskannaukselle antaman määritelmän mukaista toimintaa. Sen perusperiaate on etsiä verkosta laitteita sekä niiden TCP- ja UDP-porteissa toimivia palveluita lähettämällä kohdekoneille tietyllä tavalla muotoiltuja paketteja. Vastauksia analysoimalla voidaan selvittää, mitä palveluita kohdekoneissa on käynnissä. Jos portti on auki, kyseisessä portissa toimii jokin palvelu, joka kuuntelee porttiin tulevaa liikennettä. Porttiskannauksen tuloksena saattaa selvitä esimerkiksi, että kohdekoneessa SSH-palvelu (*Secure Shell Protocol*) vastaa portissa 22, ja HTTP-palvelu portissa 80.

Verkkoskannauksella kerättyjä tietoja analysoimalla hyökkääjän on mahdollista suorittaa kohteiden profilointia ja löytää tietojärjestelmistä mahdollisia aukkoja tai heikkoja kohtia. Porttiskannaus on hyvä apuväline myös tietoturvatestaajalle, koska sen avulla voi etsiä omasta verkosta palveluita, joiden ei tulisi olla avoinna ulkoverkkoon tai avoinna lainkaan. Jos porttiskannauksen tavoitteena on pahantahtoinen avointen palveluiden tai haavoittuvuuksien etsiminen tai hyväksikäyttö, se saatetaan tulkita lainvastaiseksi toiminnaksi, kuten tietomurron tai tietoliikenteen häirinnän yritykseksi (Rikoslaki 21.4.1995/578).

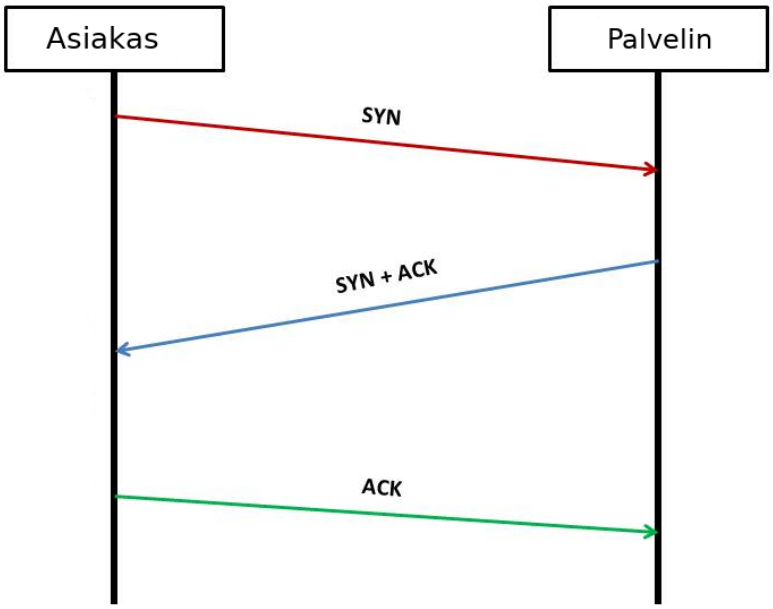
TCP/IP-maailmassa tietokoneissa on 65 536 loogista tietoliikenneporttia, jotka on numeroitu 0–65 535. Palvelusta riippuen portissa toimii joko TCP- (*Transmission Control Protocol*) tai UDP-protokolla (*User Datagram Protocol*). Portit voidaan jakaa kolmeen ryhmään: järjestelmäportteihin (porttinumerot 0–1023), käyttäjäportteihin (1024–49151) ja dynaamisiin portteihin (49152–65535). 1024 ja sitä suurempia porttinumeroita kutsutaan usein yläporteiksi. (Cotton et al. 2011)

Palvelimesta tarjottava palvelu kytkeytyy kuuntelemaan jotain tiettyä porttia. Palveluilla on tyypillisesti omat vakioporttinsa, mutta ylläpitäjä voi määrittää palvelun toimimaan myös jossain muussa portissa. Esimerkiksi HTTP-palvelun (*Hyper Text Transfer Protocol*) oletusportti on TCP-portti 80. (Cotton et al. 2011) Yleisesti tunnetuissa oletusporteissa pysyminen helpottaa palveluiden käyttöä, mutta toisaalta altistaa ne massaskannauksille. Jos verkkosivuston *www.tuni.fi* ylläpitäjä päätyisi siirtämään palvelun toimimaan oletusportin sijasta portissa 8080, täytyisi vierailevan käyttäjän tietää uusi porttinumero ja yhdistää siihen URL:lla *http://www.tuni.fi:8080*. Tämä luonnollisesti heikentäisi merkittävästi julkisten palveluiden käytettävyyttä.

Tämän päivän internetissä skannauksia haavoittuvuuksien löytämiseksi tehdään taukoamatta, joten on järkevää suorittaa myös itse skannauksia omaan verkkoon, johon tuntemattomat rikollisetkin kohdistavat niitä joka tapauksessa. Julkisiin IP-osoitteisiin kohdistuvaa skannailua on mahdotonta estää kokonaan, mutta rajoitusmahdollisuuksia on olemassa. Tyypillisimmät skannaajat omassa verkossa voi selvittää esimerkiksi reunareitittimen flowdataa tai palomuurilokeja analysoimalla. Skannausliikennettä omaan verkkoon voi rajoittaa esimerkiksi estämällä tietyt lähdeosoitteet kokonaan tai rajoittamalla tietyistä lähteistä tulevien yhteyksien määriä tai nopeuksia. Skannauslähteitä voi olla paljon, ja osoitteet saattavat vaihtua tiheästi. Rajoitustoimenpiteillä saattaa olla myös ei-toivottuja vaikutuksia, sillä estetyksi tai rajoitetuksi voi joutua myös hyötyliikennettä.

### 3.1.1 TCP-yhteys ja TCP-skannaukset

TCP on yhteydellinen protokolla, ja normaalisti TCP-yhteys avataan kolmivaiheisella kättelyllä (Kuva 11), joka alkaa SYN-paketin lähetyksellä kohdekoneen tiettyyn porttiin. SYN-paketti tarkoittaa, että TCP-paketin SYN-valitsin on asetettuna. Kohde vastaa SYN+ACK-viestillä, jolloin kättelyn aloittanut taho tietää portin olevan käytettävissä ja päättää kättelyn lähettämällä ACK-kuittausviestin. Tämän jälkeen varsinainen TCP-tiedonsiirto voi alkaa. Jos SYN+ACK-vastausta ei tule, portti ei ole käytettävissä, tai avausyritys suodatettiin. (Weidman 2014)




***Kuva 11.*** *TCP:n kolmivaihekättely, jossa yhteys muodostetaan, ja jonka jälkeen varsinainen tiedonsiirto kahden osapuolen välillä voi alkaa.*

Tavanomaisessa TCP-skannauksessa (myös nimeltään *TCP Connect scan*) skannaaja pyrkii avaamaan yhteyden TCP:n kolmivaiheisen kättelyn (Kuva 14) mukaisesti, kuitenkin aloittamatta varsinaista tiedonsiirtoa. Jos viimeinenkin ACK-paketti päästään lähettämään, tiedetään varmuudella, että yhteys on auki, ja kohdekone kuuntelee kyseistä TCP-porttia. Tämän jälkeen skannaaja sulkee yhteyden välittömästi. Näin suoritettu skannaus on turvallinen kohdekoneenkin kannalla, koska avointen yhteyksien ylläpito ei jää kuluttamaan sen resursseja. (Engebretson 2013)

SYN-skannaus tai puoliavoin TCP-skannaus on skannaustekniikka, jolla etsitään avoimia TCP-portteja ilman aikomustakaan viimeistellä yhteyttä. SYN-skannauksessa kättely aloitetaan aivan kuten TCP:n kolmivaiheisessa kättelyssä normaalisti SYN-yhteydenmuodostuspaketilla. Skannaaja odottaa saako se vastausta, mutta vastauksesta riippumatta se ei itse lähetä ACK-kuittausta, vaan yhteys jätetään tarkoituksellisesti avaamatta. Vastaanottopäässä SYN-skannaukset saattavat jäädä havaitsematta, koska kaikki järjestelmät eivät kirjaa lokeihin yhteyksiä, joita ei ole avattu kokonaan. Tämän vuoksi skannaustekniikkaa kutsutaan myös *Stealth*-skannaukseksi, vaikkakin tämän päivän palomuurit ja tunkeutumisenhavaitsemisjärjestelmät huomaavat ja rekisteröivät SYN-skannaukset. (Engebretson 2013)

Samaa menetelmää käytetään TCP SYN Flood -hyökkäyksessä, jossa kohteena olevaan IP-osoitteeseen lähetetään suuri määrä TCP-protokollan SYN-paketteja, joihin kohde vastaa. Kättely jätetään viimeistelemättä, eli ACK-viestejä ei lähetetä, ja kohdekone jää turhaan odottamaan kuittauksia sekä yhteyden valmistumista.

Kuittauksia odottavat yhteydet varaavat resursseja kohdepalvelimelta, palomuurilta ja muilta verkon aktiivilaitteilta. Flood-hyökkäys eroaa skannauksesta siinä, että pakettimäärä on suuri, eikä hyökkääjä ole kiinnostunut vastauspaketeista, ja tämän vuoksi lähdeosoite voi olla väärennetty.

FIN-, NULL- ja Xmas-skannaukset muistuttavat läheisesti toisiaan. Ne hyödyntävät TCP:n ominaisuutta, jonka mukaan suljetun portin tulisi vastata RST-paketilla, eli RST-bitti asetettuna, mikäli vastaanotetussa paketissa RST ei ole asetettuna. Näin suljetun portin tulisi välittää tieto, että portti ei ole käytettävissä. Käänteisesti, jos vastausta ei tule, portti saattaa olla auki. Nämä kolme skannaustapaa jättävät kuitenkin paljon arvailujen varaan, koska välissä saattaa olla myös palomuuri. (Nmap 2021a)

### 3.1.2 UDP-palvelut ja UDP-skannaukset

TCP-pohjaisten palveluiden lisäksi on olemassa myös UDP-protokollan päällä toimivia palveluita. Internetin keskeisiä UDP:tä käyttäviä protokollia ovat esimerkiksi DNS (*Domain Name System*) ja SNMP (*Simple Network Management Protocol*). UDP-protokolla on yhteydetön protokolla, ja UDP-pohjaisten palveluiden porttien skannauslogiikka eroaa yhteydellisen TCP-protokollan skannauksista. Merkittävin ero protokollien perustoiminnan välillä on siinä, että vastaanotettuja UDP-sanomia ei kuitata vastaanotetuiksi, kun taas jokaisen TCP-paketin perillemeno kuitataan, ja kuittauksen puuttuessa se lähetetään uudelleen.

Skannaajan näkökulmasta UDP-palveluiden skannaaminen on hankalaa, koska protokollan normaalin toimintaperiaatteen mukaisesti vastauspaketit puuttuvat. Vaikka portti on auki, ja palvelu vastaanottaa skannaajan paketin, mitään vastausta ei välttämättä tule. Tilanne on sama, jos paketti jää palomuurin suodattamaksi tai portti on kiinni. Joissain tapauksissa UDP-palvelukin vastaa, mikä on luonnollisesti selvä merkki siitä, että portti on auki. (Engebretson 2013) Auki olevien UDP-porttien havaitsemiseen on silti olemassa keinoja. Nämä perustuvat portissa toimivan palvelun tunnistamiseen, ja aihetta käsitellään seuraavassa luvussa.

Puutteellisesti suojattuja avonaisia UDP-palveluita voi olla mahdollista käyttää välikappaleina peilatuissa palvelunestohyökkäyksissä (*reflection attack*). Tälläinen hyökkäys toteutetaan tyypillisesti lähettämällä avoimiin UDP-palveluihin esimerkiksi bottiverkosta paketteja väärennetyillä lähdeosoitteilla. Lähdeosoitteeksi väärennetään hyökkäyksen kohteeksi päätetyn palvelun osoite, jolloin UDP-palvelun vastauspaketit päätyvät, eli peilautuvat, bottiverkon sijaan uhriksi valikoituneeseen kohteeseen. Kun välikappaleina toimivia laitteita on paljon, liikennemäärä voi kasvaa hyvinkin suureksi.

Tällöin kohdepalvelin ei ehdi käsittelemään kaikkia sille saapuvia pyyntöjä, tai palvelulle varattu kaistanleveys loppuu kesken. (US-CERT 2019)

Hyökkäystä kutsutaan myös vahvistinhyökkäykseksi (*amplification attack*), koska eri palveluiden tuottama vastaus saattaa olla merkittävästi sen vastaanottamaa alkuperäistä kyselyä suurempi. Taulukossa 2 on listattu muutamia UDP-palveluita ja niiden vahvistuskertoimia. Esimerkiksi aikapalveluiden NTP-protokollan monlist-komentoa hyödyntämällä on mahdollista saavuttaa kyselylle lähes 600-kertainen vahvistus. Monlist-kyselyllä NTP-palvelimelta voi kysyä aikapalvelua käyttävät asiakaskoneet.

Memcached-välimuistipalvelun haavoittuvuus CVE-2018-1000115 mahdollistaa palvelun hyväksikäytön palvelunestohyökkäykseen jopa kertoimella 50 000. Avoimeen memcached-palveluun pystyy syöttämään dataa, ja siltä voi kysyä välimuistin sisältöä. Vuonna 2018 avoimeksi jätettyjä memcached-palveluita hyödynnettiin siihen mennessä voimakkaimmassa tunnetussa palvelunestohyökkäyksessä (1,35 Tbit/s). Palvelun päivityksen jälkeen UDP-protokolla ei ole enää oletuksena päällä, vaan palvelu toimii oletusarvoisesti TCP:llä. (Singh & Singh 2018)

Omien UDP-palveluiden käyttöä peilaushyökkäyksiin voi rajoittaa sulkemalla tarpeettomat ja päivittämällä haavoittuvat palvelut. Jos verkossa havaitsee liikennettä, joka ei kuulu mihinkään oikeaan yhteyteen, sen voi suodattaa. Epätavalliset liikennepiikit suuren vahvistuskertoimen UDP-palvelun liikennemäärässä ovat usein merkkejä vahvistinhyökkäyksistä. (US-CERT 2019)

***Taulukko 2.*** *Vahvistinhyökkäysten vahvistinkertoimia eri UDP-palveluilla (US-CERT 2019).*

| **Protokolla** | **Vahvistuskerroin** |
|---|---:|
| BitTorrent | 3,8 |
| SNMPv2 | 6,3 |
| DNS | 28–54 |
| SSDP | 30,8 |
| NTP | 556,9 |
| Memcached | 10 000–50 000 |

### 3.1.3 Nmap ja laitteiden tunnistaminen porttiskannauksella

Porttiskannauksella voidaan selvittää, mitä portteja ja palveluita kohteessa on auki ja käytettävissä. *Nmap* (*Network Mapper*) on perinteinen ja tunnettu porttiskanneriohjelmisto. Se on ilmainen avoimen lähdekoodin sovellus, jonka ensimmäinen versio julkaistiin jo vuonna 1997. Nmapin ominaisuuksia on täydennetty lukuisilla lisäosilla, jotka tunnistavat muun muassa käyttöjärjestelmiä ja ohjelmistojen versioita. (Lyon 2011) Nmap on hyvin laajasti käytetty ja suosittu työkalu tietoliikenne- ja tietoturva-asiantuntijoiden keskuudessa, minkä ansiosta sen käyttöön löytää internetistä runsaasti oppaita ja tukea. Niinpä tässä työssä ei mennä työkalussa yksittäisten komentojen tasolle.

Avointen porttien löytäminen IP-osoitteesta on suoraviivaista. Tehtävä vaikeutuu, kun yritetään selvittää mikä laite skannauksen kohdekone on. Tässä ei päästä täyteen tarkkuuteen ja varmuuteen, koska kohdekoneen ohjelmisto on voitu asettaa vastaamaan juuri tietyllä tavalla esittämään jotain muuta, kuin mikä se oikeasti on. Monesti kuitenkin riittää, että laite tai palvelu tunnistetaan jollain riittävän suurella todennäköisyydellä. Tällaista tiedonkeruuprosessia ja tietojen vertaamista aiemmin kerättyyn vertailudataan kohteen tunnistamiseksi kutsutaan termillä sormenjälkitunnistus tai englanniksi *fingerprinting*. (Weidman 2014)

Sormenjälkitunnistusta voi tehdä monella tapaa, ja Nmap on yksi soveltuva työkalu. Skannattaessa Nmap luo kohteesta niin sanotun sormenjäljen, jota se vertaa omassa tietokannassaan ennestään oleviin sormenjälkiin. Nmapin oma *nmap-os-db*-tietokanta sisältää tunnistetiedot tuhansista laitemalleista, ja se täydentyy koko ajan.

```
# nmap -sV -O 10.10.0.5

Starting Nmap 7.60 ( https://nmap.org ) at 2021-03-23 17:26 EET
Not shown: 997 filtered ports
PORT     STATE SERVICE  VERSION
22/tcp  open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux)
80/tcp  open  http     Apache httpd 2.4.29
443/tcp open  ssl/http Apache httpd 2.4.29 ((Ubuntu))
Aggressive OS guesses: Linux 2.6.32 (92%), Linux 3.10 - 4.8 (92%)
No exact OS matches for host
```

***Kuva 12.*** *Nmap-esimerkki palvelinohjelmistojen tunnistamisesta.*

Kuvassa 12 on esimerkki Nmapin käytöstä palvelimen käyttöjärjestelmän, avointen porttien ja palvelinohjelmistojen tunnistamiseksi. Nmap pyrkii tunnistamaan palvelinohjelmiston lisäksi myös sen versionumeron, mikä on merkityksellistä, koska haavoittuvuudet ovat usein ohjelmistoversiokohtaisia. Tunnistuksessa Nmap käyttää monimutkaista mekanismia, joka hyödyntää muun muassa TCP/IP-pinojen käsittelyssä

olevia käyttöjärjestelmäkohtaisia pieniä eroja (Nmap 2021b). Tulosteesta on karsittu rivejä tilan säästämiseksi.

Yksinkertaisempi tunnistuskeino on lukea palvelimen banner-tietoja ja paluupakettien otsikoita, joissa usein suoraan kerrotaan käyttöjärjestelmä, palvelinohjelmisto ja sen versionumero. Tekniikkaa kutsutaan nimellä *banner grabbing*. Se on sovelluskerroksen skannaustekniikka, jossa kohdekoneeseen avataan ensin TCP-yhteys, ja tehdään pyyntö esimerkiksi HTTP-protokollalla. Vastausviestissä skannaaja saa sitä kiinnostavia tietoja. (Feng et al. 2016) Soveltuvia palveluita ovat HTTP:n lisäksi muun muassa FTP (*File Transfer Protocol*, portti 21) sekä SMTP (*Simple Mail Transfer Protocol*, portti 25), ja soveltuvia työkaluja kyselyiden tekemiseen esimerkiksi Nmap, Netcat sekä Telnet.

Kuvassa 13 on saman WWW-palvelimen antama vastaus Netcat-ohjelmistolla tehtyyn kyselyyn. Jälleen pitää muistaa, että tiedot ovat vain suuntaa antavia. Palvelinylläpitäjän puolestaan olisi viisasta pyrkiä piilottamaan tunnistetietoja niin paljon kuin mahdollista.

```
# nc 10.10.0.5 80

HTTP/1.1 400 Bad Request
Date: Tue, 23 Mar 2021 15:50:21 GMT
Server: Apache/2.4.29 (Ubuntu)
Content-Length: 308
Connection: close
Content-Type: text/html; charset=iso-8859-1
```

***Kuva 13.*** *Banner grabbing -esimerkki Netcat-sovelluksella.*

Nmapin ideaa on viety pidemmälle Zmap-työkalussa. Zmapin kehityksessä on keskitytty ennen kaikkea parempaan skannausnopeuteen, joka on saavutettu muun muassa asynkronisella skannauksella. Asynkroninen skannaus tarkoittaa, että Zmap käyttää lähetykseen ja vastaanottoon eri prosesseja, eli yksi prosessi vain lähettää ja toinen vain vastaanottaa, jolloin lähettäjän ei tarvitse odottaa kohteen vastauksia. Zmap ei myöskään tallenna yhteyksien tilatietoa eikä tee uudelleenlähetyksiä. Yhden gigabitin verkkoyhteydellä Zmapilla voi optimaalisissa olosuhteissa skannata yhden portin koko internetin IPv4-osoiteavaruudesta 45 minuutissa. (Durumeric et al. 2013)

## 3.2 IP-verkon haavoittuvuusskanneri

Porttiskannausta syvällisempää tietoa kohteesta saadaan kerättyä haavoittuvuusskannauksilla. Haavoittuvuusskannerit ovat ohjelmistoja, jotka asennetaan tietokoneelle, josta skannauksia ajetaan joko manuaalisesti tai automatisoidusti. Tässä luvussa esitellään, minkä tyyppisiä haavoittuvuusskannereita on olemassa ja millainen on tyypillisen haavoittuvuusskannerin arkkitehtuuri.

### 3.2.1 Haavoittuvuusskannerityypit

Haavoittuvuusskannereita voidaan ryhmitellä eri tavoin, mutta tyypillinen tapa on jaotella ne skannattavien kohteiden perusteella. Tällöin skannerit voidaan jakaa karkeasti neljään ryhmään (Sultan & Salman 2019), jotka esitellään lyhyesti seuraavaksi:

- verkkopohjaiset skannerit (*network-based*),
- isäntäpohjaiset skannerit (*host-based*),
- verkkosovellusskannerit (*web application*) ja
- langattoman verkon skannerit (*wireless-based vulnerability scanners*).

Tämä työ keskittyy IP-verkkopohjaisiin haavoittuvuusskannereihin. Niitä operoidaan tietoliikenneverkossa, ja yhteydet skannattaviin kohteisiin otetaan verkon yli, eli skannattavat kohteetkin ovat osa verkkoa. Isäntäpohjaiset skannerit sen sijaan ovat ohjelmistoja, jotka asennetaan kohteeseen, eivätkä tarvitse jatkuvaa verkkoyhteyttä toimiakseen. Tällaiset skannerit löytävät paikallisesti haavoja esimerkiksi tiedostojen luku- ja kirjoitusoikeuksista tai Windowsin rekisteristä. Tietokoneeseen asennettu virustentorjuntaohjelmakin on eräänlainen isäntäpohjainen skanneri.

Verkkosovellusskannerit tunnistavat WWW-palveluissa ja muissa verkkosovelluksissa olevia haavoittuvuuksia. Verkkopohjaisiin skannereihin on usein yhdistetty verkkosovellusskannerin ominaisuuksia, eli näiden raja on hälventynyt.

Langattoman verkon skannerit nähdään tyypillisesti omana ryhmänään, koska ilmarajapinta on toiminta-alueena erityislaatuinen. Langaton verkko on hyvä muistutus siitä, ettei verkko ole rajoittunut fyysisellä turvallisuudella, kuten seinillä ja kulkuoikeuksilla rajattuihin alueisiin.

Skannerituotteen tai -palvelun valinnassa on paljon muuttujia. Jos päätyy haavoittuvuusskannerin hankintaan, vaihtoehtoja ovat ainakin palveluntarjoajan tuottama palvelu ja On-Premise-tyylisesti itse asennettava ja ylläpidettävä skanneri. Palveluntarjoajan skanneri voi olla sen omassa konesalissa tai kolmannen osapuolen, esimerkiksi Amazonin, ylläpitämässä ympäristössä.

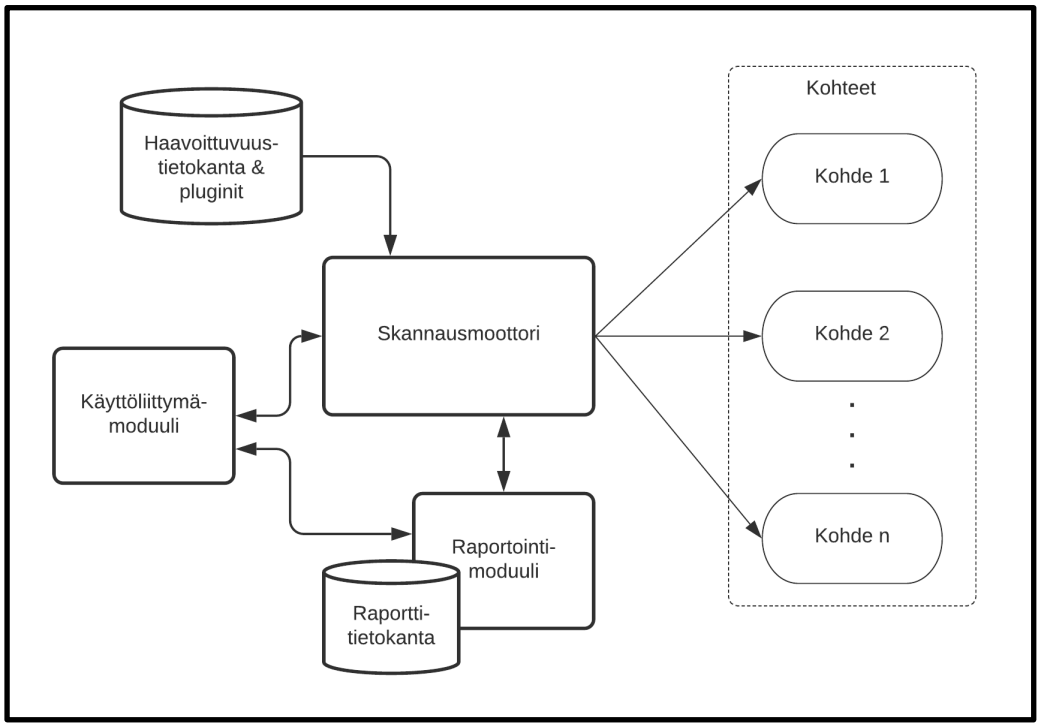
Eri skannerituotteiden vertailu keskenään ilman käytännön skannaustestejä on vaikeaa. Tavallisesti skannerituotteet ovat saatavilla valmistajilta kokeilukäyttöön. Ominaisuuksiin tutustuminen ja käytön opettelu vaativat aina jonkin verran aikaa. Toinen hyvä keino saada vertailtavia testituloksia on osoittaa myyjätaholle omasta verkosta skannattava kohde, ja pyytää siitä skannausraportit. Vertailtavia lukuarvoja on esimerkiksi skannerin tunnistamien haavoittuvuuksien määrä. Ilmoitettu suurempi lukumäärä ei kuitenkaan

välttämättä kata kaikkia niitä haavoittuvuuksia, jotka vähemmän haavoittuvuuksia tunnistava skanneri havaitsee. Eri skannerit voivat täydentää toisiaan etenkin, jos yksi keskittyy verkkotason skannaamiseen, ja toinen sovellustason palveluihin. Jos skannattava verkkoympäristö kattaa tuhansia laitteita, skannausnopeudellakin alkaa olla merkitystä.

Itse skannerituotteen ja sen ominaisuuksien lisäksi tulisi kiinnittää huomiota muun muassa saatavilla olevaan tukeen ja ohjelmistopäivityksiin. Kehityksen on oltava aktiivista, jotta skannerin havaintokyky voisi päivittyä jatkuvasti. Skannerin käyttäjäyhteisöllä on suuri vaikutus tuotteen kehitykseen ja kehityssuuntiin.

### 3.2.2 Haavoittuvuusskannerin arkkitehtuuri

Erilaisista skannerityypeistä huolimatta haavoittuvuusskannerista yleiskäsitteenä voidaan tunnistaa muutamia pääkomponentteja (Kuva 14). Sama malli soveltuu niin IP-verkon laitteiden kuin esimerkiksi WWW-sovellusten skannaamiseen käytettävän skannerin arkkitehtuurin kuvaamiseen.




***Kuva 14.*** *Tyypillinen haavoittuvuusskannerin arkkitehtuuri (mukaillen lähteitä Wang et al. 2020; Atymtayeva et al. 2017).*

Skannerin käyttäjälle eli operaattorille näkyvin osa on käyttöliittymä, jonka kautta skanneria ohjataan, ja josta voi seurata skannerin toimintaa. Käyttöliittymä on tyypillisesti

HTML-pohjainen selainkäyttöliittymä tai komentorivikäyttöliittymä, jota operaattori käyttää omalta tietokoneeltaan. Jotkut skannerit tukevat skanneriyksiköiden hajauttamista eri puolille verkkoa siten, että kaikkiin on näkyvyys saman käyttöliittymän kautta.

Skannausmoottori on haavoittuvuusskannerin keskeisin komponentti, sillä se suorittaa varsinaiset skannaustoimenpiteet. Toimenpiteet perustuvat ennalta ohjelmoituihin komentosarjoihin, joita usein kutsutaan plugineiksi, eli laajennuksiksi. Kun käyttäjä on asettanut skannauskohteet ja valinnut skannausmenetelmän, skannausmoottori valitsee tarpeelliset pluginit ja tulkitsee niitä toteuttaakseen skannaustoimenpiteet. Moottori huolehtii plugineiden suorituksesta sekä verkon ja muistin käytöstä. Se lähettää paketit sille käyttöliittymän kautta syötettyihin IP-osoitteisiin, eli skannauskohteisiin, ja lukee vastauspaketit. (Wang et al. 2020)

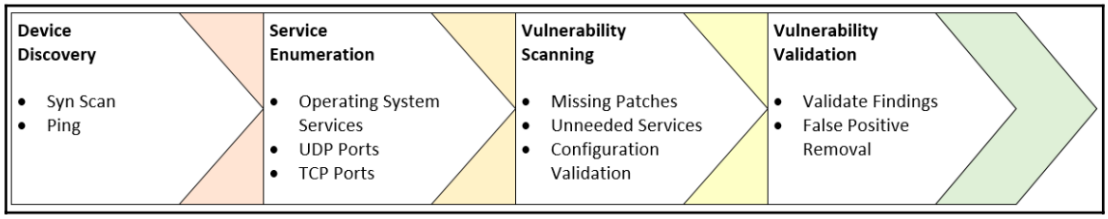
Plugineihin on koodattu tieto siitä, miten haavoittuvuuden olemassaolo tunnistetaan ja testataan. Pluginit sekä tiedot skannerin tuntemista haavoittuvuuksista ovat haavoittuvuustietokannassa, joka päivittyy verkosta internetin yli skannerin kehittäjän ylläpitämästä järjestelmästä. Haavoittuvuuskohtaisia tietoja ovat esimerkiksi haavoittuvuuden nimi, yksilöllinen tunniste sekä ohjeet korjaus- tai rajoitustoimenpiteiksi. Tietolähteinä hyödynnetään tavallisesti CVE- ja CVSS-järjestelmiä.

Raportointimoduulin tehtävä on tuottaa skannausmoottorin havainnoista ihmisen luettavaksi soveltuva raportti tai tuloslistaus. Raporttiin sisällytetään tietoa muun muassa löydettyjen haavoittuvuuksien vakavuuksista. Skannaustulokset tallennetaan tyypillisesti skannerin omaan raporttitietokantaan, josta ne ovat luettavissa käyttöliittymän kautta, tai esimerkiksi lähetettävissä sähköpostitse.

## 3.3 Haavoittuvuusskannausprosessin eteneminen

Tämä luku selvittää, miten järjestelmällisesti toteutettu haavoittuvuusskannausprosessi etenee, ja mitä tulee ottaa huomioon prosessin eri vaiheissa. Haavoittuvuusskannaukset voi suorittaa itse tai työn voi ostaa palveluna ulkopuoliselta toimijalta. Molemmissa tapauksissa prosessin kulku on pääpiirteissään sama.

Kuvassa 15 on esitetty tyypillisen haavoittuvuusskannauksen kulku neljässä vaiheessa. Jo ennen aloittamista työhön liittyy valmisteluja, suunnittelua, viestintää ja verkkoympäristöön tutustumista. Neljä vaihetta on selostettu kuvan jälkeen ensin lyhyesti ja sen jälkeen yksityiskohtaisemmin seuraavissa alaluvuissa skannausprosessin etenemisjärjestyksessä.




***Kuva 15.*** *Haavoittuvuusskannausprosessin eteneminen vaiheittain (Death 2017).*

1. Verkon ja verkkolaitteiden kartoitus (*Discovery*)
2. Palveluiden tunnistaminen (*Enumeration*)
3. Haavoittuvuusskannaus (*Vulnerability Scanning*)
4. Tulosten läpikäynti ja raportointi (*Validation, Reporting*)

Kun haavoittuvuusskannaus on päätetty suorittaa, skannausprosessin ensimmäisessä vaiheessa tutustutaan skannattavaan verkkoon ja kartoitetaan verkosta löytyvät laitteet. Tässä vaiheessa ei vielä analysoida laitteiden palveluita. Kun skannattavat IP-osoitteet ovat tiedossa, selvitetään skannattavien laitteiden palvelut, ja missä porteissa ne toimivat. Portteja tutkittaessa niihin lähetetään paketteja, joihin saatavista vastausviesteistä voidaan päätellä yksityiskohtia palvelimen ohjelmistoista. Kolmannessa vaiheessa suoritetaan varsinaiset haavoittuvuusskannaukset aiemmin löydettyihin palveluihin. Neljännessä ja viimeisessä vaiheessa analysoidaan tuloksia.

### 3.3.1 Skannausten suunnittelu

Haavoittuvuusskannausprosessi käynnistyy verkkoympäristöön tutustumisella. Mitä vieraampi kohdeverkko on skannaajalle, sitä tarkemmin tulisi tutustua verkon rakenteeseen, jotta esimerkiksi skannerikone osataan sijoittaa oikein. Skannerin loogisella sijainnilla on merkittävä vaikutus skannaustuloksiin. Kun halutaan selvittää, miltä omat palvelut näyttävät ulkoverkkoon eli internetiin päin, tulisi skannauksetkin suorittaa samasta suunnasta. Tällöin mahdolliset haavoittuvuudetkin näkyvät skannauksissa samoin, kuin ne näkyvät kaikille internetin käyttäjille.

Mahdollisimman kattavan kokonaiskuvan saamiseksi palvelimet tulisi skannata säännöllisesti sekä ulkoverkosta että organisaation sisäverkosta, myös mahdollisesti käytössä olevat internetissä reitittymättömät yksityiset IP-osoitteet huomioiden. Tämä on tärkeää, koska kaikki hyökkäykset eivät tule ulkopuolelta. Sisäverkkoon saattaa päätyä haittaohjelma tai joku, jolla on pääsy sisäverkkoon, voi käyttäytyä odottamattomasti. Hyvä nykyaikainen ratkaisu onkin sijoittaa skannereita eri puolille ja ohjata niitä erilliseltä asiakaskoneelta, jolla on näkyvyys kaikkiin skannereihin.

Seuraavaksi tulee valita skannattavat kohteet, eli skannerikoneelle määriteltävien kohteiden IP-osoitteet. Suoraviivaisinta on skannata mahdollisuuksien mukaan koko käytössä oleva osoiteavaruus, mutta etenkin suurissa verkoissa voidaan tehdä myös tarkempia rajauksia. Kohteiksi voidaan valita esimerkiksi tietty aliverkko tai valikoida vain tiedostopalvelimet tai pelkästään edellisen skannauksen jälkeen verkkoon lisätyt uudet kohteet. Tämänkaltainen valikointi edellyttää organisaatiolta ymmärrystä oman verkkoympäristönsä sisällöstä ja inventaarion hallintaa (*asset management*). Oleellista on, ettei kohteita tai palveluita jää huomaamatta huonon suunnittelun takia.

Oman inventaarion hyödyntämisen sijaan kartoitus voidaan tehdä myös automatisoidummin käyttämällä passiivisia tai aktiivisia tunnistusmenetelmiä tai näiden yhdistelmää. Passiivinen havainnointi tapahtuu kuuntelemalla verkkoliikennettä ja ottamalla talteen liikennöivien koneiden tietoja, kuten IP-osoitteita ja porttien numeroita. Tällä menetelmällä on samalla mahdollista muodostaa käsitys muun muassa eniten liikennöivistä koneista sekä ylipäänsä liikennelähteistä, -kohteista ja liikenteen luonteesta. Myös käytössä olevan IP-osoiteblokin osoitteiden käyttöaste kyseisessä verkossa voidaan selvittää. (NIST 2008)

Aktiivisessa kartoituksessa käytetään yleensä tarkoitukseen kehitettyä automatisoitua työkalua. Verkkolaitteita selvitetään esimerkiksi ICMP Ping (*Internet Control Message Protocol*) -paketeilla tai TCP SYN -skannauksilla. Monet haavoittuvuusskannerit suorittavat tämän osuuden itsenäisesti ennen varsinaista haavoittuvuusskannausta. Tunnettuja työkaluja verkon kartoitukseen ovat esimerkiksi SolarWinds Network Performance Monitor ja Paessler PRTG Network Monitor.

Passiivisella kartoituksella tiedon kerääminen kestää aktiivista kartoitusta kauemmin. Lisäksi laitteet, jotka eivät passiivisen tarkastelun aikana lähetä tai vastaanota paketteja, jäävät huomaamatta. Aktiivisessakin kartoituksessa jää luonnollisesti huomaamatta sellaiset laitteet, jotka ovat sen aikana pois päältä tai irti verkosta. Verkkolaitteiden kartoitusvaihe on samalla tilaisuus löytää verkkoon kuulumattomat laitteet. Mitä vakiintuneempi verkon laitekanta on, sitä helpompaa on havaita sinne kuulumattomat vieraat laitteet. Jos kartoituksessa löydetään yllättäen uusia, tuntemattomia laitteita, täytyy syyt selvittää ja laitteet paikantaa viipymättä.

Jos skannattavia IP-osoitteita on korkeintaan satoja tai joitain tuhansia, yleensä suoraviivaisin ja hyvä vaihtoehto on skannata koko käytössä oleva osoiteavaruus. Tällöin vältytään myös erillisten skannauslistojen ylläpitämiseltä ja niiden mahdollisilta virheiltä. Uuden palvelimen asentamisen ja käyttöönoton jälkeen se voi helposti jäädä

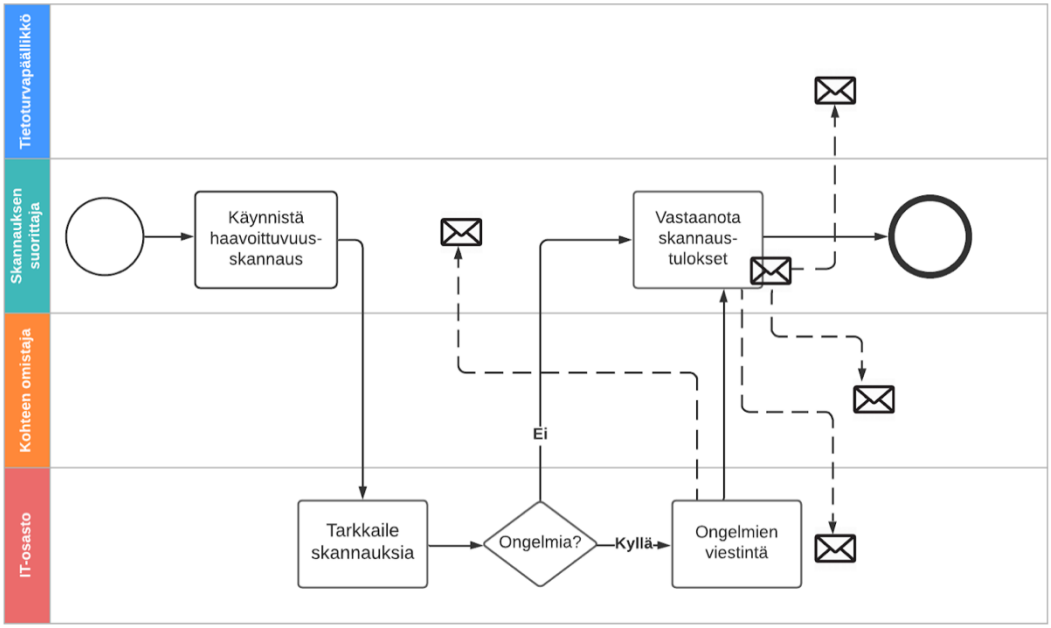
lisäämättä skannauslistoille unohduksen tai muun syyn takia. Kun kaikki koneet tulevat skannatuiksi, saadaan paras kokonaiskuva verkosta ja sen sisältämistä laitteista.

IPv6-osoitteet tuovat mukanaan omat haasteensa. IPv6-osoiteavaruudet ovat valtavasti IPv4-osoitealueita laajempia, eli skannattavia osoitteita on moninkertaisesti. Läpikäytävien osoitteiden määrällä on keskeinen vaikutus skannausten nopeuteen, ja skanneri toimii tehottomasti tutkiessaan osoitteita, joissa ei ole laitteita. Vertailun vuoksi tyypillisessä IPv4-aliverkossa on 254 osoitetta, ja jos verkkoon on kytketty esimerkiksi sata laitetta, suhdeluku on 0,394 laitetta per osoite. Vastaavasti IPv6:n tyypillinen /64-osoiteavaruus sisältää $2^{64}$ osoitetta, mikä tekee samassa sadan laitteen verkossa suhdeluvuksi $5{,}42 \cdot 10^{-18}$. Vaikka jokaista osoitetta skannattaisiin vain yhden sekunnin ajan, mikä ei olisi edes realistista, aikaa kuluisi miljoonia vuosia. IPv6-verkossa skannauskohteet on siis määriteltävä skannerille täsmällisemmin yksittäisinä IP-osoitteina tai DNS-niminä. Monet skannerituotteet eivät edes suostu aloittamaan skannausta IPv6-osoitelohkoihin, eli ne hyväksyvät vain host-osoitteita. Joistain skannereista IPv6-tuki puuttuu kokonaan.

Skannauskohteita valittaessa täytyy huomioida, että IP-osoitteilla ei välttämättä löydä kaikkia verkon palveluita. Esimerkiksi WWW-palvelimessa voi olla saman IP-osoitteen takana useita sivustoja (*virtual host*), jolloin eri sivustot pitää skannata DNS-nimillä tai sivustojen URL:eilla. Paras lähde nimille on palvelimen konfiguraatio. Myös nimipalvelu voi olla hyödyllinen.

Kuvan 16 vuokaavio havainnollistaa haavoittuvuusskannauksen suoritusvaiheen etenemistä, ja eri toimijoiden osallistumista siihen. Skannaus alkaa vasemmalta, ja kaaviota luetaan vasemmalta oikealle. Neljä eri osallistujaa on pinottu päällekkäin, alhaalta ylöspäin IT-osasto, skannauskohteiden eli palveluiden omistajat, skannauksen suorittaja ja ylimpänä tietoturvapäällikkö tai muu skannaustoiminnasta vastaava henkilö.

Haavoittuvuusskannauksen suorittaja käynnistää skannauksen sovittuna ajankohtana. Koska skannauksista aiheutuu aina jonkin verran ylimääräistä verkkoliikennettä ja kuormitusta skannattaville kohteille, skannaukset voi olla hyvä ajoittaa tapahtumaan hiljaiseen aikaan esimerkiksi toimistoaikojen ulkopuolella. Samasta syystä skannausten etenemistä on hyvä seurata ja varautua pysäyttämään skannaus mikäli siitä aiheutuu oleellista häiriötä skannattaviin palveluihin.




***Kuva 16.*** *Vuokaavio haavoittuvuusskannauksen suorittamisvaiheesta (mukaillen lähdettä Palmaers 2013).*

Kuten Kuvan 16 vuokaaviossa viestintää kuvaavista kirjekuoren kuvista huomataan, tiedonvaihto on tärkeää pitkin skannausprosessin eri vaiheita. Jo ennen skannausten aloittamista skannauksen suorittaja tarvitsee listan skannauskohteista. Skannauksen aikana mahdollisista ongelmista on viestittävä skannaajalle, jotta tämä voi tarvittaessa keskeyttää skannaukset kokonaan tai tietyn kohteen osalta. Kun skannaukset ovat valmistuneet, tulokset viestitään eri osapuolille niiden analysointia ja korjaustoimenpiteiden suorittamista varten.

Ennen skannausten aloittamista skannausten ajankohdasta voi harkinnan mukaan tiedottaa etukäteen. Tiedotus auttaa ylläpitäjiä varautumaan mahdollisiin skannauksista aiheutuviin häiriöihin kohdepalveluissa. Toisaalta järjestelmiin jo mahdollisesti päässyt ulkopuolinenkin voi saada saman tiedon ja pystyy varautumaan.

### 3.3.2 Skannaus ja palveluiden tunnistaminen

Kun skannauskohteet ovat tiedossa, päätetään skannattavat portit. 65 536 portin skannaaminen suuressa verkossa kestää kauan, joten ajan säästämiseksi usein skannataan vain yleisimmät portit tai järjestelmäportit 0–1023. Jos yläporteissa on tuotantopalveluita, kannattaa ne sisällyttää mukaan skannaukseen. Taulukkoon 3 on listattu Nmapin tietokannan 20 skannatuinta porttia porttinumerojärjestyksessä. Listaus on saatu komennolla ”*nmap --top-ports 20 localhost*”. Listaus on yksi esimerkki porteista, joihin haavoittuvuusskannauksessa voi keskittyä.

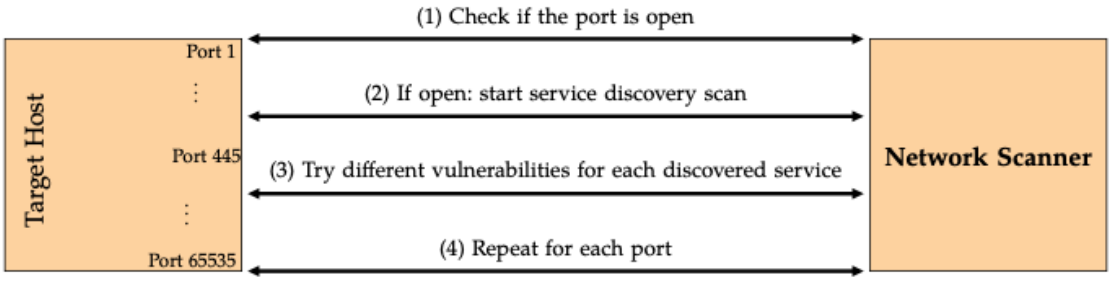
***Taulukko 3.*** *20 skannatuinta porttia Nmapin tietokannan mukaan.*

| Portti | Palvelu | Portti | Palvelu |
|---|---|---|---|
| 21 | FTP | 143 | IMAP |
| 22 | SSH | 443 | HTTPS |
| 23 | Telnet | 445 | SMB |
| 25 | SMTP | 993 | IMAPS |
| 53 | DNS | 995 | POP3S |
| 80 | HTTP | 1723 | PPTP |
| 110 | POP3 | 3306 | MySQL |
| 111 | RPCBIND | 3389 | MS WBT |
| 135 | MSRPC | 5900 | VNC |
| 139 | NetBIOS | 8080 | HTTP |

Skannauksen kestoon vaikuttavat eniten skannattavien osoitteiden ja porttien lukumäärät. Muita keskeisiä tekijöitä ovat skannerin suorituskyky sekä mahdollinen skannerin ja kohteiden välissä oleva palomuuri.

Jos palomuuri pudottaa (*drop*) skannerin lähettämän paketin, aikaa kuluu skannerin odottaessa vastauspakettia. Jos palomuuri lähettää vastauksena hylkäyksen (*reject*), skannerin odotusaika ja skannauksen kokonaiskesto lyhentyvät. Nopeatahtisessa skannauksessa erityisesti tilallisen palomuurin suorituskyky voi nousta pullonkaulaksi. Jos skannattavana on koko 65 535 portin avaruus 30 laitteessa, tästä voi teoriassa muodostua 1,96 miljoonaa sessiota verkon palomuurin tilatauluun. Lisäksi palomuurin on hallittava mahdolliset paluupaketit. Ongelmien ilmetessä skannaukset voi suorittaa pienemmissä paloissa tai hitaammin, palomuurin voi määritellä skannauksen ajaksi pudottamaan yhteyksiä tavallista nopeammin tai skannerissa voi rajoittaa samanaikaisesti auki olevien yhteyksien määrää.

Palveluiden tunnistamisessa hyödynnetään aiemmissa luvuissa esiteltyjä laitteiden ja palveluiden tunnistusmenetelmiä, kuten fingerprint-skannausta. Työvaihetta kutsutaan myös enumeroinniksi (*enumeration*). Monet haavoittuvuusskannerit suorittavat sekä palveluiden tunnistamisen että niiden haavoittuvuusskannauksen yhtenä kokonaisuutena. Kuva 17 esittää haavoittuvuusskannerin työvaiheet haavoittuvuusskannauksen aikana.




***Kuva 17.*** *Haavoittuvuusskannerin työvaiheet haavoittuvuusskannauksen aikana (Sultan & Salman 2019).*

Haavoittuvuusskannauksen aikana skannerikone kohdistaa kohdekoneisiin erilaisia tarkistuksia. Tyypillisiä haavoittuvuusskannauksissa havaittavia järjestelmien ongelmakohtia ovat esimerkiksi vanhentuneet, haavoittuviksi todetut ohjelmistoversiot sekä virheelliset tai puutteelliset konfiguraatiot, kuten liian avoimet tietokannat ja palvelut. Monet skannerit osaavat testata myös kirjautumista ohjelmistojen oletussalasanoilla tai muilla helposti arvattavilla käyttäjätunnuksilla.

Oletussalasanojen testaaminen on keskeinen haavoittuvuustestaamisen osa-alue. Salasanojen vaihtaminen ennen palveluiden avaamista internetiin on tärkeää, koska tällaiset salasanat ovat kenen tahansa löydettävissä esimerkiksi ohjelmistojen julkisista dokumentaatioista tai internetissä jaettavista salasanalistoista. Oletusarvoisilla ja muilla helposti arvattavilla salasanoilla tehtävät kirjautumisyritykset ovat tyypillinen hyökkäysmenetelmä ja keino saada haltuun hyökkäyksen kohde. (Samtani et al. 2016; US-CERT 2016)

Oletussalasanahaavoittuvuudet määritellään CVSS-luokittelussa tyypillisesti kriittisiksi haavoittuvuuksiksi (Samtani et al. 2016). Laadukkaiden salasanojen käyttämisen lisäksi palveluiden sisäänkirjautumisessa tulisi selvittää mahdollisuuksia rajoittaa yhteydenotto-osoitteita sekä hyödyntää monivaiheista tunnistautumista (*Multi-Factor Authentication*, MFA) ja X.509-varmenteita.

Seuraavassa käydään läpi esimerkki skannauksessa löydetyn laitteen tunnistamisesta ja salasanojen testaamisesta. Kuvassa 18 on erään kohdejärjestelmän WWW-palvelimen lokitiedostosta ote, josta näkyy, kuinka skanneri käy läpi erilaisia osoitteita. Skannerilla on tiedossaan luettelo esimerkiksi tyypillisistä ylläpitokäyttöliittymien tiedostonimistä. Skanneri käy näitä läpi löytääkseen toimivan sivun. Esimerkissä toimivaa sivua ei löytynyt, vaan kaikista sivulatausyrityksistä palautui HTTP-virhe 404.

```
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /index.cgi HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /index.pl HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /index.sh HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /nph-mr.cgi HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /query.cgi HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /session_login.cgi HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /show_bug.cgi HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /test HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /test.cgi HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /whois.cgi HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /wp-login.php HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /wwwadmin.cgi HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /wwwboard.cgi HTTP/1.1" 404 - "-"
x.x.50.220 - - [04/Apr/2021:12:02:38 +0300] "GET /xampp/cgi.cgi HTTP/1.1" 404 - "-"
```

***Kuva 18.*** *Ote kohdekoneen lokitiedostosta skannerista etsimässä ylläpitäjän sivua.*

Jos kirjautumissivu löytyy, siihen voidaan heti kokeilla tyypillisiä ylläpitäjän käyttäjätunnuksia, kuten *admin* tai *root*. Vaihtoehtoisesti voidaan yrittää tunnistaa tarkemmin, mistä laitteesta on kyse. Tarkempi tunnistus perustuu esimerkiksi tiettyjen etukäteen tunnettujen elementtien paikantamiseen sivun HTML-lähdekoodista.

Kuvassa 19 on erään Ciscon langattoman tukiaseman kirjautumissivun lähdekoodin alkua. Skanneri voisi hyödyntää esimerkiksi kuvassa ylärivillä tummemmalla merkattua valmistajan nimeä tunnistaakseen, minkä laitteen käyttöliittymästä on kyse, ja sen jälkeen testata juuri siihen toimiviksi tiedettyjä oletussalasanoja. Verkkorikolliset käyttävät samoja menetelmiä etsiessään avoimia laitteita ja saadakseen ne haltuunsa.

```
<!--
#  Copyright (C) 2009, CyberTAN Corporation
#  All Rights Reserved. #
#  THIS SOFTWARE IS OFFERED "AS IS", AND CYBERTAN GRANTS NO WARRANTIES OF ANY
#  KIND, EXPRESS OR IMPLIED, BY STATUTE, COMMUNICATION OR OTHERWISE. CYBERTAN
#  SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS
#  FOR A SPECIFIC PURPOSE OR NONINFRINGEMENT CONCERNING THIS SOFTWARE.
-->
<HTML ><HEAD><TITLE></TITLE>
<meta http-equiv="expires" content="0">
<meta http-equiv="cache-control" content="no-cache">
<meta http-equiv="pragma" content="no-cache">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

***Kuva 19.*** *Ote erään WLAN-tukiaseman kirjautumissivun lähdekoodista.*

### Pilvipalveluiden skannaaminen

Erilaisten pilvipalveluiden yleistyttyä yhä useammat palvelut tuotetaan muualla kuin organisaation omassa verkossa. Palvelimet voidaan esimerkiksi siirtää *Infrastructure as a Service* (IAAS) -mallin mukaisesti virtuaalipalvelimiksi pilvipalveluntarjoajan konesaliin. Usein puhutaan myös niin sanotusta julkipilvestä. Tässäkin tapauksessa omien palveluiden haavoittuvuusskannaus on yleensä sallittua, mutta asia on syytä varmistaa omalta palveluntarjoajalta. Joillain pilvipalveluntarjoajilla palveluvalikoimaan kuuluu haavoittuvuusskannauspalveluita.

Esimerkiksi Amazonin AWS-palvelussa yleisperiaate on, että asiakas saa skannata omia palveluitaan, mutta ei AWS-infrastruktuuriin lukeutuvia palveluita, kuten AWS:n ylläpitämiä nimipalvelimia. Palvelunestohyökkäykset omiinkin palveluihin on AWS:n käyttöehdoissa kielletty. (Amazon Web Services 2021)

### 3.3.3 Tulosten analysointi ja raportointi

Kun skannaukset ovat valmistuneet, skanneri tuottaa havaintojen perusteella raportin. Tässä luvussa esitetään, miten skannaustuloksia tulisi tulkita, ja mitkä ovat seuraavat toimenpiteet.

Skannerisovellukset tuottavat skannauksista yleensä HTML-, CSV- tai PDF-muotoisen raportin. Yksinkertaisimmillaan raportti voi kertoa selväsanaisesti haavoittuvuuksista, kuten puutteellisista konfiguraatioista ja antaa korjausohjeita. Usein tulosten analysoinnissa kuitenkin vaaditaan verkkoympäristön tuntemusta ja teknistä osaamista.

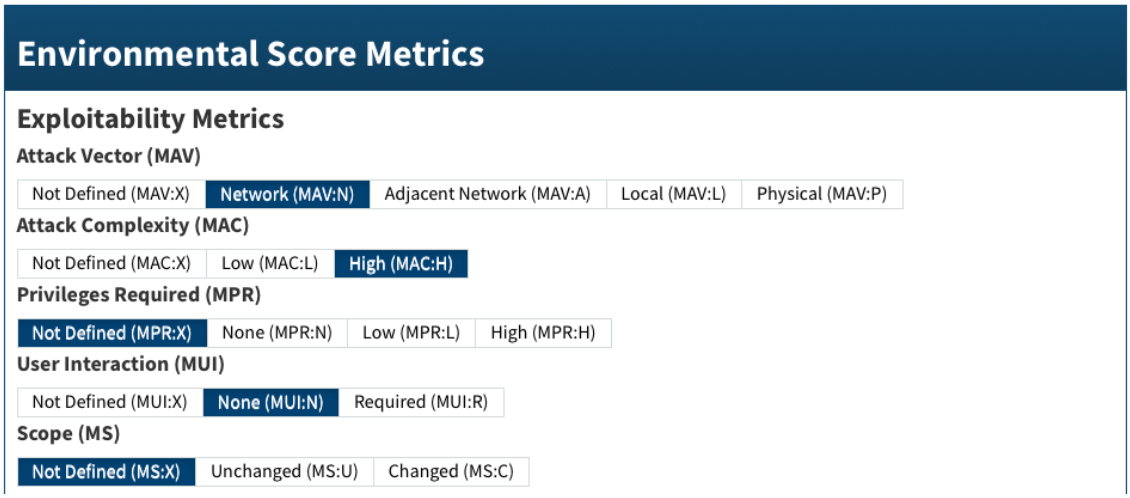
Raporttia lukiessa pitää samalla arvioida, vaikuttaako tulos oikealta. Arviossa voi esimerkiksi tehdä vertailua edelliseen vastaavaan skannaukseen. Arvio voi perustua myös oletuksiin siitä, mitä verkosta pitäisi löytyä. Jos esimerkiksi verkossa pitäisi olla vain Linux-koneita, saattavat Windows-haavoittuvuudet olla vääriä havaintoja. Jos tulokset ovat muuttuneet edellisen skannauksen jälkeen merkittävästi, pitää arvioida muutoksen taustalla olevia syitä. Onko verkkoympäristön topologiassa tai esimerkiksi palomuurauksessa tapahtunut muutoksia? Pitäisikö muutokset huomioida seuraavissa skannauksissa?

On hyvin tavallista, että haavoittuvuusskannausten tuloksena saadaan myös niin sanottuja vääriä positiivisia, eli *false positive* -havaintoja. Nämä ovat havaintoja, jotka skanneri merkkaa haavoittuvuuksiksi, mutta joita järjestelmässä ei kuitenkaan todellisuudessa ole. Vääriin positiivisiin on olemassa useita syitä. Yleisesti voidaan sanoa, että nopeaan tunnistukseen perustuvat skannausmenetelmät eivät pysty tunnistamaan haavoittuvuuksia yhtä hyvin, kuin syvällisempään analysointiin perustuvat menetelmät. Mitä syvemmälle skanneri pystyy kohdistamaan skannauksensa, sitä paremmat mahdollisuudet sillä on tehdä oikeita havaintoja. (Qualys 2020) Esimerkiksi tunnuksellinen SSH-kirjautuminen auttaa skanneria tunnistamaan kohdekoneelta ohjelmistoversiot, kun taas pelkkä banner-tietoihin perustuva palveluiden tunnistus on menetelmänä huomattavasti epävarmempi ja voi johtaa suureen määrään vääriä positiivisia.

Väärien positiivisten juurisyy täytyy selvittää ennen kuin ne jätetään kokonaan huomioimatta. Koneoppimista on tutkittu haavoittuvuusskannausten väärien positiivisten tunnistamiseksi (Gowda et al. 2018). Koneoppimisen hyödyntäminen saattaa säästää

merkittävästi aikaa, sillä käsityönä väärien positiivisten testaaminen ja todentaminen on usein hidasta. Todennetut väärät positiiviset täytyy pyrkiä suodattamaan raporteista seuraavissa skannauksissa. Jos raporteissa väärien positiivisten osuus havaintojen kokonaismäärästä nousee liian suureksi, ne voivat saada liikaa huomiota, jolloin oikeisiin haavoittuvuuksiin puuttuminen saattaa kärsiä.

Raporteissa on tärkeää kiinnittää huomiota oikeisiin asioihin. Monet skannerit osaavat ilmaista haavoittuvuuden yhteydessä siihen liittyvän CVE-tunnisteen sekä haavoittuvuuden vakavuutta kuvaavan CVSS-pistearvon (ks. luvut 2.2.1 ja 2.2.3). CVSS-luokittelussa kriittisiksi arvioidut haavoittuvuudet saavat pistearvon 9,0–10,0, ja korjaustoimenpiteet tulisi aloittaa niistä. Toimintaympäristökohtaiset metriikat saa otettua huomioon CVSS-laskurilla, jollaisen muun muassa NIST tarjoaa verkkosivuillaan. Laskuriin saa lähtöarvoiksi minkä tahansa CVSS-pisteytetyn haavoittuvuuden lähtöpisteet. Tämän jälkeen ympäristökohtaiset muuttujat asettamalla (Kuva 20) pistearvon saa mukautettua omaa ympäristöä vastaavaksi.




***Kuva 20.*** *Toimintaympäristökohtaisten metriikoiden syöttäminen NIST:n CVSS-laskuriin.*

Pisteytyksen ansiosta kriittiset haavoittuvuudet on mahdollista erottaa vähemmän merkittävistä haavoittuvuuksista, jolloin korjaustoimenpiteet voidaan asettaa kiireellisyysjärjestykseen. Priorisointia kannattaa tehdä myös palveluiden perusteella esimerkiksi niin, että internetiin avoinna olevista tai muuten organisaation toiminnan kannalta keskeisistä tuotantojärjestelmistä huolehditaan ensin, ja kehitysvaiheessa tai sisäisessä käytössä olevista järjestelmistä vasta myöhemmin. Yksilölliset CVE-tunnisteet auttavat löytämään verkosta lisätietoa haavoittuvuuksista.

Haavoittuvuusskannauksia voidaan täydentää penetraatiotestauksella eli pentestingillä. Pentesting on tunkeutumistestausta, usein erilaisia manuaalisia toimenpiteitä, joilla

testataan miten havaitun aukon hyväksikäyttö käytännössä onnistuu. Pentesting auttaa myös tunnistamaan ja priorisoimaan turvallisuusriskejä, kun käsitys haavojen hyväksikäyttömenetelmistä paranee.

Kun tarvittavat toimenpiteet haavoittuvuuden hyväksikäytön estämiseksi on tehty, täytyy korjauksen toimivuus vielä varmistaa uusintaskannauksella. Jotkut haavoittuvuusskannerit tallentavat historiatietoa tehdyistä skannauksista omaan tietokantaansa tai skanneriin liitettyyn tiketöintijärjestelmään. Nämä auttavat seurannassa, ja helpottavat puuttumista pitkään korjaamattomina olleisiin haavoittuvuuksiin. Haavoittuvuuden hyväksikäytön riski kasvaa, jos haavoittuvuuden korjaus kestää tai korjausta siirretään myöhemmäksi. Jos korjaus ei ole heti mahdollista, tulisi selvittää vaihtoehtoisia keinoja rajoittaa hyväksikäytön mahdollisuuksia.

## 3.4 Automaatiojärjestelmien haavoittuvuudet

SCADA- (*Supervisory Control and Data Acquistion*) ja IoT-laitteet (*Internet of Things*) muodostavat oman kokonaisuutensa verkon haavoittuvuuksien tarkastelussa. SCADA-laitteet eli automaatiojärjestelmien etähallittavat laitteet sekä muut IoT-laitteet ovat viime vuosina siirtyneet laajasti TCP/IP-verkkoihin. Kun automaatiojärjestelmät ja niiden keskeiset protokollat aiemmin olivat internetistä fyysisesti eristettyjä, niiden tietoturvan tason merkitys oli vähäisempi. Tänä päivänä yhä useammasta laitteesta on etähallinnan toteuttamiseksi yhteyksiä muihin verkkoihin ja järjestelmiin, joten tietoturvariskit ovat kasvaneet.

Tutkimusten (Samtani et al. 2016; Yadav & Paul 2019) mukaan SCADA-laitteiden tietoturvan taso on heikko. Uudetkin automaatiolaitteet perustuvat usein tekniikkaan, jota ei ole suunniteltu nykyisenkaltaisiin verkkoympäristöihin. Tästä seuraa, että laitteiden ominaisuudet ja resurssit eivät ole riittäviä nykypäivän tarpeisiin. Käyttöjärjestelmät ovat vanhoja, eikä niihin välttämättä ole saatavilla päivityksiä, ja muistia sekä suoritintehoa on vähän. Päivitettävyyden ongelmat yhdistettynä laitteiden usein pitkään elinkaareen avaavat entisestään tilaa tietoturvaongelmille.

SCADA-laitteiden suojaaminen on tärkeää muun muassa siksi, koska niillä on konkreettinen yhteys fyysiseen maailmaan. Laitteiden takana on esimerkiksi teollisuusautomaatioon tai kiinteistöjen lämmitykseen, ilmanvaihtoon, valaistukseen ja kulunvalvontaan liittyviä sensoreita ja aktuaattoreita. Laitteissa on usein selainpohjainen hallinta portissa TCP/80, ja hyökkääjät kohdistavat niihin hyökkäyksiä massoittain automatisoidusti. Tyypillisiä ongelmia ovat vaihtamatta jääneet salasanat sekä syötteiden tarkistuksen puutteista seuraavat alttiudet injektiohyökkäyksille ja cross-site

scripting -tunkeutumisille. Monissa laitteissa on käytössä salaamattoman tiedonsiirron telnet-protokolla. (Samtani et al. 2016; Yadav & Paul 2019)

Suojaamattoman laitteen muodostama uhka vaihtelee sen käyttöympäristön mukaan. Yksittäinen laite aiheuttaa uhkan paitsi kyseiselle järjestelmälle tai kiinteistölle, myös kolmansille osapuolille, mikäli laitetta voi käyttää osana palvelunestohyökkäystä. Teollisuusympäristössä laite muodostaa toiminnan keskeytymisen uhkan ja on siten riski yrityksen liiketoiminnalle. Tietyissä ympäristöissä, kuten esimerkiksi voimalaitoksissa, vaikutusalue voi olla vielä merkittävästi laajempi.

SCADA-laitteita voi haavoittuvuusskannata muiden tietojärjestelmien tavoin. Skannaukset on järkevintä suorittaa suunnitelmallisesti huoltokatkojen yhteydessä. Näin varmistetaan, etteivät skannaukset aja herkkiä järjestelmiä alas hallitsemattomasti.

SCADA-laitteen suojaamisessa keskeistä on tapa, jolla se on kytketty internetiin. Laitteen ei tulisi näkyä sellaisenaan julkiseen verkkoon, vaan yhteydenottotapoja on rajattava esimerkiksi niin, että internetin ja etähallittavan laitteen välissä on palomuuri tai VPN-reititin. Yhteydenotot tulee sallia vain tunnetuista ylläpito-osoitteista, oletussalasanat pitää vaihtaa laadukkaisiin, tarpeettomat palvelut pitää poistaa käytöstä ja laitteen ohjelmistoversion ajanmukaisuus pitää varmistaa. Lisäksi tulee selvittää mahdollisuuksia poistaa käytöstä turvattomat protokollat, kuten telnet.

SCADA- ja IoT-laitteiden määrän nopean kasvun vuoksi niiden hallinta verkkoympäristössä on vaikeutunut. Hallinnassa voi auttaa laitekannan automatisoitu tunnistaminen, missä voidaan hyödyntää koneoppimista. Tutkijoiden (Feng et al. 2018) esittelemä koneoppiva malli oppi tunnistamaan IoT-laitteen tyypin, valmistajan ja mallinimen noin 96 prosentin varmuudella. Tunnistamisessa hyödynnettiin fingerprinting- ja banner grabbing -menetelmiä sekä laitearvosteluja ja valmistajien tuotesivuja internetissä.

Kun uusi laite havaitaan ja tunnistetaan ajoissa, on verkon ylläpitäjillä paremmat mahdollisuudet varmistua laitteen tietoturvan tasosta ja huomioida sen kytkentätapa verkossa. On keskeistä ymmärtää, mikä laite voi olla avoimessa ympäristössä ja mikä ei. Nopea reagointi auttaa ennaltaehkäisemään tietoturvauhkien toteutumista.

Koneoppimisella on lukemattomia sovelluskohteita. Samaa koneoppivaa tunnistusmenetelmää voi hyödyntää myös esimerkiksi murrettujen laitteiden tunnistamisessa. Tutkimuksessa (Feng et al. 2018) verkkoon asennettiin niin sanottuja hunajapurkkeja (*honeypot*), jotka ovat tarkoituksellisesti huonosti suojattuja tietokoneita. Hunajapurkki toimii syöttinä hyökkäyksille, ja kun se joutuu hyökkäyksen kohteeksi, se kerää tietoja hyökkääjästä ja sen käyttämistä menetelmistä. Oman laitteen ei normaalisti pitäisi hyökätä hunajapurkkiin. Jos oman IoT-laitteen havaitaan yrittävän hyökkäystä, voidaan

hyvin suurella todennäköisyydellä sanoa, että laite on murrettu ja se pitää irrottaa verkosta sekä puhdistaa ennen käyttöön palauttamista. Myös murron syy pitää selvittää, ja tehdä tarvittavat toimenpiteet murron estämiseksi jatkossa.

## 3.5 Nessus Professional -haavoittuvuusskanneri

Markkinoilla on useita haavoittuvuusskannereita. Tässä luvussa tarkastellaan yhdysvaltalaisen Tenablen Nessus Professional 8 -skanneria. Lisäksi selvitetään, miten edellisissä luvuissa esitetyt haavoittuvuusskannausten yleisperiaatteet on toteutettu Nessus-skannerissa.

Nessus on nykyisin yksi käytetyimmistä haavoittuvuusskanneriohjelmistoista. Se on IP-verkossa toimiva skanneri, jossa on myös WWW-sovellusten testaamiseen soveltuvia ominaisuuksia. Nessus on alun perin syntynyt avoimen lähdekoodin projektina, kunnes vuonna 2005 kehittäjät siirsivät sen kaupallisen lisenssin alle. Viimeisistä avoimen lähdekoodin versioista syntyi muutamia uusia kehityshaaroja, joista pisimmälle on edennyt Greenbonen kehittämä OpenVAS-skanneri (*Open Vulnerability Assessment Scanner*) (OpenVAS 2021).

Tenablen ilmoituksen mukaan Nessus tunnistaa yli 60 000 erilaista CVE-tunnisteen saanutta haavoittuvuutta. Tenable tarjoaa Nessuksesta kahta eri versiota: Nessus Essentials on muun muassa kotikäyttöön soveltuva maksuton versio, jolla voi skannata 16 IP-osoitetta. Nessus Professional on ammattilaisille suunnattu versio, joka on paremmin kustomoitavissa, eikä skannausmääriä ole rajoitettu. (Tenable 2021a)

Nessus päätyi tarkemman tarkastelun kohteeksi, koska se oli entuudestaan tuttu, ja siitä on ladattavissa myös maksuton versio. Osa muista Tenablen tuotteista on saatavissa kokeiluun lyhyiksi testijaksoiksi. Nessus on On-Premise-tuote, eli se asennetaan itse sopivaan tietokoneeseen, johon tekstissä jatkossa viitataan skannerikoneena. Nessuksesta on saatavilla versiot Linuxille, Macille ja Windowsille. Nessus tukee vain yhtä skannerikonetta, eli skannereiden nykyaikainen hajauttaminen eri puolille verkkoa, ja niiden ohjaaminen saman käyttöliittymän kautta, ei ole mahdollista. Tenablen vaihtoehto tähän on Tenable.sc-skanneriohjelmisto, joka ennen nimenvaihtoa tunnettiin nimellä Nessus Security Center.

Nessus noudattaa luvussa 3.2.2 esiteltyä tyypillistä haavoittuvuusskannerin arkkitehtuuria. Nessuksen *nessusd*-palvelinohjelmisto suorittaa skannaukset, ja ylläpitäjä operoi skanneria omalta tietokoneeltaan Nessuksen selainkäyttöliittymän välityksellä. Käyttöliittymän kautta voi käynnistää uusia skannauksia, seurata käynnissä olevien skannausten etenemistä ja selata raportteja suoritetuista skannauksista. Luvussa myöhemmin on

kuvakaappauksia Nessuksen selainkäyttöliittymästä. Kuvista on rajattu tai peitetty tunnistetietoja, kuten oikeita käyttäjätunnuksia ja IP-osoitteita. Nessus tukee vain yhtä käyttäjätunnusta, minkä vuoksi se soveltuu vain sellaiseen ympäristöön, jossa käyttäjiä on yksi, tai jossa käytetään yhteiskäyttöistä käyttäjätunnusta.

Skannattaviin kohdekoneisiin ei tarvitse asentaa asiakasohjelmistoja tai tehdä muitakaan valmistelutoimenpiteitä ennen skannauksia. Tosin skannauksissa päästään syvemmälle kohteeseen, mikäli skannerilla on tiedossaan toimiva käyttäjätunnus (tunnukselliset skannaukset).

Nessuksen monet toiminnallisuudet perustuvat niin sanottuihin plugineihin, eli lisäosiin tai laajennuksiin. Kukin skannaustyyppi on oma erillinen plugininsa, ja aina kun Nessukseen tuodaan tuki uuden haavoittuvuuden tunnistamiselle, se otetaan käyttöön lataamalla ja asentamalla kyseinen plugin tai pluginien sarja. Plugineita on kirjoitushetkellä 157 tuhatta kappaletta, ja ne tulee päivittää säännöllisesti, jotta ohjelmisto tunnistaa uudetkin haavoittuvuudet. Tenablen oman ilmoituksen mukaan se julkaisee noin sata pluginia viikoittain. Pluginit on kirjoitettu C-kieltä muistuttavalla NASL-kielellä (*Nessus Attack Scripting Language*), ja niitä on mahdollista kirjoittaa myös itse. Suuri osa valmiista plugineista on lähdekoodiltaan avointa. (Tenable 2021a)

### 3.5.1 Nessuksen skannausominaisuudet

Tässä luvussa tutustutaan Nessuksen skannausominaisuuksiin. Nessuksessa skannaus etenee käyttöliittymän tasolla nelivaiheisesti:

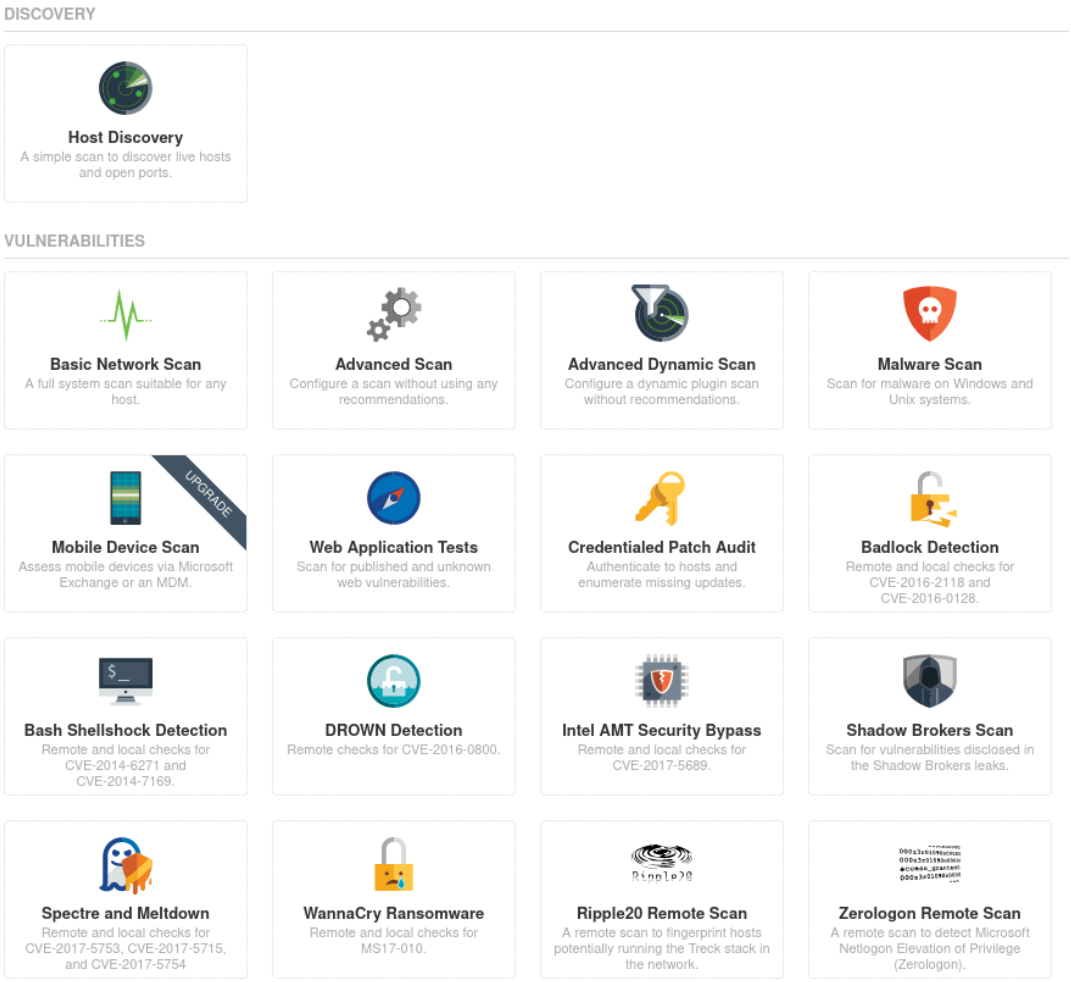
1. Määritellään skannaussäännöt (*Policy*)
2. Luodaan uusi skannaus ja asetetaan kohteet (*New Scan*)
3. Käynnistetään skannaus (*Launch Scan*)
4. Analysoidaan tulokset (*View Results*)

Skannaussäännöissä määritellään, mitä toimintoja skannauksen aikana suoritetaan, ja mitä tietoja raportteihin sisällytetään. Nessuksessa kustomointimahdollisuudet ovat varsin kattavat. Säännöissä määritellään muun muassa skannattavat portit, käytettävät porttiskannausmenetelmät ja suorituskykyyn sekä skannaussyvyyteen ja -nopeuteen liittyviä asetuksia. Säännöillä voi rajata skannaukset vain tiettyihin haavoittuvuuksiin esimerkiksi CVE- tai CVSS-luokitusten perusteella. Skannaus voi olla kertaluontoinen tai sen voi ajastaa tapahtumaan esimerkiksi viikottain tiettynä ajankohtana.

Säännöstön voi tallentaa myöhempää käyttöä varten, jolloin samaa säännöstöä voi käyttää helposti eri skannauskohteille. Nessuksen valmiit skannaussäännöt on jaettu

kolmeen ryhmään. Valintanäkymä on esitetty Kuvassa 21, ja päätasojaottelu on seuraava:

- Kohteiden kartoitus (*Discovery*)
- Haavoittuvuudet *(Vulnerabilities*)
- Vaatimustenmukaisuus (*Compliance*)




***Kuva 21.*** *Yleiskuva Nessuksen skannaustyypin valintanäkymästä. Kuvassa Discovery- ja Vulnerabilites-tyyppisiä skannauksia.*

Discovery-tyyppinen skannaus on tarkoitettu verkon kartoittamiseen ja verkossa olevien laitteiden sekä niissä olevien avointen porttien löytämiseen. Tuloksena saadaan esimerkiksi IP-osoitteita, DNS-nimiä, porttinumeroita ja käyttöjärjestelmiä.

Nessuksen ydintä ovat Vulnerabilities-tyyppiset skannaukset, joilla etsitään haavoittuvuuksia. Usein tunnetut ja paljon julkisuutta saavat haavoittuvuudet lisätään Nessukseen erillisinä, juuri tätä kyseistä haavoittuvuutta skannaavina esivalmisteltuina policyina. Tällaisia haavoittuvuuksia ovat esimerkiksi Spectre, WannaCry ja Zerologon, jotka näkyvät myös Kuvassa 20. Useimmat haavoittuvuudet lisätään uusina plugineina

geneerisempiin Vulnerabilites-ryhmän skannauksiin. Nessus tunnistaa järjestelmistä esimerkiksi haavoittuvia ohjelmistoversioita, virheellisiä ohjelmistokonfiguraatioita sekä oletusarvoisia ja muita helposti arvattavia salasanoja.

Compliance-skannaukset testaavat kohdelaitteiden vaatimustenmukaisuutta jotain tiettyä tietoturvastandardia vasten. Nessuksessa on esimerkiksi PCI DSS -skannaus (*Payment Card Industry Data Security Standard*), joka perustuu kansainväliseen korttimaksamisen teknisten vaatimusten minimitason määrittelevään standardiin. PCI DSS -skannauksiin kuuluu ennalta määritelty joukko tarkistuksia, kuten varmistus, ettei tietokantoja pääse lukemaan ulkoverkosta.

Skannauksissa voi käyttää sekä IPv4- että IPv6-protokollaa. Nessuksessa on natiivi IPv6-tuki, mikä tarkoittaa, että sillä voi skannata IPv6-verkkoja ilman osoitemuunnosta. Skannausten lisäasetuksista on mahdollista valita käyttöön vain niin sanotut *Safe Checks* -skannaukset. Tällöin Nessus kytkee pois sellaiset skannaukset, eli pluginit, jotka saattaisivat aiheuttaa häiriöitä skannauskohteen toiminnalle. Tällaisia skannauksia ovat esimerkiksi testit, joiden suorittaminen saattaisi sotkea tietokannan tai aiheuttaa kohteessa muistivuodon ja kaataa järjestelmän. Joidenkin pluginien ajonaikaisista toimenpiteistä saa esiin yksityiskohtaisempaa tietoa *Audit Trail* -näkymässä. (Tenable 2021b)

**Tunnukseton ja tunnuksellinen skannaus**

Tunnukseton skannaus (*Non-credentialed scan*) perustuu banner-metadataan tai muuhun skannauskohteen ulkopuolelta luettavaan tietoon. Skannauksen suorittaminen on suoraviivaista laajassakin verkossa, koska se ei edellytä etukäteisjärjestelyjä.

Tunnuksellisessa skannauksessa (*Credentialed scan*) skannerille annetaan paikallinen käyttäjätunnus ja pääsy kohdejärjestelmään, minkä ansiosta sillä on mahdollisuus tehdä merkittävästi tunnuksetonta skannausta laadukkaampia havaintoja. Skanneri pääsee esimerkiksi lukemaan asennettujen ohjelmistojen versionumeroita ja Windows-koneissa rekisteriarvoja, minkä ansiosta havainnot ovat täsmällisiä.

Tunnuksetonkin skannaus riittää havaitsemaan verkkoon auki olevat palvelut, minkä jälkeen tarpeettomat palvelut voidaan sulkea tai pääsyä rajoittaa. Tunnuksellisessa skannauksessa yhteys voidaan avata esimerkiksi Linux-koneisiin SSH-protokollalla, ja kirjautuminen voi tapahtua salasanalla tai SSH-avaimella. Jotta skanneri yrittää kirjautumista vain haluttuihin kohteisiin, eikä tarjoa salasanoja kaikille kohteille, voi kirjautumisessa käyttää SSH-avaimia, ja tallentaa julkiset avaimet skannerin *known_hosts*-tiedostoon.

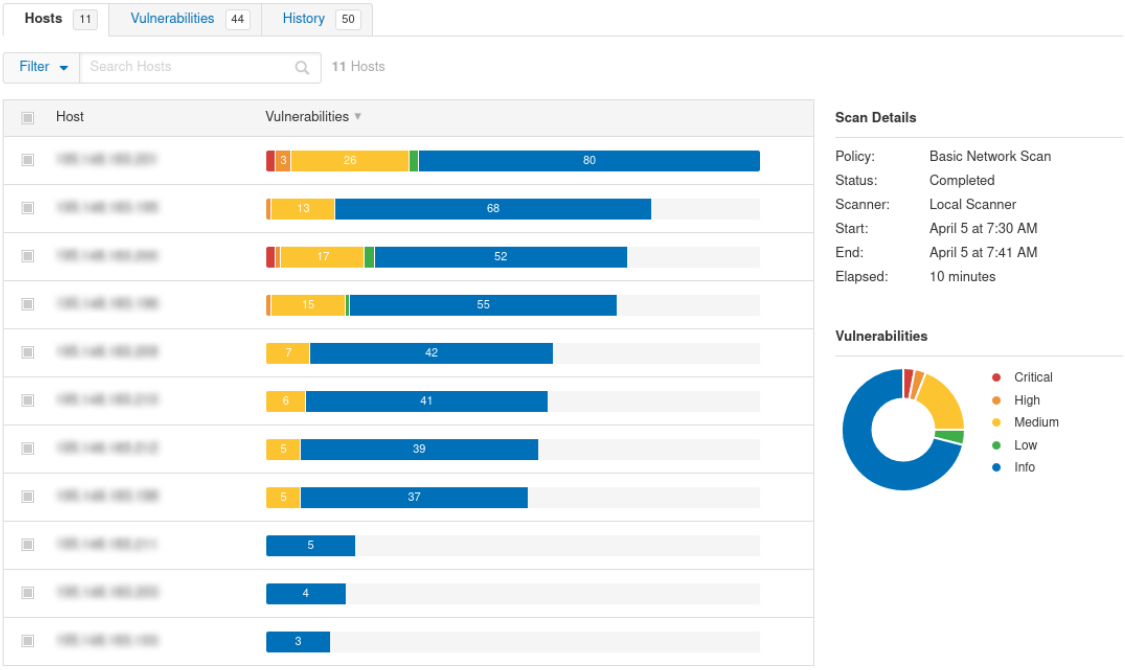
Tunnuksellisessa skannauksessa täytyy huomioida siihen liittyvät riskit. Skannauksia varten kohdekoneisiin on hyvä luoda erillinen, vain skannauskäyttöön tarkoitettu käyttäjätunnus, jolla saa kirjautua tunnetusta skannerin IP-osoitteesta. Tunnuksellista skannausta voi syventää antamalla skannerille oikeudet nousta root-käyttäjäksi kohdekoneessa. Tässä tapauksessa kirjautumiseen käytettävän salasanan ja SSH-avaimen turvallinen käyttö ja säilyttäminen on korostetun tärkeää. Turvallisuuden parantamiseksi kirjautumistiedot voidaan syöttää vain skannausten ajaksi ja poistaa palvelimelta, kun niitä ei tarvita. Avaintenhallinnassa ja muussa automatisoinnissa voi hyödyntää esimerkiksi Ansible-työkalua. Ansible soveltuu keskitetyssä ylläpidossa olevien konfiguraatiotietojen jakeluun SSH- ja Windows Remote Management -yhteyksien yli.

Jos skannerikone on julkisessa verkossa, on sen käytön ja tietokannassa olevien haavoittuvuustietojen suojaamiseksi yhteyksiä koneeseen rajoitettava. Oikeudettoman käytön estämiseksi pääsy on sallittava vain skannauksia suorittavan käyttäjän ja ylläpitäjän IP-osoitteista. Skannaustuloksia ei tulisi säilyttää skannerikoneella tarpeettomasti. Skannerikoneen paikallisessa palomuurissa on hyvä sallia skannaukset vain omiin IP-verkkoihin. Tämä toimii lisäsuojana skannerin skannaussääntöjen rinnalla, jotta ei tulisi edes vahingossa skannanneeksi muita kuin omia järjestelmiä.

### 3.5.2 Nessuksen skannausraportit

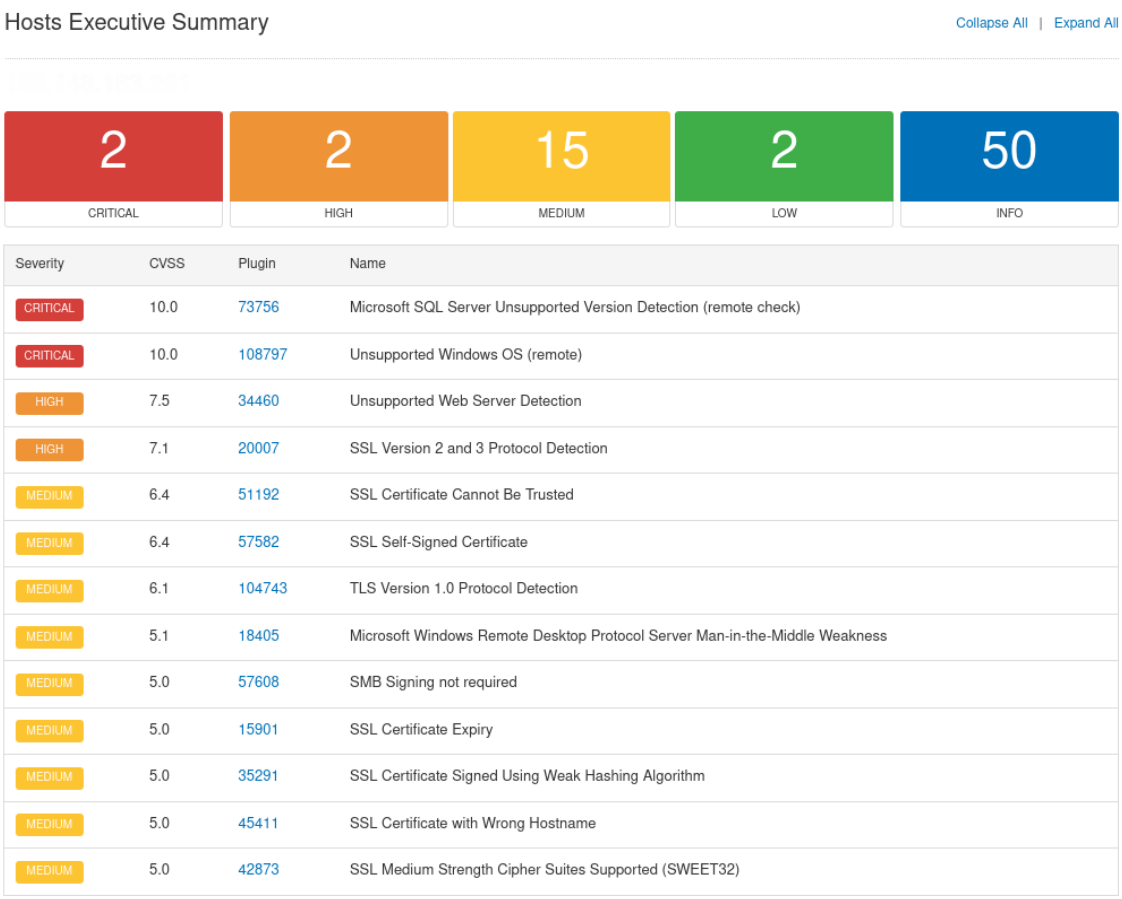
Kun skannaus on päättynyt, tuloksia voi selata Nessuksen selainkäyttöliittymän kautta. Tulokset voi myös ladata esimerkiksi HTML- tai CVS-muotoisina raporttitiedostoina jatkokäsittelyä varten tai lähettää sähköpostitse. (Tenable 2021b) Kuvassa 22 on Nessuksen selainnäkymän raportti valmistuneesta skannauksesta, jossa on ollut 11 kohdetta. Oletusarvoisesti Nessus listaa Hosts-välilehdellä ylimmäiseksi kohteen, jossa on sen mukaan eniten haavoittuvuuksia. Vulnerabilities-välilehdeltä näkee skannauksessa havaitut haavoittuvuudet, ja History-välilehdeltä voi selata saman skannauksen aiempien suorituskertojen tuloksia.

Nessuksessa haavoittuvuuksien luokittelu perustuu kansainväliseen CVSS-järjestelmään, jossa haavoittuvuudet järjestetään niiden vakavuuden perusteella neljään eri tasoon matalasta kriittiseen. Eri tasojen havainnollistamiseksi Nessus käyttää värikoodausta, jossa esimerkiksi kriittiset haavoittuvuudet saavat punaisen värin. Kuvassa 21 näkyvät siniset Info-tason havainnot eivät ole haavoittuvuuksia, vaan skannerin tekemiä informatiivisia havaintoja esimerkiksi käyttöjärjestelmästä ja ohjelmistoversioista.




***Kuva 22.*** *Nessuksen raportoima yhteenveto skannauksen tuloksista.*

Eri tarkoituksiin kannattaa käyttää erilaisia raportteja. *Executive Symmary* -yhteenveto soveltuu koosteeksi, josta saa hyvän yleiskäsityksen. Järjestelmäylläpitäjille tulee toimittaa konekohtaiset raportit. Kuvassa 23 on ote yhdestä koneesta laaditusta haavoittuvuusraportista, jossa kustakin haavoittuvuudesta on listattu vakavuus sanallisessa ja CVSS-pistemuodossa, haavoittuvuuden löytänyt plugin sekä haavoittuvuuden selkokielinen nimi tai kuvaus. Pluginin tunnistenumero toimii linkkinä tenable.comiin kyseisen haavoittuvuuden tarkempaan kuvaukseen. Verkkosivulla voi olla myös korjausehdotuksia.



Hosts Executive Summary

Collapse All | Expand All

| 2 | 2 | 15 | 2 | 50 |
|---|---|---|---|---|
| CRITICAL | HIGH | MEDIUM | LOW | INFO |

| Severity | CVSS | Plugin | Name |
|---|---|---|---|
| CRITICAL | 10.0 | 73756 | Microsoft SQL Server Unsupported Version Detection (remote check) |
| CRITICAL | 10.0 | 108797 | Unsupported Windows OS (remote) |
| HIGH | 7.5 | 34460 | Unsupported Web Server Detection |
| HIGH | 7.1 | 20007 | SSL Version 2 and 3 Protocol Detection |
| MEDIUM | 6.4 | 51192 | SSL Certificate Cannot Be Trusted |
| MEDIUM | 6.4 | 57582 | SSL Self-Signed Certificate |
| MEDIUM | 6.1 | 104743 | TLS Version 1.0 Protocol Detection |
| MEDIUM | 5.1 | 18405 | Microsoft Windows Remote Desktop Protocol Server Man-in-the-Middle Weakness |
| MEDIUM | 5.0 | 57608 | SMB Signing not required |
| MEDIUM | 5.0 | 15901 | SSL Certificate Expiry |
| MEDIUM | 5.0 | 35291 | SSL Certificate Signed Using Weak Hashing Algorithm |
| MEDIUM | 5.0 | 45411 | SSL Certificate with Wrong Hostname |
| MEDIUM | 5.0 | 42873 | SSL Medium Strength Cipher Suites Supported (SWEET32) |

***Kuva 23.*** *Yksittäisen skannauskohteen raportti löydetyistä haavoittuvuuksista.*

Haavoittuvuuksien vakavuuden mittaamiseksi Tenable julkaisi vuonna 2020 *Vulnerability Priority Rating* (VPR) -järjestelmän. Tenable toi järjestelmän ensin Tenable.sc-ohjelmistoon ja huhtikuussa 2021 myös Nessukseen. VPR tuotiin sovelluksiin CVSS-pisteytyksen rinnalle. Tenablen mukaan se kehitti VPR-pisteytyksen, koska CVSS-järjestelmässä liian suuri osuus kaikista haavoittuvuuksista arvioidaan korkeimmalle Critical-tasolle. Tenablen mukaan kriittisiksi arvioitujen haavoittuvuuksien suhteellisen korkea määrä vaikeuttaa korjaustoimenpiteiden priorisointia, eikä kuvasta kaikissa tapauksissa riittävän hyvin niiden todellista kiireellisyyttä. (Tai 2020)

CVSSv3- ja VPR-luokituksissa luokitusten nimet ja pisterajat ovat samat. Toteutuneita pisteytyksiä vertailemalla huomataan, että CVSSv3-luokituksessa korkeimmalle Critical-tasolle kuuluu yli 10 %, ja VPR-luokituksessa vain noin 1 % haavoittuvuuksista. VPR-pisteytyksen laskemisessa Tenable hyödyntää koneoppimista ja pyrkii esimerkiksi huomioimaan haavoittuvuuden hyväksikäyttöesimerkkien esiintymisen sosiaalisessa mediassa ja pimeässä verkossa (*dark web*). (Tai 2020) Pimeällä verkolla tarkoitetaan internetin rakenteissa toimivia verkkosivuja, joihin ei pääse tavanomaisilla verkkoselaimilla ja hakukoneilla. Tuoreet havainnot haavoittuvuuden hyväksikäytöstä ja sen

ympärillä pyörivästä keskusteluista nostavat vakavuusarviota, koska haavoittuvuus nähdään juuri sillä hetkellä ajankohtaisena.

VPR-järjestelmän dynaamisuudesta johtuen haavoittuvuuksien pistearvot voivat muuttua usein. Järjestelmä arvioi, missä elinkaarensa vaiheessa haavoittuvuus kulloinkin on. Järjestelmä pyrkii antamaan korkeimman vakavuusarvion silloin, kun haavoittuvuuden hyväksikäyttö on aktiivisimmillaan.

# 4. HAAVOITTUVUUSSKANNAUSTEN HYÖTYJEN ARVIOINTI JA KEHITYSKOHTEET

Tässä luvussa arvioidaan haavoittuvuusskannausten merkitystä ja hyödyllisyyttä organisaation tietoturvallisuuden kehittämisessä. Lisäksi etsitään kehityskohteita sekä pohditaan tietoturvauhkia, joita esitetyillä skannausmenetelmillä ei havaita, mutta jotka tulisi löytää joillain muilla tavoin.

## 4.1 Haavoittuvuusskannaukset tietoturvallisuuden hallintajärjestelmässä

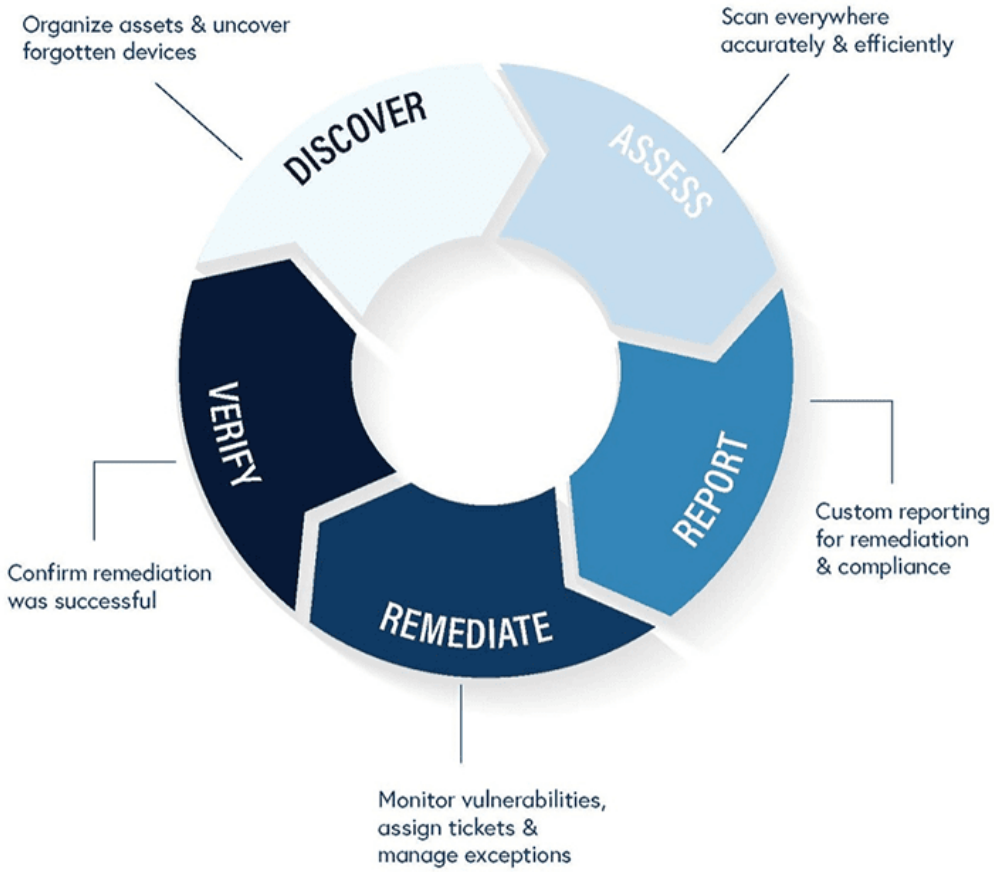
Haavoittuvuusskannausten tärkein tavoite on haavoittuvuuksien löytäminen omasta ympäristöstä ennen kuin rikolliset löytävät ne ja hyödyntävät niitä. Weidmanin (2014) sekä Whitakerin & Newmanin (2005) mukaan kohteen testaaminen hyökkääjän näkökulmasta on paras keino puolustautua.

Suurin hyöty haavoittuvuusskannauksista saadaan, kun skannaukset viedään osaksi koordinoitua haavoittuvuuksien hallinnan prosessia, joka on osa organisaation tietoturvallisuuden hallintajärjestelmää (*Information Security Management System*, ISMS). Tietoturvallisuuden hallintajärjestelmä tähtää organisaation tietoturvallisuuteen liittyvien riskien hallintaan. Hallintajärjestelmämalleja ovat esimerkiksi ISO/IEC 27000 -standardiperhe sekä kotimainen VAHTI-ohjeistus. VAHTI tarkoittaa Valtionhallinnon tieto- ja kyberturvallisuuden ohjausryhmää. VAHTI-ohjeistus on tarkoitettu valtionhallinnon organisaatioiden käyttöön, mutta sitä voi soveltaa myös muissa organisaatioissa.

Organisaation koosta ja toimialasta riippuen tietoturvallisuuden hallintajärjestelmän tärkeimpiä osa-alueita ovat esimerkiksi riskienhallinta, tietoturvapolitiikka sekä jatkuvuus- ja toipumissuunnitelmat. Tietoturvatoimenpiteet perustuvat näihin tekijöihin. Riskienhallinta sisältää riskien tunnistamisen ja arvioinnin. VAHTI-ohjeistus korostaa erityisesti tietoriskien hallinnan tärkeyttä. Tietoturvapolitiikalla tarkoitetaan organisaation sisäistä tietoturvaan ja tietosuojaan liittyvää määräys- ja ohjekokoelmaa. Jatkuvuus- ja toipumissuunnitelmilla turvataan toimintavarmuutta ja jatkuvuutta. Suunnitelmat voidaan laatia esimerkiksi toimintayksikkökohtaisesti. (VAHTI 2011)

Haavoittuvuusskannaukset voidaan nähdä riskien tunnistamisen työkaluna ja jatkuvuussuunnittelua tukevana tekijänä. Haavoittuvuusskannauksilla tunnistetaan omista teknisistä järjestelmistä niiden heikkoja kohtia, jotka korjaamalla vähennetään riskiä joutua esimerkiksi tietomurtojen kohteeksi. Haavoittuvuuksien hallintaa voidaan kuvata

jatkuvana prosessina, kuten Kuvassa 24. Ensimmäisessä vaiheessa kartoitetaan (*Discover*) sen hetkinen laitekannan ja verkon tilanne. Seuraavaksi suoritetaan skannaukset (*Assess*), raportoidaan tulokset (*Report*), tehdään korjaavat toimenpiteet (*Remediate*) ja viimeisessä vaiheessa varmistetaan (*Verify*) esimerkiksi uusintaskannauksella, että korjaustoimenpiteet toimivat. Kuvassa kierros alkaa uudestaan, mikä kuvaa prosessin jatkuvuutta ja säännöllisyyttä sekä toimintaympäristön jatkuvaa muutosta VAHTI-ohjeistuksen esittelemän tietoturvallisuuden vuosikellon tapaan.




***Kuva 24.*** *Haavoittuvuuksien hallinta on jatkuva prosessi (Ascend Technologies 2019).*

Haavoittuvuusskanneri voidaan integroida osaksi muita tietoturvallisuuden hallintaa edistäviä tai tukevia järjestelmiä, kuten SIEM-järjestelmiä (*Security Information and Event Management*) tai tiketöintijärjestelmiä. Haavoittuvuusskannerin hankinnassa tulee ottaa huomioon palvelun toteutustapa. Palveluntarjoajan ylläpitämän skannerin valitessaan välttyy skannerin ja palvelinalustan ylläpidolta. Jos oman verkon haavoittuvuustiedot ovat muualla kuin organisaation omassa järjestelmässä, on niiden käytöstä ja säilytyksestä sovittava. Sovittavia asioita ovat ainakin tietojen säilytyspaikka, säilytyksen kesto ja kenellä on oikeudet tietoon.

Kaupallisen haavoittuvuusskannerin suorat kustannukset koostuvat tavallisesti hankintahinnasta tai kuukausittaisesta lisenssi- tai käyttömaksusta. Välillisissä kuluissa on huomioitava mahdolliset oman palvelinalustan kustannukset sekä työstä koituvat henkilöstökulut. Kokonaiskustannuksia mietittäessä maksuton avoimen lähdekoodin skanneri voi osoittautua ylläpidettävyydeltään ja muulta työmäärältään raskaammaksi, kuin hankintahinnaltaan kalliimpi kaupallinen tuote.

Haavoittuvuusskannauksia ei välttämättä tarvitse suorittaa itse. Haavoittuvuusskannaukset kuuluvat monen tietoturvapalveluita tarjoavan yrityksen palveluvalikoimaan. Jos IT ja tietoturva eivät ole organisaation ydinosaamista, säännöllisen haavoittuvuusskannauksen tai koko haavoittuvuuksien hallinnan prosessin ostaminen ulkopuoliselta voi olla hyvä ratkaisu. Ulkopuolisesta erityisosaamisesta voi olla muutakin hyötyä. Tietoturvallisuuteen ja haavoittuvuusskannauksiin erikoistunut taho pystyy työssään hyödyntämään ajankohtaista tietoa sellaisistakin uhkista, joista tilaaja ei välttämättä ole tietoinen.

Ulkoistuksessa keskeistä ovat sopimukset. Oikeus- ja vastuukysymysten vuoksi kumppanin kanssa on täsmällisesti sovittava, mitä skannataan ja kenen valtuutuksella. Erityisesti kolmannen osapuolen pilvipalveluissa sijaitsevien palveluiden skannaamisessa on selvitettävä lupa-asiat tarkasti, jotta ei syyllisty käyttöehtojen vastaiseen toimintaan. On myös keskeistä selvittää, minkälainen kyvykkyys kumppanilla on skannausten suorittamiseen, missä maassa kumppani toimii ja miten tietosuoja-asiat hoidetaan. Osaamisen lisäksi tarvitaan teknisiä resursseja. Joka tapauksessa on hyvä muistaa, ettei tietoturvan toteutumisen vastuuta voi ulkoistaa.

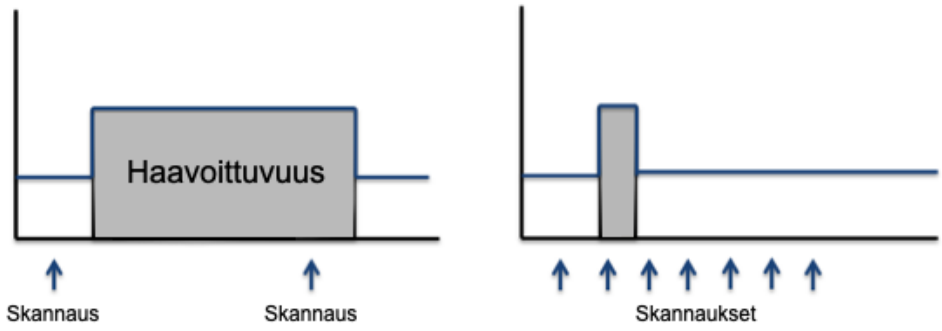
## 4.2 Haavoittuvuusskannausten hyödyt ja haasteet

Tässä luvussa tarkastellaan haavoittuvuusskannausten hyötyjä, haasteita ja kehityskohteita. Haavoittuvuusskannauksilla voidaan löytää omista järjestelmistä haavoittuvuuksia, joista ei muuten oltaisi tietoisia. Haavoittuvuuksien korjaamisella estetään niiden hyväksikäyttö. Järjestelmissä olevia heikkouksia ja aukkoja hyödyntämällä rikollisten on mahdollista muun muassa suorittaa tietomurtoja, aiheuttaa palvelukatkoja ja levittää haittaohjelmia. Proaktiivisella toiminnalla organisaatio välttää siihen kohdistuvia kyberhyökkäyksiä, joista muuten aiheutuisi kustannuksia, mainehaittoja ja katkoja liiketoiminnalle. Haavoittuvuusskannaukset lisäävät organisaation uhkatietoisuutta (*threat intelligence*, ks. luku 4.4.1) ja parantavat tilannekuvaa.

Automatisoidun haavoittuvuusskannauksen etuna on myös kustannustehokkuus. Kun skannauksia ajetaan automatisoidusti taustalla, ne eivät vaadi henkilötyötä. Työaikaa

käytetään skannausten valmisteluun ja tulosten analysointiin, mutta itse skannaustapahtuma pystytään pitkälle automatisoimaan. On kuitenkin selvää, että massoittain tehdyt automaattiskannaukset eivät voi olla kohdekohtaisesti räätälöityjä, joten kaikkien tai edes merkittävimpien haavoittuvuuksien löytämisestä ei voi olla varmuutta.

Haavoittuvuusskannausten suorittamisessa skannausten tiheys ja säännöllisyys ovat tärkeitä tekijöitä. Tätä havainnollistaa Kuva 25, jossa harmaan alueen leveys kuvaa haavoittuvuuden aikaikkunaa jossain tietyssä järjestelmässä. Haavoittuvuudet havaitaan sitä nopeammin, mitä tiheämmin skannauksia suoritetaan. Kunhan myös korjaukset tehdään viipymättä havainnon jälkeen, aikaikkuna kapenee, altistumisaika lyhenee, ja mahdollisuus haavoittuvuuden hyväksikäytölle pienenee.




***Kuva 25.*** *Havainnollistus haavoittuvuusskannausten tiheyden merkityksestä haavoittuvuuden altistumisajalle.*

Haavoittuvuusskannauksia tehdessä on tärkeää muistaa, että pelkkä skannaus ei vielä auta paljoakaan. Skannauksista saatavat raportit täytyy käydä läpi, havainnoista on viestittävä palveluista vastaaville ja korjaustoimenpiteet on vietävä käytäntöön. Työhön sisältyy skannausraporteissa väistämättä silloin tällöin olevien false positive -havaintojen selvittäminen ja suodattaminen. Korjausten tekeminen pitää jalkauttaa järjestelmien ylläpitotasolle, ja korjausten toteutumista on seurattava. Palveluiden omistajia on vastuutettava huolehtimaan omien palveluidensa tietoturvan tasosta.

Panostukset turvallisuuteen ja varautuminen häiriötilanteisiin lisäävät kustannuksia, mutta ovat toisaalta sijoitus liiketoiminnan jatkuvuuteen. Proaktiivinen kyberturvallisuuteen panostaminen voi pienentää kuluja muun muassa vähentyneinä palvelukatkoina.

Kun omasta järjestelmästä on löytynyt haavoittuvuus, se on luonnollisesti pyrittävä paikkaamaan viipymättä. Ensimmäinen vaihtoehto on selvittää, tarjoaako ohjelmiston tai järjestelmän valmistaja korjauspäivitystä. Läheskään aina haavoittuvuuksiin ei kuitenkaan ole olemassa helppoa korjausta. Haavoittuvuuslöytö voi esimerkiksi olla niin tuore, ettei sille ole vielä julkaistu virallista päivitystä. Tällöin tulisi miettiä muita

rajoitustoimenpiteitä, kuten konfiguraatiomuutosta, haavoittuvan komponentin korvaamista toisella, haavoittuvuuden hyväksikäytön mahdollistavan ominaisuuden tilapäistä poiskytkentää tai kokonaisen palvelun sulkemista. Vähintäänkin tilannetta on valvottava tehostetusti.

Haavoittuvuuden korjaustoimien lisäksi pitää selvittää, onko haavoittuvuutta mahdollisesti jo ehditty hyväksikäyttää. Pahimmassa tapauksessa hyökkääjä saattaa olla jo päässyt järjestelmään ja avannut yhteyksiä myös eteenpäin. Erilaiset pääsylokit, lokienhallinta ja tietoturvatietoa hallinnoivat järjestelmät auttavat selvityksessä.

Ideaalitilanteessa haavoittuvuudet korjataan heti niiden löytymisen jälkeen. Käytännössä korjaustoimenpiteille on kuitenkin usein erilaisia esteitä tai hidasteita. Kaikenlainen päivitys- ja kehitystyö vaatii työaikaresursseja ja osaamista, jota ei välttämättä ole saatavilla kaikkina hetkinä tai ainakaan sopivaan hintaan. Seuraavissa kappaleissa käydään läpi Britannian kyberturvallisuuskeskuksen (NCSC 2016) mukaan neljä merkittävintä hidastetta tai estettä, jotka ovat

- kustannukset,
- käyttökatkot,
- yhteensopivuushaasteet ja
- operatiiviset riskit.

Laitteiden ja ohjelmistojen uusimisesta aiheutuvat menot ovat yleensä suhteellisen helposti mitattavia välittömiä kustannuksia. Kun esimerkiksi työaseman käyttöjärjestelmään ei enää saa päivityksiä tai verkkolaitetoimittaja lopettaa tukemasta edellisen sukupolven laitekantaansa, on käyttäjä pakotettu jatkamaan käyttöä ilman tukea tai ryhtymään laitteiden uusimiseen. Käyttökatkoilla tarkoitetaan päivityksien suorittamisesta aiheutuvia katkoja palvelun saatavuuteen. Ilman toimivaa kahdennusta palvelu ei ole käytettävissä huoltokatkojen aikana, mikä haittaa liiketoimintaa tai sisäisiä prosesseja. Lisäksi päivityksiin osallistuva henkilökunta on pois muista projekteista. Vakituiset huoltoikkunat esimerkiksi kerran kuukaudessa helpottavat päivitysten ja muiden huoltojen organisointia ja suorittamista. Erityisen kriittisessä tapauksessa normaalia huoltoikkunaa ei voi odottaa.

Yhteensopivuusongelmat viittaavat esimerkiksi tilanteisiin, joissa tarpeellinen ohjelmisto lakkaa toimimasta, kun alla oleva käyttöjärjestelmä päivitetään uudempaan. Muun muassa teollisuudessa on käytössä paljon juuri tiettyyn erikoistehtävään luotuja ohjelmistoja, joiden toimivuus alustan vaihtumisen jälkeen on epävarmaa. Tällöin alustapäivityksen jälkeen saatetaan joutua uusimaan myös kyseiset erikoisohjelmistot.

Operatiiviset riskit tarkoittavat erityisesti työtapojen muuttumista. Kun tietojärjestelmä muuttuu ratkaisevasti, se saattaa pakottaa muutokseen myös tutut työ- ja toimintatavat.

Haavoittuvuusskannaukset ovat vain yksi osa teknisen tietoturvan toimenpiteistä. Haavoittuvuusskannerin aktiivinenkaan käyttö ei estä hyökkäyksiä tai poista tarvetta esimerkiksi palomuureille. Haavoittuvuusskannaukset toimivat osana proaktiivista tietoturvatyötä, kuten koulutus ja uhkatiedon seuranta, mutta edelleen tarvitaan myös reaktiivista, hetkeen reagoivaa tietoturvaa, kuten palomuureja, verkonvalvontaa ja virustentorjuntaa.

## 4.3 Haavoittuvuusskannausten täydentäminen

Edellä esitellyillä haavoittuvuusskannauksillakaan ei voi löytää kaikkia haavoittuvuuksia. Seuraavassa käydään läpi joitain haavoittuvuustyyppejä, joiden olemassaolosta organisaation tulee olla tietoinen, ja joita sen tulee etsiä sekä hallita muilla tavoin.

Langattomien verkkojen haasteet vaativat erityishuomiota. Niiden erityinen hankaluus on verkon ulottuminen haluttujen rajojen ulkopuolelle. Seurauksia voivat olla esimerkiksi salakuuntelu ja häirintähyökkäys. Yrityksen toimitilojen läheisyyteen tai vastaanottoon laukussa tuotu verkonkuuntelulaite tai valetukiasema (*rogue*) saattaa lyhyessä ajassa kaapata runsaasti liikennettä. Yrityksen verkon nimellä perustettua valetukiasemaa voi käyttää käyttäjätunnusten keräämiseen. Eri laitevalmistajat ovat kehittäneet WLAN-kontrollereihinsa valetukiasemien tunnistusmenetelmiä. Valetukiasemaksi voidaan tulkita mikä tahansa vieras tukiasema, joka on organisaation oman langattoman verkon kuuluvuusalueella.

Verkon kattavuutta voi säätää tukiasemien sijainneilla, lähetystehoilla ja antennien suuntauksella. Yrityksen tulisi rajoittaa langattomaan verkkoon näkyviä palveluita. Esimerkiksi tiedostopalvelimiin pääsyn pitäisi edellyttää vahvempaa autentikointia, kuten VPN:ää ja kaksivaiheista tunnistautumista.

Teknisten rajapintojen lisäksi heikkoja kohtia on muuallakin. Haavoittuvuusskannauksilla huomaamatta jäävät kaikki ihmisten väliseen vuorovaikutukseen liittyvät ongelmakohdat ja vaikutuskanavat. Peltierin (2006) mukaan kaikista hyökkäystekniikoista juuri käyttäjään kohdistuva manipulointihyökkäys, eli social engineering, on vaikein torjua. Tutkimusten mukaan henkilöstön säännöllinen koulutus ja harjoittelu ovat tehokkaita keinoja kehittää henkilöstön valmiuksia manipulointihyökkäyksiä vastaan. Koulutuksella ja organisaation tietoturvapolitiikalla tulee ohjata turvallisiin työskentelytapoihin. Tähän lukeutuvat muun muassa turvallisten työskentelytapojen ja -välineiden valinta etätyöpisteissä, jotta salasanat ja muut luottamukselliset tiedot eivät ole urkittavissa.

Lisäksi tarvitaan haitallista vaikuttamista estävää tekniikkaa, kuten roskapostisuodatusta ja haittaohjelmatorjuntaa. (Purushotham & Gowthamaraj 2019)

Manipulointihyökkäysten määrä on viime vuosina kasvanut ja laatu parantunut, mikä on saanut tutkijat ja yritykset kiinnostumaan sosiaalisen puolustuskyvykkyyden tutkimisesta ja kehittämisestä. Keskeinen ongelma on ollut tunnistaa niin sanotut käyttäjähaavoittuvuudet, eli sosiaaliselle vaikuttamiselle ja social engineering -hyökkäyksille eniten alttiit henkilöt. Astakhova & Medvedev (2020) esittävät tutkimuksessaan, että henkilön alttius vaikuttamiselle on pääteltävissä hänen persoonallisuudestaan. Persoonallisuuden selvittämisessä tutkijat ovat käyttäneet tietolähteinä muun muassa kohdehenkilöiden sosiaalisen median profiileja. Koneoppimista on käytetty profiilien analysointiin ja kohdehenkilöille kohdistettujen viestien luomiseen. Tutkimuksessa henkilöille lähetettiin kohdennettuja viestejä juuri heitä kiinnostavista teemoista, ja seurattiin, ketkä klikkasivat viesteissä olleita kalastelulinkkejä ja syöttivät tietojaan kalastelusivustoille.

Menetelmään liittyy eettisiä ja oikeudellisia kysymyksiä. Suomessa työnantajalla on hyvin rajalliset oikeudet tutkia työnhakijan tai työntekijän käyttäytymistä internetissä ja sosiaalisessa mediassa. Kuvattu menettelytapa olisi työntekijän teknistä valvontaa. Jos työnantaja kerää henkilötietoja muualta kuin työntekijältä itseltään, työntekijältä on hankittava suostumus tietojen keräämiseen (Laki yksityisyyden suojasta työelämässä 13.8.2004/759). Tiedon hakemisessa internet-hauilla on vaarana samannimisten henkilöiden sekoittuminen.

## 4.4 Haavoittuvuustiedon seuraaminen

Oman ympäristön haavoittuvuuksista voi kerätä tietoa ja pysyä ajan tasalla myös muilla tavoilla, kuin suorittamalla haavoittuvuusskannauksia. Tässä luvussa esitellään näitä keinoja sekä pohditaan niiden toimivuutta ja soveltuvuutta.

### 4.4.1 Uhkatieto ja tilannekuva

Oman tietoteknisen toimintaympäristön tunteminen on tärkeää, jotta osaa välttää sitä uhkaavia tekijöitä ja varautua niihin. Tämä tarkoittaa ymmärrystä siitä, mitä verkkoon kytkettyjä ohjelmistoja, verkkolaitteita, päätelaitteita ja muita järjestelmiä organisaatiossa käytetään ja minkälaista tietoa niillä käsitellään. Tässä auttaa inventaarion hallinta. Organisaatiossa pitää olla käsitys siitä, miten huomataan ja korjataan, jos jostakin keskeisestä järjestelmästä löytyy vakava haavoittuvuus. Kaikkia organisaatioita pitäisi

kiinnostaa ainakin asiakastietojen käsittelyyn liittyvien järjestelmien tietoturvan taso. Tässä auttaa uhkatiedon järjestelmällinen kerääminen ja seuraaminen.

Uhkatieto on tietoa olemassa olevista ja potentiaalisista uhkista. Uhkatietoa tuottavat muun muassa tietoturvayhtiöt ja tutkijayhteisöt. Tietoa voi saada myös sosiaalisesta mediasta, uutissivustoilta ja ohjelmistovalmistajilta. Järjestelmien ylläpitäjien tulisi seurata aktiivisesti käyttämiensä ohjelmistojen haavoittuvuuksia ja päivityksiä. Monilla ohjelmistoilla on tuotekohtaiset sivustot ja postituslistat, joiden kautta ajankohtaista tietoa on saatavilla. Vaikka ohjelmistojen automaattipäivitykset eivät olisi käytössä tai mahdollisia, ainakin pitää olla tietoinen uuden päivityksen julkaisusta.

Uhkatietoa voidaan kerätä ja jakaa esimerkiksi *MISP - Open Source Threat Intelligence Platform* -järjestelmällä. MISP-järjestelmän keskeisimpiä ominaisuuksia ovat suurien tietomassojen automaattinen kerääminen ja indeksointi sekä tiedon jakaminen muiden vastaavien järjestelmien kanssa. (MISP Project 2021) Kun yhteen liitettyjä tietolähteitä on monia, voivat kaikki osallistujat hyötyä. Samalla on mahdollista hankkia esimerkiksi maarajat ja yrityksen toimialarajat ylittävää uhkatietoa, jota muuten olisi vaikea seurata. Jaetun tiedon laadusta ja muodosta huolehtiminen on tärkeää.

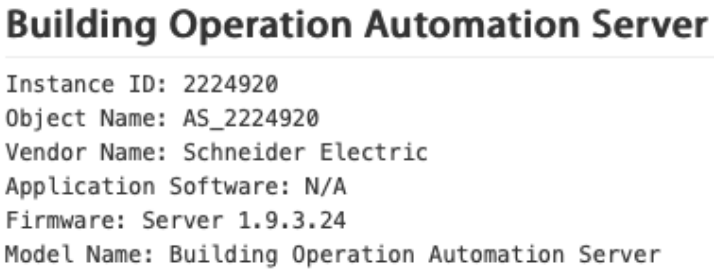
Uhkatiedon rinnalla voidaan puhua tilannekuvasta. Tilannekuvan tavoitteena on oikean ja reaaliaikaisen tiedon saatavuuden varmistaminen. Kaiken tiedon kerääminen ja analysoiminen ei ole tarpeellista eikä kustannustehokasta, mutta oman infrastruktuurin ja siinä käsiteltävän tiedon luonteen tunteminen on tärkeää. (Limnéll et al. 2014)

Useimmat tietomurrot ja muutkin tietoturvaongelmat välttää jo sillä, että perusasiat ovat kunnossa. Tyypillinen tietomurto ei ole kohdennettu tai erityisen kompleksinen, vaan kohdeympäristöön on ollut helppo murtautua.

### 4.4.2 Ulkoiset havaintolähteet

Tässä luvussa esitellään muutamia internetissä olevia palveluita, joita voi hyödyntää omaan organisaatioon kohdistuvien uhkien havaitsemisessa. Shodan on internetiin kytkettyjä laitteita indeksoiva hakukone. Palvelu skannaa jatkuvasti yleisimpiä portteja koko internetin IP-osoiteavaruudesta ja tallentaa havainnot tietokantaansa. Shodan lukee muun muassa palveluiden banner-metadataa. (Shodan 2020) Tietokanta on kenen tahansa selattavissa, ja rahaa vastaan saa käyttöönsä muun muassa monipuolisemmat hakutyökalut. Shodanin kaltaisen palvelun käyttöä omien järjestelmien haavoittuvuuksien löytämiseen voidaan kutsua passiiviseksi haavoittuvuuksien etsimiseksi (Samtani et al. 2016).

Kaikki tuntevat *Googlen* WWW-sivujen hakukoneena, mutta *World Wide Web* on vain yksi osa internetiä. Shodan löytää internetiin kytketyt laitteet, kuten FTP-palvelimet, webbikamerat, älytelevisiot ja kiinteistöautomaatiolaitteet. Shodan listaa löytämistään laitteista tietoja, kuten valmistajan, mallitiedot, IP-osoitteen, avoimet palvelut ja laitteessa ohjelmistojen versionumeroiden perusteella mahdollisesti olevat tunnetut haavoittuvuudet (niiden CVE-tunnisteet). Monesta laitteesta paljastuu arkaluontoistakin tietoa. Kirjoitushetkellä keväällä 2021 Shodan löytää pelkästään Suomesta yli miljoona laitetta. Kuvassa 26 on esimerkinomainen otos Shodanin löytämän etähallittavan kiinteistöautomaatiolaitteen tiedoista.




***Kuva 26.*** *Esimerkki Shodanin löytämästä etähallittavasta kiinteistöautomaatiolaitteesta (Shodan 2020).*

Shodanin keräämää tietoa voi käyttää moneen tarkoitukseen. Sitä voidaan hyödyntää oman verkon turvallisuuden kartoittamisessa, markkinaselvityksessä tai esimerkiksi haittaohjelmatutkimuksessa. Uuden haavoittuvuuden paljastuttua Shodan on yksi keino selvittää, miten laajasti haavoittuva sovellus on käytössä. Tämä auttaa ongelman laajuuden kartoittamisessa. Epäilemättä palvelun tietoja on mahdollista käyttää myös vääriin tarkoituksiin.

Shodanin löytöjä omasta verkosta kannattaa seurata, koska tiedot ovat myös internetin rikollisten käytettävissä. Se on kevyt ja helppo vaihtoehto itse suoritettaville haavoittuvuusskannauksille. Maksullisella tilillä voi tilata havainnot sähköpostitse. Kolmannen osapuolen skannaustietojen käyttämisen etu on siinä, ettei skannauksia tarvitse suorittaa itse. Huonona puolena uusien kohteiden skannaamista joutuu odottamaan, eikä varmuutta skannausten säännöllisyydestä tai tulosten oikeellisuudesta ole. Shodan skannaa vain yleisimpiä portteja, eikä Shodanin kaltainen palvelu pysty skannaamaan palomuurin takana olevia tai yksityisissä IP-osoitteissa olevia laitteita. Etenkin, jos oma verkkoympäristö muuttuu usein, tai palvelut ovat epätavallisissa porteissa, pelkkä Shodanin seuraaminen haavoittuvuuksien seuraamiseksi ei riitä. Toinen samankaltainen palvelu on Censys.io.

Shadowserver on tietoturvatutkijoiden yhteisö, jonka tavoitteena on edistää internetin turvallisuutta. Se etsii verkosta haavoittuvia palveluita sekä muita tietoturvauhkia ja raportoi havainnoistaan. Shadowserver toimii pääasiassa lahjoitusvaroin, ja se on tehnyt yhteistyötä muun muassa Euroopan poliisivirasto Europolin kanssa. (Europol 2015; Shadowserver Foundation 2021)

Palveluun ei voi liittyä, vaan Shadowserver skannaa jatkuvasti koko IPv4-osoiteavaruutta ja ilmoittaa havainnoistaan kyseisen IP-osoiteavaruuden omistajalle, kuten internet-operaattorille. Osoitteiden omistaja voi kuitenkin kieltäytyä skannauksista omalla ilmoituksellaan. Yhteystiedot Shadowserver poimii IP-osoitteiden julkisista whois-tietokannoista, jotka ovat alueellisten internet-rekistereiden (*Regional Internet Registry, RIR*), kuten Euroopassa RIPE NCC:n, ylläpidossa. (Shadowserver Foundation 2021) Verkko-operaattoreiden lisäksi omia IP-osoiteavaruuksia on muun muassa viranomaisilla, yrityksillä ja yliopistoilla. Internet-operaattorit voivat välittää saamansa raportit eteenpäin omille asiakkailleen. Suomessa myös Kyberturvallisuuskeskus vastaanottaa ja välittää eteenpäin Shadowserverin havaintotietoja. Vaikka itse ei voisi suoraan vastaanottaa Shadowserverin raportteja, sen verkkosivuilta voi seurata haavoittuvuuksiin liittyviä trendejä ja tilastoja.

## 4.5 Muut teknisen tietoturvan testaamisen keinot

Haavoittuvuusskannaus ei ole ainoa tapa testata organisaation teknisen tietoturvan tasoa. Tähän lukuun on koottu keinoja, joita voi käyttää haavoittuvuusskannausten rinnalla. Muista keinoista esitellään

- tietoturva-auditoinnit,
- harjoittelu,
- lähdekoodin analysointi,
- penetraatiotestaus eli pentesting sekä
- bug bounty -ohjelmat.

Perinteinen tietoturva-auditointi ostetaan ulkopuoliselta kumppanilta. Sen tarkoitus on selvittää tietoturvan tasoa joko koko organisaation laajuudelta tai jonkin pienemmän osa-alueen, kuten tuotteen, prosessin tai palvelun osalta. Auditoinnin jälkeen tilaaja saa raportin tietoturvan tilasta sekä parannusehdotuksia sen kehittämiseksi. Auditointi voi perustua dokumenttien katselmukseen tai käytännön toimintojen tarkastukseen sovitulta laajuudelta esimerkiksi tilaajan toimialaan liittyvät vaatimukset huomioiden.

Auditointi kohdistuu tyypillisesti valitun auditointikriteeristön vaatimusten täyttämiseen. Tunnettuja tietoturva-auditointikriteeristöjä ovat esimerkiksi ISO 27000 -sarjan standardit. Auditointi voi johtaa sertifikaattiin, joka toimii asiakkaille ja muille sidosryhmille osoituksena, että ulkopuolinen riippumaton taho on todennut organisaation tietoturvan tason arvioidulta osin arviointikriteeristön mukaiseksi. Se on tapa esittää tiiviissä muodossa, että tietoturvapolitiikka toteutuu. Sertifioinnin taustalla voi olla asiakasvaatimus tai se voi toimia kilpailutekijänä, jonka avulla erottautua kilpailijoista riskittömämpänä vaihtoehtona. (Kyberturvallisuuskeskus 2019)

Tietoturvan poikkeustilanteiden harjoittelu auttaa tositilanteisiin varautumisessa. Harjoittelu voi olla pienimuotoista, esimerkiksi varmuuskopioiden palautusten harjoittelua tai laajimmillaan koko organisaation reagointikykyä testaavaa toimintaa. Erilaisten harjoitusten avulla voidaan luoda kuvitteelliset olosuhteet esimerkiksi vakavan tietomurron käsittelyn harjoittelemiseksi. Harjoitus voi sisältää teknisen osuuden ohella myös muun muassa viestinnän harjoittelua. Myös oikea haavoittuvuusskannaus voi toimia harjoituksena. Skannauksen aikana voidaan harjoitella esimerkiksi skannauksen havaitsemista.

Lähdekoodin analysoinnilla tarkoitetaan sovelluksen ohjelmakoodin automaattista tai manuaalista tarkastusta esimerkiksi tietoturvapuutteiden löytämiseksi. Automaattisessa testaamisessa tähän tarkoitukseen tehty testaussovellus etsii ohjelmakoodin rakenteista haavoittuvuuksia, kuten ohjelmointivirheitä. Staattisessa testauksessa ohjelmistoa ei ajeta, vaan tarkistusta suoritetaan lähdekooditasolla. Dynaamisessa testauksessa puolestaan ohjelmaa ajetaan ja siihen kohdistetaan hyökkäyksiä tavoitteena löytää poikkeamia. Manuaalinen lähdekoodin analysointi on katselmointia, jossa kyseisen ohjelmointikielen osaaja käy läpi koodia. (DuPaul 2019)

Haavoittuvuusskannaukset ja penetraatiotestaus eli pentesting sekoitetaan joskus toisiinsa. Nagpure & Kurkuren (2017) mukaan näiden ero on siinä, että haavoittuvuusskannauksilla pyritään luomaan kokonaiskuva ympäristön haavoittuvuuksista ja niiden vakavuuksista, kun penetraatiotestauksessa puolestaan tutkitaan syvällisemmin ja konkreettisemmin, miten löydettyjä haavoittuvuuksia on mahdollista hyväksikäyttää. Automatisoitu haavoittuvuusskannaus voidaan siis nähdä eräänlaisena valmistelevana vaiheena pentestingille, joka on enemmän manuaalista työtä. Yksi tunnetuimmista penetraatiotestaustyökaluista on Metasploit.

Bug bounty -haavoittuvuuspalkkio-ohjelman perustaminen on yksi tapa kartoittaa tietoturvaheikkouksia omassa organisaatiossa. Menetelmää kutsutaan myös yhteisölliseksi tietoturvatestaamiseksi. Bug bounty -ohjelma on palkinto-ohjelma, johon yritys

kutsuu tietoturvatutkijoita etsimään järjestelmistään ja palveluistaan tietoturvahaavoittuvuuksia. Osallistujat voivat olla ammattimaisia tietoturvatutkijoita tai esimerkiksi alan harrastajia. Osallistujat raportoivat löytämistään haavoittuvuuksista. Yritykselle ohjelma on kustannustehokas vaihtoehto haavoittuvuuksien etsimiseen, sillä yleensä palkkiot maksetaan vain todennetuista haavoittuvuuksista. (Salmikivi 2020)

# 5. YHTEENVETO

Tämän diplomityön tavoitteena oli selvittää, miten haavoittuvuusskannaukset voivat auttaa organisaatiota sen tietoturvallisuuden kehittämisessä. Työssä käytiin läpi haavoittuvuusskannerin toimintaa ja selvitettiin, minkälaisia haavoittuvuuksia sen avulla voi löytää. Työ keskittyi IP-verkossa suoritettavien TCP- ja UDP-pohjaisten palveluiden skannaamiseen.

Esimerkkien avulla lukija saa kattavan yleiskuvan haavoittuvuusskannauksista ja niissä käytettävistä menetelmistä. Esimerkiksi haavoittuvuusskannausten suorittamista joko itse tai ostettuna palveluna harkitsevalle työ antaa käsityksen vaaditusta työmäärästä ja ulkoistuksessa huomioitavista seikoista.

Työ toteutettiin etupäässä kirjallisuusselvityksenä. Työtä olisi voinut syventää tutkimalla ja vertailemalla eri haavoittuvuusskannerituotteita tarkemmin ja käytännön skannausten tasolla. Kiinnostavia mittareita olisivat esimerkiksi skannereiden käytettävyyteen, kuormittavuuteen, nopeuteen ja havaintojen tarkkuuteen liittyvät tekijät. Tämän kaltaista vertailua hankaloittaa oleellisesti kaupallisten skannerituotteiden hintataso. Myös uusien haavoittuvuuksien tunnistusmenetelmien julkaisun nopeutta voisi vertailla. Skannereiden kattava vertailu pelkästään valmistajien antamien tietojen perusteella on vaikeaa.

Haavoittuvuuksien aihepiiristä tehtyjä tutkimuksia selaamalla huomaa, että muun muassa haavoittuvuuksien hallintaa on tutkittu paljon. Näissä tutkimuksissa näkökulma on yleensä vähemmän tekninen.

Tässä työssä havaittiin, että oikein toteutettuina haavoittuvuusskannaukset voivat olla kustannustehokas tapa löytää haavoittuvuuksia omista tietojärjestelmistä. Pelkkä skannaaminen ei kuitenkaan vielä riitä, vaan tulokset on analysoitava ja korjaukset vietävä käytäntöön. Aina yksinkertaista korjauspäivitystä ei ole olemassa, jolloin joudutaan pohtimaan muita keinoja mahdollisen uhkan lieventämiseksi.

On myös sellaisia haavoittuvuuksia ja heikkoja kohtia, joita tässä työssä esitellyillä skannauksilla ei voi löytää. Organisaation tulisi aktiivisesti pohtia, mitä juuri omassa ympäristössä voi jäädä huomaamatta ja alttiiksi hyökkäyksille. Eräs merkittävä ja nouseva uhka ovat verkon kautta henkilöstön sosiaaliseen vaikuttamiseen tähtäävät hyökkäykset. Tähän social engineering -ilmiöön paras puolustautumiskeino on henkilöstön säännöllinen koulutus.

Lopuksi todettakoon, että kerran turvalliseksi todettu ympäristö ei ole sitä välttämättä enää viikon tai kuukauden kuluttua. Tietoturvallisuuden ylläpidossa ja kehittämisessä

säännöllisyys sekä jatkuvuus ovat keskeisiä tekijöitä. Jokainen tietoturvauhka tulee ottaa huomioon sen vaatimalla vakavuudella.

# LÄHTEET

Amazon Web Services, AWS Customer Support Policy for Penetration Testing. Saatavissa (viitattu 13.4.2021): https://aws.amazon.com/security/penetration-testing/.

Andress J., The Basics of Information Security: Understanding the Fundamentals of InfoSec in Theory and Practice, Second Edition, Syngress Publishing, 2014.

Ascend Technologies, The Five Stages of Vulnerability Management, 26.3.2019. Saatavissa (viitattu 11.4.2021): https://blog.teamascend.com/stages-of-vulnerability-management.

A-studio, Pandemia lisäsi verkkovakoilua, Yleisradio, 16.3.2021. Saatavissa: https://areena.yle.fi/1-50646755.

Astakhova, L., Medvedev, I., Scanning the Resilience of an Organization Employees to Social Engineering Attacks Using Machine Learning Technologies, 2020 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT), IEEE, 2020. Saatavissa: https://ieeexplore.ieee.org/document/9117746.

Atymtayeva, L., Nurmyshev, S., Tulemissova, G., An Intelligent Approach and Data Management in Active Security Auditing Processes for Web Based Applications, Kazakh National Research Technical University KazNRTU, 2017. Saatavissa: http://www.scitepress.org/Papers/2017/65282/.

Cotton, M., Eggert, L., Touch, J., Westerlund, M., Cheshire, S., Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry, RFC 6335, Internet Engineering Task Force (IETF), 2011. Saatavissa: https://tools.ietf.org/html/rfc6335.

Death, D., Information Security Handbook, Packt Publishing, 2017.

DuPaul, N., Static Testing vs. Dynamic Testing, Veracode, 2019. Saatavissa: https://www.veracode.com/blog/secure-development/static-testing-vs-dynamic-testing.

Durumeric, Z., Wustrow, E., Halderman, J. A., ZMap: Fast Internet-wide Scanning and Its Security Applications, Proceedings of the 22nd USENIX Security Symposium, 2013.

Engebretson, P. The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy, Elsevie Science & Technology Books, Rockland, MA, 2013.

ENISA, European Union Agency for Cybersecurity, Glossary: Zero-Day. Saatavissa (viitattu 7.3.2021): https://www.enisa.europa.eu/topics/csirts-in-europe/glossary/zero-day.

Europol, Shadowserver Foundation Steps Up Cooperation with Europol to Combat Cybercrime, 10.5.2015. Saatavissa (viitattu 27.4.2021): https://www.europol.europa.eu/newsroom/news/shadowserver-foundation-steps-cooperation-europol-to-combat-cybercrime.

Feng, X., Li, Q., Han, Q., Zhu, H., Liu, Y., Cui, J., Sun, L., Active Profiling of Physical Devices at Internet Scale, 2016 25th International Conference on Computer Communication and Networks (ICCCN), IEEE, 2016. Saatavissa: https://ieeexplore.ieee.org/document/7568486.

Feng, X., Li, Q., Wang, H., Sun, L., Acquisitional Rule-based Engine of Discovering Internet-of-Things Devices, Proceedings of the 27th USENIX Security Symposium, 2018. Saatavissa: https://www.usenix.org/conference/usenixsecurity18/presentation/feng.

FIRST, Common Vulnerability Scoring System Version 3.1: Specification Document. Saatavissa (viitattu 12.12.2020): https://www.first.org/cvss/specification-document.

FIRST, Common Vulnerability Scoring System Version 3.1 Calculator. Saatavissa (viitattu 6.3.2021): https://www.first.org/cvss/calculator/3.1.

Gowda, S., Prajapati, D., Singh, R., Gadre, S., False Positive Analysis of software vulnerabilities using Machine learning, 2018 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), IEEE, 2018. Saatavissa: https://ieeexplore.ieee.org/document/8648633/.

Haber, M., Hibbert B., Asset Attack Vectors: Building Effective Vulnerability Management Strategies to Protect Organizations, Apress, 2018.

ISO/IEC 27000, Information technology - Security techniques - Information security management systems - Overview and vocabulary, International Standard, Fifth edition 2018-02, 2018.

Kaspersky, What is Ransomware? Security definitions, 2021. Saatavissa (viitattu 8.4.2021): https://www.kaspersky.com/resource-center/definitions/what-is-ransomware.

Kyberturvallisuuskeskus, Luottamuksen lähteillä - Näkökulmia tietoturvanstandardointiin ja sertifiointiin, 2019. Saatavissa: https://www.kyberturvallisuuskeskus.fi/sites/default/files/media/publication/Luottamuksen_lahteilla.pdf.

Kyberturvallisuuskeskus, Tietoturvan vuosi 2020: Kyberturvallisuuskeskuksen vuosikatsaus. Saatavissa (viitattu 18.3.2021): https://www.kyberturvallisuuskeskus.fi/sites/default/files/media/publication/Tietoturvan-vuosi-2020_210212_FIN.pdf.

Laki yksityisyyden suojasta työelämässä, 2 luku 13.8.2004/759, 2004. Saatavissa: https://www.finlex.fi/fi/laki/ajantasa/2004/20040759#L2P4.

Limnéll, J., Majewski, K., Salminen, M., Kyberturvallisuus, Docendo, 2014.

Little, E. G. & Rogova, G. L., An Ontological Analysis of Threat and Vulnerability, 9th International Conference on Information Fusion, IEEE, 2006. Saatavissa: https://ieeexplore.ieee.org/document/4086002.

Lyon G., The Official Nmap Project Guide to Network Discovery and Security Scanning. Saatavissa (viitattu 11.3.2021): http://www.nmap.org/book/toc.html.

Makrushin, D., The cost of launching a DDoS attack, Securelist, 2017. Saatavissa (viitattu 8.4.2021): https://securelist.com/the-cost-of-launching-a-ddos-attack/77784/.

McAfee, The Hidden Costs of Cybercrime, 2021.

MISP Project, MISP - Open Source Threat Intelligence Platform & Open Standards For Threat Information Sharing. Saatavissa (viitattu 9.5.2021): https://misp-project.org.

Mitre Corporation, About CVE, 2020. Saatavissa: https://cve.mitre.org/about.

Mitre Corporation, ATT&CK, 2021b. Saatavissa: https://attack.mitre.org.

Mitre Corporation, Common Weaknesses Enumeration, 2021a. Saatavissa: https://cwe.mitre.org/about.

Nagpure, S. & Kurkure, S., Vulnerability Assessment and Penetration Testing of Web Application, 2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA), IEEE, 2017. Saatavissa: https://ieeexplore.ieee.org/document/8463920.

Nappa, A., Johnson, R., Bilge, L., Caballero, J., Dumitras, T., The Attack of the Clones: A Study of the Impact of Shared Code on Vulnerability Patching, 2015 IEEE Symposium on Security and Privacy, IEEE, 2015. Saatavissa: https://ieeexplore.ieee.org/document/7163055.

NCSC, Vulnerability Management, National Cyber Security Centre, 23.9.2016. Saatavissa (viitattu 12.4.2021): https://www.ncsc.gov.uk/guidance/vulnerability-management.

NIST, Technical Guide to Information Security Testing and Assessment, NIST Special Publication 800-115, National Institute of Standards and Technology, 2008.

NIST, Glossary: Vulnerability, National Institute of Standards and Technology. Saatavissa (viitattu 5.12.2020): https://csrc.nist.gov/glossary/term/vulnerability.

NIST, CVE-2020-1472 Detail, National Vulnerability Database, National Institute of Standards and Technology, 2020. Saatavissa (viitattu 13.12.2020): https://nvd.nist.gov/vuln/detail/CVE-2020-1472.

NIST, CVSS Severity Distribution Over Time. National Institute of Standards and Technology, 2021. Saatavissa (viitattu 5.3.2021): https://nvd.nist.gov/general/visualizations/vulnerability-visualizations/cvss-severity-distribution-over-time.

Nmap, TCP FIN, NULL, and Xmas Scans (-sF, -sN, -sX), Nmap.org, 2021a. Saatavissa (viitattu 26.3.2021): https://nmap.org/book/scan-methods-null-fin-xmas-scan.html.

Nmap, TCP/IP Fingerprinting Methods Supported by Nmap, Nmap.org, 2021b. Saatavissa (viitattu 25.3.2021): https://nmap.org/book/osdetect-methods.html.

OpenVAS, OpenVAS - Open Vulnerability Assessment Scanner, 2021. Saatavissa (viitattu 4.4.2021): https://www.openvas.org/.

OWASP, OWASP Top Ten, The OWASP Foundation, 2021. Saatavissa (viitattu 25.4.2021): https://owasp.org/www-project-top-ten/.

Palmaers, T., Implementing a Vulnerability Management Process, SANS Institute, 2013.

Peltier, T., Social Engineering: Concepts and Solutions, Information Systems Security, EDPACS, 33:8, 2006.

Poliisi, Kyberrikokset - mitä kyber tarkoittaa? Saatavissa (viitattu 17.3.2021): https://poliisi.fi/kyberrikokset.

Ponemon Institute, Cost of a Data Breach Report 2020, IBM Security, 2020.

Purushotham, P., Gowthamaraj, R., Identification and prevention of social engineering attacks on an enterprise, 2019 International Carnahan Conference on Security Technology (ICCST), IEEE, 2019. Saatavissa: https://ieeexplore.ieee.org/abstract/document/8888441.

Qualys, Common causes of False Positive and False Negative detections in Vulnerability Management, 24.11.2020. Saatavissa (viitattu 11.4.2021): https://qualys-secure.force.com/discussions/s/article/000006461.

Reid, R., Gilbert, A., Using the Parkerian Hexad to Introduce Security in an Information Literacy Class, InfoSecCD '10: 2010 Information Security Curriculum Development Conference, 2010.

Rikoslaki, 38 luku 21.4.1995/578, 1995. Saatavissa: https://www.finlex.fi/fi/laki/ajantasa/1889/18890039001#L38.

Salmikivi, J., Miten Bug Bounty -ohjelma soveltuu tietoturvahaavoittuvuuksien ratkomiseen? OP Tech Podcast, Jakso #35, 2010. Saatavissa: https://op-careers.fi/content/optech35/.

Samtani, S., Yo, S., Zhu, H., Patton, M., Chen, H., Identifying SCADA vulnerabilities using passive and active vulnerability assessment techniques, 2016 IEEE Conference on Intelligence and Security Informatics (ISI), IEEE, 2016. Saatavissa: https://ieeexplore.ieee.org/document/7745438.

Sanastokeskus TSK ry, TEPA-termipankki, hakusana ‘haavoittuvuus’. Saatavissa (viitattu 5.12.2020): https://termipankki.fi/tepa/fi/haku/haavoittuvuus.

Scully, T., The cyber threat, trophy information and the fortress mentality. Journal of Business Continuity & Emerging Planning Volume 5 Number 3, 2011.

Shadowserver Foundation, Our Story, 2021. Saatavissa (viitattu 27.4.2021): https://www.shadowserver.org/who-we-are/.

Shirey, R., Internet Security Glossary, Version 2, RFC 4949, Internet Engineering Task Force (IETF), 2007. Saatavissa: https://tools.ietf.org/html/rfc4949.

Singh, K., Singh, A., Memcached DDoS Exploits: Operations, Vulnerabilities, Preventions and Mitigations, 2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS), IEEE, 2018. Saatavissa: https://ieeexplore.ieee.org/document/8586810.

Shodan, What is Shodan? 2020. Saatavissa (viitattu 23.3.2021): https://help.shodan.io/the-basics/what-is-shodan.

Sipior, J. C., Ward, B. T., A Framework for Information Security Management Based on Guiding Standards: A United States Perspective, Issues in Informing Science and Information Technology, Volume 5, 2008.

Spring, J., Hatleback, E., Householder, A., Manion, A., Shick, D., Time to Change the CVSS?, IEEE Security & Privacy, Volume 19, Issue 2, IEEE, 2021. Saatavissa: https://ieeexplore.ieee.org/document/9382369.

Sultan, S., Salman, A., Calcium Vulnerability Scanner (CVS): A Deeper Look, 2019. Saatavissa: https://arxiv.org/abs/1911.00950.

Suojelupoliisi, Kansallisen turvallisuuden katsaus 2020, Suojelupoliisin julkaisu, 2020. Saatavissa: https://supo.fi/documents/38197657/39761269/FI+Kansallisen+turvallisuuden+katsaus_2020.pdf.

Tai, W., How to Use VPR to Manage Threats Prior to NVD Publication, Tenable, 2020. Saatavissa (viitattu 11.4.2021): https://www.tenable.com/blog/what-is-vpr-and-how-is-it-different-from-cvss.

Tenable, The Nessus Family, 2021a. Saatavissa (viitattu 26.3.2021): https://www.tenable.com/products/nessus.

Tenable, Nessus User Guide, 2021b. Saatavissa (viitattu 26.4.2021): https://docs.tenable.com/Nessus.htm.

Trend Micro, A Constant State of Flux - Trend Micro 2020 Annual Cybersecurity Report, 2021.

United States Naval Academy, Cyber Science Department, Risks and Vulnerabilities. Saatavissa (viitattu 13.3.2021): https://www.usna.edu/CyberDept/sy110/calendar.php?type=class&event=3.

US-CERT, Alert (TA13-175A), Risks of Default Passwords on the Internet, 2016. Saatavissa (viitattu 3.5.2021): https://us-cert.cisa.gov/ncas/alerts/TA13-175A.

US-CERT, Alert (TA14-017A), UDP-Based Amplification Attacks, 2019. Saatavissa (viitattu 7.4.2021): https://us-cert.cisa.gov/ncas/alerts/TA14-017A.

VAHTI, Johdon tietoturvaopas, Valtionhallinnon tietoturvallisuuden johtoryhmä, VAHTI-julkaisu 2/2011, Valtiovarainministeriö, 2011.

Verizon, 2020 Data Breach Investigations Report (DBIR), 2021.

Wang, Y., Bai, Y., Li, L., Chen, X., Chen, A., Design of Network Vulnerability Scanning System Based on NVTs, 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), IEEE, 2020. Saatavissa: https://ieeexplore.ieee.org/abstract/document/9141812/.

Weidman, G., Penetration Testing: A Hands-On Introduction to Hacking, No Starch Press, Inc. 2014.

Yadav, G., Paul, K., Assessment of SCADA System Vulnerabilities, 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, 2019. Saatavissa: https://ieeexplore.ieee.org/abstract/document/8869541.