Ilari Kilkki

KUMPPANI-INTEGRAATIOARKKITEHTUURIN TIETOTURVAUHAT JA NIILTÄ SUOJAUTUMINEN

JYVÄSKYLÄN YLIOPISTO
INFORMAATIOTEKNOLOGIAN TIEDEKUNTA
2023

TIIVISTELMÄ

Kilkki, Ilari
Kumppani-integraatioarkkitehtuurin tietoturvauhat ja niiltä suojautuminen
Jyväskylä: Jyväskylän yliopisto, 2023, 38 s.
Kyberturvallisuus, pro gradu -tutkielma
Ohjaaja: Siponen, Mikko

Digitalisaatio on siirtänyt usean yrityksen liiketoiminnan kybermaailmaan. Yritykset etsivät yhä tehokkaampia tapoja arvon tuottamiseen sidosryhmilleen saavuttaakseen kilpailuetua. Alustaekosysteemiksi kutsutaan liiketoiminta- ja sovellusarkkitehtuurista mallia, jossa pääasiallinen toimija tarjoaa teknologisen alustan sidosryhmiensä käyttöön. Sidosryhmien rakentaessa omia sovellusympäristöjään alustan varaan muodostuu symbioottista arvontuottoa, josta jokainen ekosysteemin sidosryhmä hyötyy. Kumppanirajapinta on rajatuille sidosryhmille palveluna tarjottava teknologinen alusta. Kumppani-integraatioarkkitehtuuriksi kutsutaan arkkitehtuurista mallia, joka mahdollistaa alustaekosysteemin muodostamisen kumppanirajapintojen avulla. Kumppanirajapintojen avaaminen organisaation ulkopuolisille tahoille lisää organisaation kyberhyökkäyksille alttiin hyökkäyspinta-alan määrää. Tietovarojen turvallisuuden varmistaminen on alustaekosysteemin alustan ylläpitäjän liiketoiminnan turvaamisen kannalta välttämätöntä. Tietomurrot muodostavat organisaation jatkuvuutta uhkaavia riskejä, joihin täytyy varautua. Tutkimuksen toimeksiantona tilannut yritys X halusi vastauksen tutkimuskysymyksiin: ”Mitkä ovat kumppani-integraatioarkkitehtuurin kannalta keskeisimmät tietoturvauhat?” ja ”Millaisilla toimilla näitä tietoturvauhkia tulisi ennaltaehkäistä?” Vastausta tutkimuskysymyksiin etsittiin integroivan kirjallisuuskatsauksen avulla, koska menetelmä soveltuu pirstaleisen tiedon kokoamiseen eheäksi kokonaisuudeksi. Tutkielmassa esitellään REST-standardin mukaisten verkkorajapintojen yleisimmät tietoturvauhat ja parhaat käytännöt, joita hyödyntämällä niiltä voidaan suojautua. Kumppani-integraatioarkkitehtuurin käsite määritellään tutkielmassa parhaan tietoni mukaan ensimmäistä kertaa tieteellisessä kirjallisuudessa.

Asiasanat: kumppani-integraatioarkkitehtuuri, rajapinta, kumppanirajapinta, tietoturva, tietoturvauhka

ABSTRACT

Kilkki, Ilari
Information security threats of partner-integration architecture and how to defend against them
Jyväskylä: University of Jyväskylä, 2023, 38 pp.
Cyber security, Master's Thesis
Supervisor: Siponen, Mikko

Many companies have moved their businesses to the cyber world due to digitalization. Companies are actively looking for new and more effective ways of creating value for their stakeholders. A platform ecosystem is a business and software architecture model in which the platform administrator offers a technological platform for the use of its stakeholders. When stakeholders build their own software environments based on the platform, a symbiotic value generation is formed, from which every stakeholder in the ecosystem benefits. A partner application programming interface (API) is a technological platform offered as a service to limited stakeholders. Partner-integration architecture is an architectural model that enables the formation of a platform ecosystem with the help of partner APIs. Opening partner APIs to parties outside of the organization increases the amount of the organization's attack surface exposed to cyberattacks. Ensuring the safety of the information assets is essential for the safety of the business of the platform ecosystem's platform administrator. Data breaches constitute risks that threaten the continuity of the organization, which must be prepared for. Company X, which commissioned the study, was seeking answers to the research questions: "Which are the most important information security threats concerning partner-integration architecture?" and "What measures should be taken to prevent these information security threats?" An integrative literature review was chosen as the research method because it is suitable for assembling fragmented information into a whole. The thesis presents the most common information security threats of REST-based APIs and the best practices that can be used to defend against them. After extensive research on the subject, to my knowledge the concept of partner-integration architecture is defined in this thesis for the first time in scientific literature.

Keywords: partner-integration architecture, application programming interface, API, partner API, information security, information security threat

KUVIOT

TAULUKOT

SISÄLLYS

1 JOHDANTO

Yhteiskuntien nopea digitalisoituminen on johtanut lukuisten palveluiden siirtymiseen verkkoavaruuteen. Ohjelmistot, palvelut ja niiden tuotantomenetelmät ovat kehittyneet nopeasti, koska nopea pääsy digitaalisille markkinoille on tarjonnut organisaatioille usein merkittävää kilpailuetua kilpailijoihin nähden. Vielä muutama vuosikymmen sitten järjestelmäsuunnittelussa ei useinkaan otettu huomioon, kuinka järjestelmä saataisiin tarpeen vaatiessa toimimaan yhteistyössä muiden toimijoiden kehittämien järjestelmien kanssa (Gholami et al., 2017). Kybermaailman jatkuvasti monimutkaistuva verkottuminen on tehnyt integraatiotuesta keskeisen vaatimuksen yhä useammille järjestelmille. Yritysten välisessä integraatiossa hyödynnetään usein rajapintoja, jotka tarjoavat organisaation ulkopuolisille tahoille pääsyn ylläpitäjän määrittämiin toiminnallisuuksiin ja tietokantoihin (Heshmatisafa & Seppänen, 2023).

Digitaalisten palveluiden suunnittelussa on ollut keskiössä kysymys siitä, kuinka yritykset kykenevät tuottamaan mahdollisimman paljon arvoa asiakkailleen mahdollisimman tehokkaasti (Heshmatisafa & Seppänen, 2023). Maailman johtavat teknologiayritykset ovat monelta osin edelläkävijöitä myös ohjelmistoyritysten liiketoimintamallien kehittämisessä. Google, Amazon, Microsoft, Facebook ja Twitter tarjoavat ylläpitämiään alustoja asiakkaidensa käyttöön, ja asiakkaat vastavuoroisesti kehittävät omia ohjelmistojaan näiden alustojen varaan. Alustojen varaan muodostunutta ekosysteemiä kutsutaan alustaekosysteemiksi, kun sekä alustan tarjoaja että alustaa hyödyntävät liiketoimintakumppanit saavat lisäarvoa toistensa suorittamasta kehitystyöstä. Alustaekosysteemi mahdollistaa merkittävän tehokkaan arvontuoton, koska lukemattomat liiketoimintakumppanit voivat kehittää uusia järjestelmiä samanaikaisesti, ja nämä järjestelmät tuottavat arvoa kaikille ekosysteemin sidosryhmille. (Ceccagnoli & Forman, 2012). Myös kaikkien sidosryhmien asiakkaat hyötyvät alustaekosysteemistä, koska sen ansiosta yritykset kykenevät tarjoamaan aiempaa kattavampia palveluita (Ceccagnoli & Forman, 2012). Alustaekosysteemin arvontuottomalli on symbioottinen.

Teknologiajättien esimerkkiä seuraten yhä useammat yritykset pyrkivät muodostamaan omat alustaekosysteeminsä. Tämä siirtymä vaatii yrityksen lii-

ketoiminta- ja ohjelmistoarkkitehtuurien osittaista uudelleenmäärittelyä. Kumppani-integraatioarkkitehtuuriksi kutsutaan arkkitehtuurista mallia, jossa alustaekosysteemin muodostamiseen pyritään liiketoimintakumppaneille avattavien rajapintojen avulla, jotka olivat aiemmin organisaation sisäisessä käytössä. Rajapintojen avaaminen tuo mukanaan uusia tietoturvauhkia, joihin organisaatioiden on varauduttava (Munsch & Munsch, 2020). Tietovarojen luottamuksellisuuden, eheyden tai saatavuuden vaarantuminen muodostavat merkittäviä riskejä kaikille alustaekosysteemin sidosryhmille. Varmistaakseen liiketoimintansa jatkuvuuden on alustan ylläpitäjän varmistettava alustansa tietoturvallisuus. Tämä tutkielma kokoaa yhteen pirstaleista tutkimustietoa kumppani-integraatioarkkitehtuurin kannalta keskeisimmistä tietoturvauhista ja parhaista käytännöistä, joiden avulla niiltä voidaan puolustautua.

1.1 Tutkimuksen tausta

Tutkielman tilannut yritys X (myöhemmin *toimeksiantaja*) on siirtymässä hyödyntämään kumppani-integraatiopohjaista arkkitehtuurimallia, eli avaamassa aiemmin sisäisessä käytössä olleita rajapintojaan kolmansille osapuolille kasvattaakseen liiketoimintamahdollisuuksiaan. Rajapintojen kautta toimeksiantajan liiketoimintakumppanit pääsevät käsiksi tiettyihin toimeksiantajan ylläpitämien järjestelmien toimintoihin ja omistamaansa asiakasdataan, joka on tallennettu toimeksiantajan tietokantoihin. Tavoitetilanteessa toimeksiantaja onnistuu muodostamaan oman alustaekosysteeminsä, jossa kaikki sidosryhmät kykenevät tarjoamaan aiempaa parempia palveluja asiakkailleen, mikä johtaa korkeampaan asiakastyytyväisyyteen ja sidosryhmien asiakaspohjien kasvuun. (Heshmatisafa & Seppänen, 2023; Ceccagnoli & Forman, 2012).

Rajapintojen avaaminen kolmansille osapuolille lisää organisaation kyberhyökkäyksille alttiin hyökkäyspinta-alan määrää ja edellyttää täten arkkitehtuurisista muutoksista aiheutuvien tietoturvauhkien tunnistamista, jotta niihin voidaan varautua ennaltaehkäisevästi (Munsch & Munsch, 2020). Kyberhyökkäysten ennaltaehkäisemiseksi organisaation kannattaa noudattaa parhaita käytäntöjä, jotka on todettu toimiviksi (Andersson, Hedström & Karlsson, 2022).

Tämä tutkielma tarjoaa kattavan listauksen kumppani-integraatioarkkitehtuurin kannalta keskeisistä tietoturvauhista ja parhaista käytänteistä, joita hyödyntämällä niitä voidaan ennaltaehkäistä. Tutkielmalle oli selkeä tarve, koska aiempi tutkimustieto aiheesta oli hyvin pirstaloitunutta, eikä kumppani-integraatioarkkitehtuurin käsitettä ollut parhaan tietoni mukaan määritelty aiemmin tieteellisissä julkaisuissa. Kyberturvallisuus tieteenalana on jatkuvan muutoksen kourissa, ja sen tulee kyetä reagoimaan nopeasti uusiin innovaatioihin ja niiden mukanaan tuomiin tutkimusaiheisiin, jollaista tutkielma edustaa.

1.2 Tutkimusongelmat

Tutkimusongelmia on kaksi: ”*Mitkä ovat kumppani-integraatioarkkitehtuurin kannalta keskeisimmät tietoturvauhat?*” ja ”*Millaisilla toimilla näitä tietoturvauhkia tulisi ennaltaehkäistä?*”

Kumppani-integraatioarkkitehtuurin kannalta keskeisiä tietoturvauhkia käsittelevä aiempi tutkimuskirjallisuus on pirstaloitunutta. Tästä syystä tutkimusmenetelmäksi valikoitui integroiva kirjallisuuskatsaus, joka soveltuu hyvin kokonaiskuvan rakentamiseen verrattain uudesta asiakokonaisuudesta (Salminen, 2011; Torraco, 2016). Tutkimuksen laajuuden ulkopuolelle rajattiin kumppanirajapintojen kautta kulkevan informaation omistajuus- ja vastuukysymykset, koska niistä seuraavat ongelmat ovat luonteeltaan juridisia. Lisäksi toimeksiantajan edustaja rajasi tutkimuksen ulkopuolelle käyttöliittymätason tietoturvauhat, kuten esimerkiksi injektio- ja Cross Site Scripting -haavoittuvuudet. Näin ollen tutkielma keskittyy pääasiassa verkkorajapintojen kannalta keskeisiin tietoturvauhkiin. Koska kirjallisuuskatsaukseen valittu aineisto käsitteli enimmäkseen REST-standardin mukaisia verkkorajapintoja, rajattiin SOAP-standardin mukaisille verkkorajapinnoille ominaiset tietoturvauhat tutkielman laajuuden ulkopuolelle.

1.3 Tutkielman rakenne

Tutkielman johdanto-osiossa avataan tutkielman taustaa, motivaatiota ja tutkimusongelmia. Toisessa luvussa määritellään tutkielman kannalta keskeiset käsitteet alan aiempien tutkimusten pohjalta. Kolmannessa luvussa esitellään tutkimusmenetelmäksi valittu integroiva kirjallisuuskatsaus ja tutkielman tutkimusprosessi, sekä käydään läpi kirjallisuuskatsauksen tulokset. Neljännessä luvussa esitellään kirjallisuuskatsauksessa tunnistetut tietoturvauhat. Viidennessä luvussa käydään läpi parhaat käytännöt, joita hyödyntämällä aiemmin tunnistetuilta tietoturvauhilta kyetään puolustautumaan. Kuudennessa luvussa käydään läpi tutkimuksen tulokset, arvioidaan tutkimuksen reliabiliteettia ja validiteettia ja arvioidaan millaista lisäarvoa tutkielma tuottaa tietotekniikan ja kyberturvallisuuden tieteenaloille, sekä esitetään jatkotutkimusaiheita. Tutkielman lopusta löytyy luettelo lähdemateriaalina käytetystä aineistosta.

2 KÄSITTEIDEN MÄÄRITTELY

Tässä luvussa määritellään rajapinnan, kumppani-integraatioarkkitehtuurin ja tietoturvan käsitteet aiemman tutkimuskirjallisuuden pohjalta.

2.1 Rajapinta

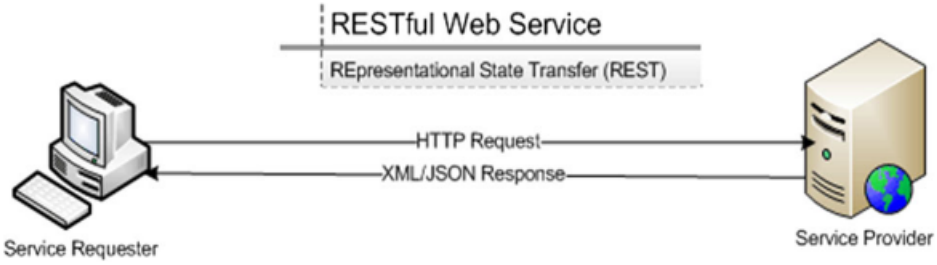
Rajapinnat (engl. *Application Programming Interface*) ovat erilaisten ohjelmien välisen yhteistoiminnan mahdollistajia. Ohjelmat voivat esimerkiksi välittää dataa järjestelmästä toiseen rajapintojen kautta, tai rajapinnat voivat avata ulkopuolisille ohjelmille pääsyn toisen järjestelmän toteuttamiin toimintoihin. (Lindman et al., 2020). Internetin kautta tavoitettavat verkkorajapinnat ovat arkkitehtuuriltaan yleisimmin REST-pohjaisia (*REpresentatiol State Transfer*). REST-rajapinnat koostuvat päätepisteistä (engl. *endpoint*) joiden kautta pääsee käsiksi rajapinnan takana oleviin toiminnallisuuksiin HTTP-pyyntöjen (*Hypertext Transfer Protocol*) avulla. REST-pohjaisten verkkopalveluiden toimintaa on havainnollistettu kuviossa 1 sivulla 11 (Beer & Hassan, 2017; Ehsan et al., 2022).

Rajapinnat voidaan jakaa kolmeen eri tyyppiin sen mukaan, keille kaikille on annettu valtuudet rajapinnan toimintojen käyttämiseen. Yksityinen rajapinta (engl. Private API) on määritelty vain tarkkaan rajatun käyttäjäjoukon käyttöön, kuten esimerkiksi pelkästään organisaation sisäisesti käytettäväksi, jotta vain tunnistetut ja valtuutetut henkilöt pääsevät käsiksi yrityksen tietokantoihin. Yksityiset rajapinnat toimivat pääsääntöisesti vain organisaation sisäisessä verkossa. (Hussain et al., 2020; OpenAPIHub, 2022).

Julkinen rajapinta (engl. *Public API*) on lähtökohtaisesti kenen tahansa käytettävissä, vaikka käyttöoikeuksia voidaankin rajoittaa esimerkiksi vaatimalla korvausta rajapinnan käyttämisestä (Hussain et al., 2020; OpenAPIHub, 2022). Julkista rajapintaa hyödyntääkseen käyttäjä joutuu usein hyväksymään käyttöehtosopimuksen, jossa määritellään tarkasti mihin tarkoituksiin rajapinnan toiminnallisuuksia saa hyödyntää (Heshmatisafa & Seppänen, 2023). Julki-

sia rajapintoja ylläpitävät tahot tarjoavat pääsyn rajapintoihinsa internetin välityksellä.

Kumppanirajapinnaksi (engl. Partner API) kutsutaan yksityistä rajapintaa, joka on avattu rajapintaa ylläpitävän organisaation ulkopuolisten tahojen käyttöön. Ulkopuoliset tahot on tarkoin määritelty organisaation toimesta, ja he sitoutuvat noudattamaan tiukkoja käyttöehtosopimuksia. Kumppanirajapintojen ylläpitäminen edellyttää organisaatiolta vahvaa panostusta tietoturvasta huolehtimiseen, koska rajapinnat avaavat suoran yhteyden organisaation sisäisten järjestelmien toimintoihin ja tietovaroihin. (Hussain et al., 2020; OpenAPIHub, 2022).



KUVIO 1. REST-based Web Service (Beer & Hassan, 2017).

2.2 Kumppani-integraatioarkkitehtuuri

Tässä alaluvussa määritellään ensin yritysintegraation, kumppani-integraation, arkkitehtuurin ja alustaekosysteemin käsitteet, jotta niiden pohjalta kyetään määrittelemään myös kumppani-integraatioarkkitehtuurin käsite.

Yritysintegraatioksi kutsutaan prosessia, jonka avulla pyritään varmistamaan yrityksen tavoitteiden kannalta kriittinen vuorovaikuttaminen (Chen, Doumeingts & Vernadat, 2008). Alsenen (1999) mukaan yritysintegraatio muodostuu niistä toimista, joiden avulla suuremman kokonaisuuden muodostavat pienemmät liiketoiminnan yksiköt saadaan toimimaan yhdessä. Kumppani-integraatio viittaa tilanteeseen, jossa kumppanirajapintoja ylläpitävä taho mahdollistaa molemminpuolisen vuorovaikutuksen liiketoimintakumppaneidensa kanssa kumppanirajapintojen avulla.

Arkkitehtuuri voidaan määritellä kuvaukseksi monimutkaisen järjestelmän komponenteista, niiden toiminnasta ja vuorovaikutussuhteista (Kuusisalmi, 2019). Kumppani-integraatioarkkitehtuurin käsite pitää sisällään sekä liiketoiminta- että ohjelmistoarkkitehtuuriset näkökulmat. Kumppanirajapintoja palveluna tarjoavan toimijan tavoitteena on kasvattaa liiketoimintamahdollisuuksiaan tarjoamalla laadukkaita alustoja, joiden varaan kolmannet osapuolet voivat kehittää omia sovellusekosysteemejään (Lindman et al., 2020; Heshmatisafa & Seppänen, 2023). Tämä tavoitetilanne tunnetaan nimellä alustaekosysteemi (engl. *platform ecosystem*), jossa teknologisen alustan välityksellä toistensa kanssa vuorovaikutuksessa olevat yritykset tuottavat toiminnallaan lisäarvoa toisilleen (Ceccagnoli et al., 2012; Kuusisalmi, 2019). Toimivassa alustaekosys-

teemissä kaikki sidosryhmät kykenevät tarjoamaan asiakkailleen aiempaa parempia palveluja, mikä johtaa sekä asiakastyytyväisyyden että sidosryhmien asiakaspohjien kasvuun (Lindman et al., 2020). Ilmiötä voidaan kutsua symbioottiseksi arvontuotoksi, koska kunkin sidosryhmän tuottama lisäarvo tuottaa potentiaalisesti lisäarvoa myös muille sidosryhmille.

Kumppani-integraatioarkkitehtuuri on kuvaus alustaekosysteemin muodostamiseen kumppanirajapintojen avulla tähtäävästä liiketoimintamallista sekä sen edellyttämästä teknologisesta toimintaympäristöstä.

2.3 Tietoturva

Tietoturva voidaan määritellä toimiksi, joilla varmistetaan tiedon luottamuksellisuus, eheys ja saatavuus. Tietoturvan tavoitteena on varmistaa liiketoiminnan jatkuvuus ja minimoida realisoituvien tietoturvaongelmien liiketoiminnalle aiheuttamat vahingot. (von Solms & van Niekerk, 2013). Käytännönläheisemmän määritelmän tarjoaa yhdysvaltalainen Committee on National Security Systems, jonka mukaan tietoturva on informaation ja sen kriittisten elementtien, kuten järjestelmien ja informaation käyttämiseen, tallentamiseen ja välittämiseen käytettävien laitteiden turvaamista (Whitman & Mattord, 2017).

Tieto on luottamuksellista, kun se on suojattu tarkoituksenmukaisesti niin, että pelkästään valtuutetut tahot pääsevät siihen käsiksi. Tietomurrosta, eli tiedon luottamuksellisuuden murtumisesta, puhutaan kun valtuuttamaton taho pääsee käsiksi tietovaroihin. Tiedon luottamuksellisuutta voidaan varjella suojaamalla tiedon säilytykseen käytettävät ohjelmistot ja laitteet, määrittelemällä tietovaroille salausluokat, organisaation laajuisen turvallisuuspolitiikan laatimisella ja henkilöstön kouluttamisella aiheen tiimoilta. (Whitman & Mattord, 2017).

Tiedon eheydellä viitataan sen sisällön muuttumattomuuteen eli korruptoitumattomuuteen. Tieto voi korruptoitua tahattomasti erilaisten laite-, ohjelmisto- tai tiedonsiirto-ongelmien takia, mutta tietoa voidaan pyrkiä korruptoimaan myös tahallisesti erilaisten kyberhyökkäysten avulla. (Whitman & Mattord, 2017). Vuonna 2016 Venäjän tiedustelupalvelu hyökkäsi useiden ukrainalaisten pankkien ja yritysten järjestelmiin Petya -nimisen haittaohjelman avulla, joka korruptoi järjestelmien tietovarat, tehden niistä käyttökelvottomia ja lamauttaen näin yhteiskunnan toimintaa (Crosignani, Macchiavelli & Silva, 2022).

Tieto määritellään saatavilla olevaksi, kun valtuutetut tahot pääsevät siihen käsiksi häiriöittä, ja saavat tiedon käyttöönsä haluamassaan formaatissa. Tiedon saatavuutta uhkaavat eheyden tapaan erilaiset laite-, ohjelmisto- ja tiedonsiirto-ongelmat, sekä varsinkin palvelunestohyökkäykset. (Whitman & Mattord, 2017).

Tietoturvasta puhuttaessa on huomioitava, että suomen kielen sana *tieto* ei vastaa merkitykseltään täysin englannin kielen sanaa *information*, vaikka *information security* suomennetaankin *tietoturvaksi*. Tieto on yläkäsite, joka pitää si-

sällään datan (esimerkiksi yksittäinen kirjain), informaation (esimerkiksi yksittäinen sana), tietämyksen, ymmärryksen ja viisauden (Suomalainen asiasanasto- ja ontologiapalvelu Finto, 2018). Turvattavalla tiedolla viitataan tässä tutkielmassa tietovaroihin (engl. *information asset*), jotka voivat koostua mistä tahansa tiedon alakäsitteistä, jotka voidaan määritellä arvokkaiksi organisaation tai yksilön näkökulmasta (Whitman & Mattord, 2017).

On tarpeen huomioida myös erot käsitteiden *tietoturva* ja *kyberturva* välillä. Vaikka termejä käytetäänkin usein toistensa synonyymeinä, pitää kyberturvan käsite sisällään tietoturvan lisäksi myös esimerkiksi kyberterrorismin, nettikiusaamisen ja laittoman tekijänoikeudella suojatun materiaalin jakamisen kaltaisilta rikoksilta suojautumisen. Vastaavasti *kyberuhka* on laajempi käsite kuin *tietoturvauhka*, koska se sisältää kaikki kybertoimintaympäristöön kohdistuvat uhat, myös tietoturvauhat. (von Solms & van Niekerk, 2013).

3 TUTKIMUKSEN TOTEUTUS

Tämän luvun ensimmäisessä alaluvussa esitellään tutkimusmenetelmäksi valittu integroiva kirjallisuuskatsaus ja perustellaan sen valinta. Toisessa alaluvussa esitellään ensin tutkielman tutkimusprosessi, jonka jälkeen luvun alaluvuissa käydään läpi aineiston haku, haun tulokset ja aineistojen valinta, ja aineiston analyysimenetelmä. Viimeisessä alaluvussa käydään läpi kirjallisuuskatsauksen tulokset.

3.1 Tutkimusmenetelmän valinta

Kirjallisuuskatsaus on tutkimusmenetelmä, jonka avulla muodostetaan uusia tutkimustuloksia aiempien tutkimusten pohjalta. Kirjallisuuskatsaukset jaetaan kuvaileviin ja systemaattisiin kirjallisuuskatsauksiin, sekä meta-analyyseihin. Narratiiviset kirjallisuuskatsaukset sallivat väljempien tutkimuskysymysten käsittelyn kuin systemaattiset kirjallisuuskatsaukset ja meta-analyysit. (Salminen, 2011). Kuvailevat kirjallisuuskatsaukset jaetaan narratiivisiin ja integroiviin kirjallisuuskatsauksiin; narratiiviset katsaukset muodostavat helppolukuisen katsauksen aiempaan tutkimustietoon, kun taas integroivat mahdollistavat laaja-alaisen ja systemaattisen perehtymisen käsiteltävään aihealueeseen. (Salminen, 2011). Tämän tutkielman tutkimuskysymyksiin vastaaminen edellytti laaja-alaista perehtymistä aiempaan tutkimuskirjallisuuteen kyberturvallisuudesta. Näiden seikkojen pohjalta tutkimusmenetelmäksi valikoitui integroiva kirjallisuuskatsaus, joka soveltuu erinomaisesti pirstaleisen tiedon kokoamiseen eheäksi kokonaisuudeksi (Salminen, 2011; Karjalainen & Vesalo, 2015). Integroivat kirjallisuuskatsaukset pyrkivät usein esittämään kritiikkiä tarkasteltavaa kirjallisuutta kohtaan, mutta kriittisyys ei ole välttämätöntä (Salminen, 2011; Torraco, 2016). Tämän tutkielman tutkimuskysymyksiin vastaaminen ei edellytä lähtökohtaisen kriittistä otetta lähdekirjallisuuteen, koska tavoitteena on tunnistaa kumppani-integraatioarkkitehtuurille ominaisia tietoturvauhkia ja keinoja, joiden avulla niiltä voidaan suojautua.

3.2 Tutkimusprosessi

Integroiva kirjallisuuskatsaus koostuu viidestä vaiheesta; tutkimuskysymysten laatimisesta, aineiston keräämisestä, aineiston arvioinnista, aineiston analysoinnista ja tulkitsemisesta sekä tulosten esittämisestä. Aineistoa kerätään mahdollisimman laajasti, jotta sen pohjalta kyetään muodostamaan kokonaisvaltaista synteesiä. Tulokset esitetään selkeästi ja ymmärrettävästi. (Karjalainen & Vesalo, 2015; Salminen, 2011).

Tutkimusprosessi alkoi tutkimuskysymysten tarkalla määrittelyllä yhteistyössä toimeksiantajan kanssa. Tutkimuskysymyksiksi muodostuivat: ”Mitkä ovat kumppani-integraatioarkkitehtuurin kannalta keskeisimmät tietoturvauhat?” ja ”Millaisilla toimilla näitä tietoturvauhkia tulisi ennaltaehkäistä?”

Ennen tiedonhakuprosessin aloittamista tutkimuksen laajuutta rajattiin kahteen otteeseen. Juridiset vastuukysymykset tietovarojen omistajuudesta rajattiin tutkimuksen ulkopuolelle, koska niiden muodostamat uhat ovat luonteeltaan liiketoiminnallisia. Lisäksi toimeksiantajan edustaja rajasi tutkimuksen ulkopuolelle käyttöliittymätason tietoturvahaavoittuvuudet, jotka altistavat järjestelmät esimerkiksi Cross-Site Scripting- ja injektiohyökkäyksille.

3.2.1 Aineiston haku

Tutkimuskysymysten muodostamisen jälkeen alkoi lähdemateriaalin etsintä verkko-osoitteesta https://jyu.finna.fi löytyvällä Jyväskylän yliopiston kirjaston hakukoneella. Haut keskitettiin ProQuest Central, DOAJ ja Elsevier ScienceDirect tietokantoihin, koska ne osoittautuivat selkeästi kattavimmiksi tietokannoiksi valittavissa olleista vaihtoehdoista. ProQuest Central on 47 tietokantaa sisältävä, maailman laajin tieteellisen tiedon tietokanta (LibGuides, 2023). DOAJ sisältää tuhansia tieteellisiä julkaisuja, joihin pääsee käsiksi vapaasti ja jotka sisältävät vain vertaisarvioituja tutkimusartikkeleita (DOAJ, 2023). ScienceDirect sisältää yli 1,4 miljoonaa vertaisarvioitua tutkimusartikkelia, joita voi lukea vapaasti (ScienceDirect, 2023). Haut rajattiin sisältämään pelkästään vertaisarvioituja artikkeleita, jotka ovat kokonaan luettavissa verkkolähteistä.

Mahdollisimman kattavan otannan julkaistusta tieteellisestä kirjallisuudesta saisi sisällyttämällä aineiston etsintään mahdollisimman useilla kielillä kirjoitettuja artikkeleita (Karjalainen & Vesalo, 2015). Tämän tutkimuksen tekemiseen varatun rajallisen aikamäärän takia etsittävä lähdeaineisto on rajattu vain suomen tai englannin kielellä kirjoitettuihin artikkeleihin. Niin kutsuttu harmaa kirjallisuus tarkoittaa konferenssijulkaisuja, opinnäytetöitä ja väitöskirjoja (Karjalainen & Vesalo, 2015). Tutkielman lähdeaineistoksi päätettiin hyväksyä myös harmaata kirjallisuutta ja kirjoja, koska aineiston otannasta pyrittiin saamaan mahdollisimman kattava.

Lähdeaineiston laadukkuuden varmistamiseksi relevanteiksi määriteltyjen artikkelien julkaisijoiden JUFO-luokitukset tarkastettiin Julkaisufoorumin Julkaisukanavahaku-työkalun avulla. Julkaisufoorumi on suomalaisen tiedeyhteisön toteuttama, tutkimuksen laadunarviointia tukeva julkaisukanavien tasoluokitus, joka luokittelee tieteellisiä julkaisuja asteikolla 1–3, jolla korkein arvosana on 3 (Julkaisufoorumi, 2023). Mikäli artikkelin julkaisijan tasoluokitus oli 0 tai julkaisua ei löytynyt Julkaisukanavahaun kautta ollenkaan, artikkeli hylättiin. Lähdeaineiston tarkemmat hyväksymiskriteerit on esitelty alla olevassa listauksessa.

Hyväksymiskriteerit

1. Aineisto on relevanttia tutkimuskysymysten kannalta
2. Aineiston julkaisukieli on suomi tai englanti
3. Englanninkielinen aineisto: tutkimusartikkeli, jonka julkaisijan JUFO-luokitus on 1 tai parempi, tai kirja
4. Suomenkielinen aineisto: kirja, pro gradu -tutkielma, väitöskirja, tutkimusartikkeli tai korkeakoulun julkaisu
5. Julkaisun lähdeluettelo on saatavilla

Taatakseen tutkimuksen luotettavuuden, tulee tutkijan olla lähdekriittinen kerätessään lähdeaineistoa ja pyrittävä parhaansa mukaan varmistamaan aineiston objektiivisuus ja totuudenmukaisuus (Karjalainen & Vesalo, 2015). Tämän takia lähdeaineiston hyväksymiskriteeriksi kirjattiin edellytys saatavilla olevasta lähdeluettelosta.

Tutkimuskysymyksiin vastauksia haettaessa hakutuloksista rajattiin ulos kaikki ennen vuotta 2013 julkaistut tutkimukset, jotta lähdemateriaali olisi mahdollisimman tuoretta. Uuden tiedon käyttäminen parantaa tutkimuksen luotettavuutta (Karjalainen & Vesalo, 2015). Molempien tutkimuskysymysten vastauksia etsittiin samoilla hakulausekkeilla, koska varsin usein tietoturvauhkaa tai -haavoittuvuutta käsittelevä artikkeli käsittelee myös siltä suojautumista.

Suomenkielistä aineistoa etsittäessä käytettiin hakusanoja ”kumppanirajapinta”, ”rajapinta”, ”rest rajapinta”, ”kyberturvallisuus”, ”tietoturva”, ”turvallisuus”, ”haavoittuvuus”, ”tietoturvahaavoittuvuus”, ”tietoturvauhka”, ”kyberhyökkäys”, ”puolustautuminen” ja ”suojautuminen”. Englanninkielistä aineistoa haettiin hakusanoilla ”partner integration architecture”, ”partner api”, ”partner application programming interface”, ”api”, ”rest api”, ”restful api”, ”application programming interface”, ”cyber security”, ”information security”, ”infosec”, ”cyber”, ”vulnerability”, ”threat”, ”attack”, ”security”, ”prevention” ja ”defend”. Hakusanoja yhdistettiin ”AND” operaattorilla.

3.2.2 Aineiston haun tulokset ja aineistojen valinta

Tiedonhakuprosessi alkoi kumppani-integraatioarkkitehtuurin käsitteen määritelmän etsimisellä. Hakutuloksia hakulausekkeella ”partner integration architecture” löytyi 0 kappaletta. Hakulausekkeella ”partner AND ”integration architecture”” löytyi 103 hakutulosta, joista hyväksymiskriteerit täytti kolme aineistoa. Hakulausekkeella ”api AND ”integration architecture”” löytyi 120 hakutulosta, joista hyväksyttiin neljä. Hakulausekkeella ””application programming interface” AND ”integration architecture”” löytyi 47 hakutulosta, joista hyväksyttiin kolme. Kaikki hyväksytyt artikkelit luettiin kokonaisuudessaan, eikä niistä yksikään sisältänyt määritelmää kumppani-integraatioarkkitehtuurille tai sitä vastaavalle arkkitehtuuriselle mallille, joka pyrkii alustaekosysteemin muodostamiseen kumppaneille tarjottavien rajapintojen avulla.

Kumppanirajapinnan määritelmää etsittiin hakulausekkeella ”partner api”, jolla löytyi viisi hakutulosta, jotka kaikki hylättiin. Hakulausekkeella ”partner application programming interface” löytyi yksi hakutulos, joka hylättiin. Kumppanirajapinnan määritelmä löytyi myöhemmän aineiston analysoinnin ohessa Hussain et al. vuoden 2020 artikkelista *Enterprise API Security and GDPR Compliance: Design and Implementation Perspective.*

Parhaan tietoni mukaan kumppani-integraatioarkkitehtuurin käsite määritellään tässä tutkielmassa ensimmäistä kertaa tieteellisessä asiayhteydessä, koska siitä ei löytynyt mainintaakaan kirjallisuuskatsauksen yhteydessä. Kumppani-integraatioarkkitehtuurin määritelmää ei löytynyt edes verkkojulkaisuista Google-haulla hakulausekkeilla ”partner integration architecture” tai ”partner api integration architecture”, joten sen määrittelyssä nojauduttiin toimeksiantajan johtavan ohjelmistoarkkitehdin antamaan kuvaukseen käsitteen sisällöstä.

Tietokantahakuja tehtiin lukuisilla hakusanayhdistelmillä kokeilumielessä, ennen kuin päädyttiin eniten relevantteja tuloksia tuottaneisiin hakulausekkeisiin ””rest api” AND ”security threat””, ””restful api” AND ”security threat””, ””rest api” AND vulnerability” ja ”api security”. Hakulausekekohtaiset tiedot aineistohaun tuloksista löytyvät taulukosta 1 sivulta 18. Hyväksyttäviä aineistoja löytyi kaikkiaan 23 kappaletta, joista 9 oli duplikaatteja. Kaksi aineistoa hylättiin, koska ne eivät olleetkaan saatavilla Full-Text muodossa. Kirjallisuuskatsauksen analyysiin valittiin jäljelle jääneet 12 aineistoa.

Kirjallisuuskatsauksen analyysiin valittujen aineistojen analysointi paljasti, etteivät ne esitelleet tietoturvauhkia ja niiden puolustautumiskeinoja riittävällä tarkkuudella. Näin ollen lisäaineistoa haettiin luvussa 3.2.1 esitellyistä tietokannoista kohdennetuilla hakulausekkeilla kuten ””MitM attack” AND prevention”.

TAULUKKO 1 Aineistohaun tulokset

Hakulauseke	Tuloksia yhteensä	Lähempään tarkasteluun valitut	Hylätyt	Hyväksytyt
”rest api” AND ”security threat”	130	5	0	5
”restful api” AND ”security threat”	46	3	2	1
”rest api” AND vulnerability	365	8	4	4
”api security”	117	7	5	2

3.2.3 Aineiston analyysimenetelmä

Aineiston analyysimenetelmäksi valikoitui sisällönanalyysi, koska tutkimuskysymyksiin vastaaminen edellytti erilaisten tietoturvauhkien ja niiden suojautumistapojen erittelyä tutkimusaineistosta. Sisällönanalyysi on systemaattinen aineiston analyysimenetelmä, jonka avulla voidaan järjestellä ja kuvata tutkimusaineistoa (Karjalainen & Vesalo, 2015). Aineiston analysoiminen alkoi valittujen 12 aineiston lukemisella, jonka aikana aineistoista koodattiin katkelmia erilliseen tiedostoon. Koodaamisella tarkoitetaan aineiston katkelmien yhdistelyä ja erottelua valittujen ominaisuuksien perusteella (Eskola & Suoranta, 2008). Aineistojen läpikäynnin jälkeen samanlaisia löytöjä yhdisteltiin taulukkoon, josta tutkimuksen tulokset ovat helposti nähtävissä.

3.3 Kirjallisuuskatsauksen tulokset

Kirjallisuuskatsauksessa havaittiin, että käytännössä kaikki verkkorajapintojen tietoturvallisuutta käsittelevä aineisto keskittyi REST-standardin mukaisiin verkkorajapintoihin. Näin ollen tutkimuksessa sivuutettiin vain SOAP-standardia koskevat tietoturvauhat, koska niiden otanta olisi jäänyt liian pieneksi. Kirjallisuuskatsauksessa havaittujen tietoturvauhkien lukumäärät on esitetty taulukossa 2 sivulla 19.

TAULUKKO 2 Kirjallisuuskatsauksessa havaitut tietoturvauhat

Valtuutus- ja todennusongelmat	DDoS	Sisäiset uhat	MitM	Brute force	Supply chain
7	4	4	3	2	1

Yleisimpiä kirjallisuuskatsauksessa havaittuja tietoturvauhkia olivat erilaiset valtuutus- ja todennusongelmat (engl. *authorization and authentication issues*), jotka mainittiin seitsemässä artikkelissa. Neljässä artikkelissa otettiin esille organisaation sisäiset uhat, joita ovat esimerkiksi vakoilu ja phishing-huijauksiin lankeaminen. Myös perinteiset hyökkäysmuodot, eli hajautettu palvelunestohyökkäys (DDoS), Man-in-the-Middle-, Brute force- ja Supply chain -hyökkäykset laskettiin keskeisiksi uhiksi. Listatut tietoturvauhat esitellään tarkemmin lähdeaineistoon pohjautuen luvussa 4.

Tunnistettujen tietoturvauhkien suojautumiskeinoja etsittiin kirjallisuuskatsauksessa hyödynnettyjen artikkelien lisäksi kohdennetuilla hakulausekkeilla luvussa 3.2.1 eriteltyihin tietokantoihin, koska joissain analysoiduissa artikkeleissa käsiteltiin tietoturvauhilta suojautumista hyvin pinnallisella tasolla. Suojautumiskeinot on esitelty tarkemmin luvussa 5.

4 TUNNISTETUT TIETOTURVAUHAT

Tässä luvussa esitellään kumppani-integraatioarkkitehtuurin kannalta oleelliset tietoturvauhat, jotka tunnistettiin kirjallisuuskatsausta laadittaessa. Ensimmäisessä alaluvussa määritellään kumppani-integraatioarkkitehtuurin kyberhyökkäyksille altis hyökkäyspinta-ala. Toisessa alaluvussa käsitellään tietomurtoja yleisesti, ja luvun alaluvuissa tutustutaan tarkemmin valtuutus- ja todennusongelmiin, muihin sisäisiin uhkiin sekä palvelunesto-, Man-in-the-Middle-, Brute force- että Supply chain -hyökkäyksiin.

4.1 Kyberhyökkäyksille altis hyökkäyspinta-ala

Organisaation toimintaympäristön kyberhyökkäyksille alttiita osia kutsutaan hyökkäyspinta-alaksi. Hyökkäyspinta-alan laajuus riippuu kyseessä olevan järjestelmän koosta, avoimuudesta ja omaksuttujen tietoturvakäytäntöjen kattavuudesta ja laadusta. Hyökkäyspinta-ala muodostuu kaikista kohteista, jotka mahdollistavat ulkopuolisille pääsyn järjestelmään. (Manadhata & Wing, 2011). Kumppani-integraatioarkkitehtuurin hyökkäyspinta-ala muodostuu elementeistä, jotka mahdollistavat tietovarojen säilyttämisen, käyttämisen ja siirtämisen. Tietovarat säilötään joko yrityksen sisäiseen tietovarantoon tai kolmannen osapuolen tarjoamaan tallennustilaan, kuten esimerkiksi pilvipalveluun. Tietovarojen käsittelyyn käytetään ohjelmistoja, jotka ovat sekä kumppanirajapintojen ylläpitäjän (myöhemmin *pääasiallinen toimija*) että sidosryhmien kehittämiä. Tietovaroja välitetään eri järjestelmien välillä kumppanirajapintojen kautta.

Sekä pääasiallisen toimijan että kumppanirajapintojen hyödyntäjien toimet voivat altistaa myös muiden sidosryhmien tietovarat kyberhyökkäyksille. Kumppanirajapinnat avaavat ulkopuolisille sidosryhmille pääsyn pääasiallisen toimijan tarjoamiin resursseihin, kuten esimerkiksi järjestelmien toimintoihin ja tietovaroihin. Pääasiallisen toimijan vastuulla on huolehtia tarjoamansa teknologisen infrastruktuurin tietoturvallisuudesta. Muiden sidosryhmien vastuulla on vastaavasti huolehtia omien, kumppanirajapintojen varaan kehitettyjen jär-

jestelmiensä tietoturvallisuudesta. Kumppanirajapintoihin pääsyn mahdollistavien pääsytunnusten (engl. *access token*) turvallinen käsittely on kunkin sidosryhmän vastuulla.

Kumppani-integraatioarkkitehtuurin liiketoiminnallinen idea jatkuvasti kasvavasta alustaekosysteemistä, jossa sidosryhmät kehittävät yhä useampia sovelluksia toimimaan kumppanirajapintojen varaan, on tietoturvan kannalta ongelmallinen. Uudet sovellukset kasvattavat hyökkäyspinta-alan määrää, samoin kuin uudet sidosryhmät, joille tarjotaan pääsy kumppanirajapintoihin.

4.2 Tietoturvauhat

Yritysten väliseen yhteistyöhön laadittujen rajapintojen merkittävimpiä tietoturvaongelmia ovat tietomurrot, eli tilanteet, joissa tiedon luottamuksellisuus, eheys tai saatavuus murtuu (Munsch & Munsch, 2020; Whitman & Mattord, 2017). Rajapintojen välillä liikkuva tieto saattaa sisältää arkaluontoisia henkilötietoja, kuten esimerkiksi luottokorttitietoja, terveystietoja tai sosiaaliturvatunnuksia, joita voidaan hyödyntää luottokorttipetoksiin, identiteettivarkauksiin tai muihin vastaaviin rikoksiin. Tietomurrot muodostavat merkittäviä liiketoimintariskejä organisaatioille. Tietovarojen vuotaminen ulkopuolisille voi johtaa vakavimmillaan vastuuorganisaation toiminnan loppumiseen, kuten näimme psykoterapiakeskus Vastaamon tietomurron tapauksessa vuonna 2021. Näin ollen sekä rajapintojen ylläpitäjällä että niitä hyödyntävillä liiketoimintakumppaneilla on merkittävä vastuu rajapintojen kautta kulkevan tiedon luottamuksellisuuden säilyttämisestä. (Munsch & Munsch, 2020).

Tietoturvauhat jaotellaan sisäisiin ja ulkoisiin uhkiin. Sisäiset uhat tarkoittavat organisaation sisäisistä, valtuutetuista henkilöistä muodostuvia uhkia. Kumppani-integraatioarkkitehtuurin kannalta merkittävimmän tietoturvauhan muodostavat erilaiset valtuutus- ja todennusongelmat (Beer & Hassan, 2018; Mateus-Coelho, Cruz-Cunha & Ferreira, 2021; Kornienko et al., 2021; Modi, Chourasia & Pandey, 2022; Hussain et al., 2020; Chen, Zhang & Lian, 2021; Idris, Sarif & Winarno, 2022). Muita merkittäviä uhkia muodostavat erilaiset kyberhyökkäykset, kuten myöhemmissä alaluvuissa esiteltävät palvelunesto-, Man-in-the-Middle-, Brute force- ja Supply chain -hyökkäykset.

4.2.1 Valtuutus- ja todennusongelmat

Valtuutus- ja todennusongelmat muodostuvat, kun valtuuttamaton tai todentamaton käyttäjä pääsee käsiksi suojattuihin resursseihin. Erilaiset valtuutusongelmat nousivat maailman yleisimmiksi verkkosovellusten tietoturvaongelmiksi vuonna 2021. (Gupta, Singh & Mohapatra, 2022). Valtuutusongelmien muodostamat tietoturvariskit voivat olla mittavia. Ongelman vakavuudesta riippuen valtuuttamattomat käyttäjät voivat kyetä esimerkiksi varastamaan, korruptoimaan tai tuhoamaan tietovaroja, tai haittaamaan järjestelmien toimintaa muilla tavoin. (Gupta, Singh & Mohapatra, 2022).

Yleisimmät verkkopohjaisten rajapintojen todennusmekanismit pohjautuvat pääsytunnuksiin. Lähtökohtaisesti pääsytunnus muodostetaan vain käyttäjälle, joka on tunnistettu ja valtuutettu onnistuneesti. Mikäli pääsytunnus joutuu ulkopuolisen tahon haltuun, saattaa tämä päästä käsiksi toimintoihin samoilla käyttöoikeuksilla kuin taho, jolle pääsytunnus alun perin myönnettiin. (Munsch & Munsch, 2020).

4.2.2 Sisäiset uhat

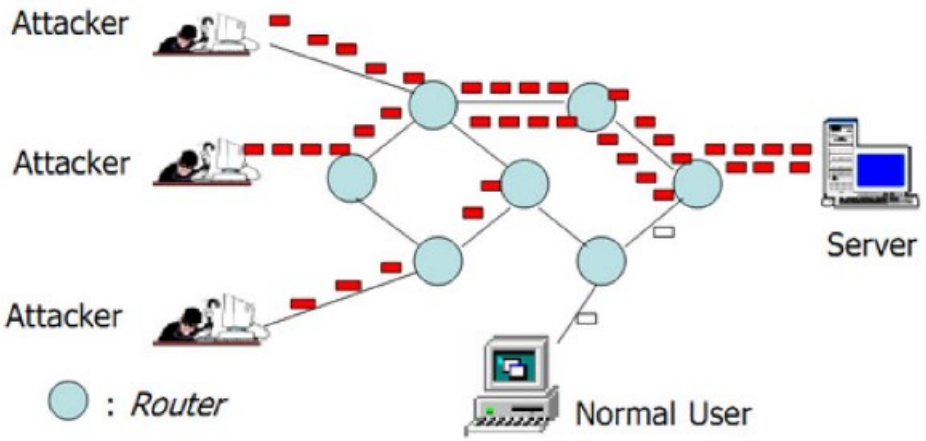
Henkilöstön huolimattomuudesta ja osaamattomuudesta johtuvat virheet ohjelmistojen ja prosessien suunnittelussa ja toteutuksessa voivat muodostaa sisäisiä uhkia. Työntekijä voi vaarantaa organisaation tietoturvan vahingossa, jos tämän tietoturvaosaaminen on puutteellista. Walker-Roberts et al. (2020) mukaan lähes kaikki haittaohjelmien avulla suoritetut tietomurrot suoritettiin sähköpostin liitetiedostona välitetyn haittaohjelman avulla. Kun organisaation sisäiseen verkkoon yhdistetty laite saastuu haittaohjelmalla, saattaa haittaohjelma päästä leviämään myös muihin verkkoon yhdistettyihin laitteisiin. Myös käyttäjätunnusten ja salasanojen vuotaminen hyökkääville tahoille sosiaalisen manipuloinnin hyökkäysten seurauksena on yleistä (Modi, Chourasia & Pandey, 2022). Yleisin sosiaalisen manipuloinnin muoto on tietojenkalasteluhyökkäys (engl. *phishing*) (Syafitri et al., 2022). Sisäisen uhan muodostaa myös tietovarojen varastaminen, jonka motiivina on lähes aina taloudellinen hyöty (Walker-Roberts et al., 2020).

4.2.3 Palvelunestohyökkäykset

Palvelunestohyökkäykseksi kutsutaan kyberhyökkäystä, joka pyrkii kuormittamaan kohteensa suorituskykyä niin voimakkaasti, ettei se kykene suoriutumaan tavallisista tehtävistään. Yleisin esimerkki tästä on palvelintietokoneen kuormittaminen lukemattomilla tekaistuilla palvelupyynnöillä. Palvelunestohyökkäykset ovat usein hajautettuja, jolloin tekaistuja pyyntöjä lähetetään useilta laitteilta samanaikaisesti. (Hoque, Bhattacharyya & Kalita, 2015). Tämä hyökkäystekniikka tunnetaan nimellä DDoS (engl. *Distributed Denial of Service*),

ja siinä hyödynnetään usein laajoja, haittaohjelmien avulla kaapattujen tietokoneiden verkostoja (Hoque, Bhattacharyya & Kalita, 2015). Palvelunestohyökkäyksen toimintaperiaate on havainnollistettu kuviossa 2 sivulla 23.

DDoS vaikuttaa ennen kaikkea tiedon saatavuuteen, ja voi johtaa merkittäviin häiriöihin verkkoinfrastruktuurissa lamauttamalla kaistanleveyden, verkkolaitteiden ja tallennustilan toimintaa. Hajautetun palvelunestohyökkäyksen kustannukset sen kohteelle voivat nousta kymmeniin tuhansiin dollareihin tuntia kohden. (Sahoo et al., 2019). DDoS-hyökkäykset yleistyvät vuosi vuodelta, koska niiden suorittaminen on verrattain yksinkertaista ja halpaa. Voimakkaimpien hyökkäysten takana ovat ammattimaiset rikollisorganisaatiot ja valtiolliset tahot. (Sahoo et al., 2019; Beer & Hassan, 2018).

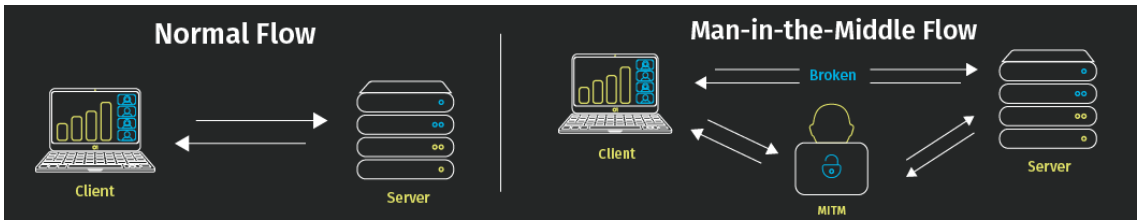


KUVIO 2 Distributed Denial of Service Attack. (Hoque, Bhattacharyya & Kalita, 2015).

4.2.4 Man-in-the-Middle -hyökkäykset

Man-in-the-Middle -hyökkäyksiksi (lyhennettynä MitM) kutsutaan kyberhyökkäystekniikoita, joissa ulkopuolinen taho pyrkii niin sanotusti salakuuntelemaan kohteensa tietoliikennettä kaappaamalla tämän lähettämiä ja vastaanottamia datapaketteja. MitM toimintaperiaate on havainnollistettu kuviossa 3 sivulla 24. Hyökkääjä saattaa kyetä myös muuttamaan datapakettien sisältöä, murtaen näin tiedon eheyden. (Conti, Dragoni & Lesyk, 2016). Spoofing-pohjaisissa MitM-hyökkäyksissä hyökkääjä esiintyy verkon luotettavana tahona käyttäen hyväkseen ARP-, DNS-, DHCP- tai IP-protokollien heikkouksia. (Conti, Dragoni & Lesyk, 2016).

Kumppani-integraatioarkkitehtuurin tapauksessa MitM-hyökkäyksien avulla voidaan pyrkiä salakuuntelemaan, korruptoimaan tai tuhoamaan kumppanirajapintojen kautta kulkevaa tietoliikennettä.



KUVIO 3 Man in the Middle Attack (Veracode.com, 2023).

4.2.5 Brute force -hyökkäykset

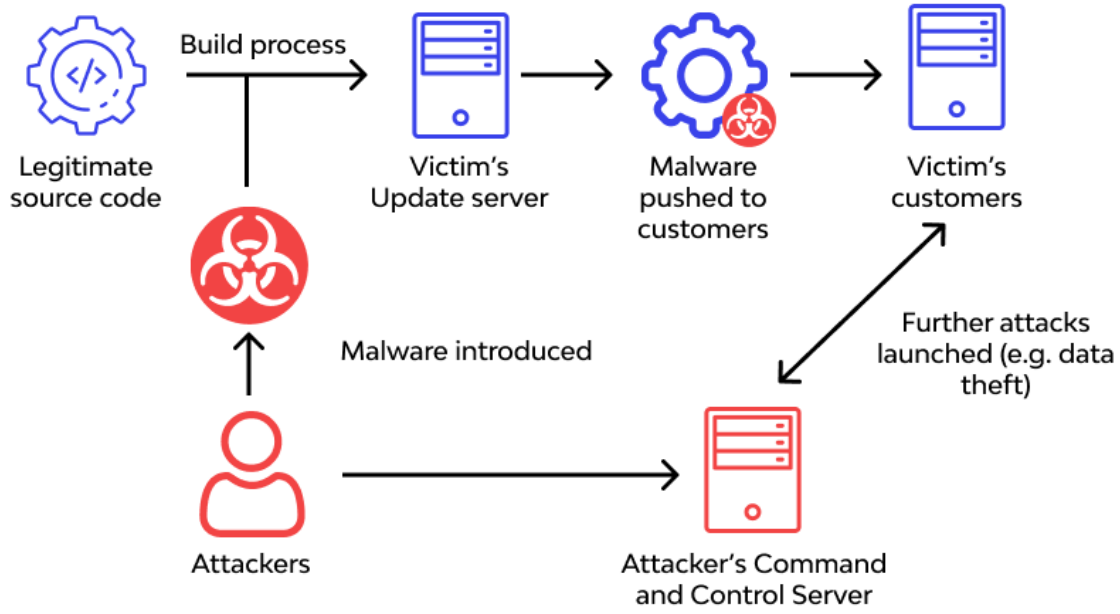
Brute force -hyökkäyksiksi kutsutaan lukuisten samankaltaisten hyökkäysyritysten sarjaa, kuten esimerkiksi salasanan automatisoitua arvailua kokeilemalla tuhansia erilaisia kombinaatioita vuorotellen. Kumppani-integraatioarkkitehtuurin tapauksessa brute force -hyökkäysten avulla voidaan pyrkiä esimerkiksi arvaamaan oikeaa pääsytunnusta, jonka avulla hyökkääjä saisi pääsyn kohdejärjestelmään. (Beer & Hassan, 2018).

4.2.6 Supply chain -hyökkäykset

Ohjelmistokehityksen yleisenä trendinä on ollut jo useamman vuosikymmenen ajan valmiin ohjelmakoodin lainaaminen ulkopuolisista lähteistä. Liiketoiminnan näkökulmasta katsottuna on kustannustehokkaampaa hyödyntää muiden jo valmiiksi laatimia ratkaisuja kuin kehittää jokaista toiminnallisuutta organisaation sisäisesti. Tämän kustannustehokkuuden vastapainona valmiin ohjelmakoodin hyödyntäminen lisää ohjelmistojen kyberhyökkäyksille alttiin hyökkäyspinta-alan määrää merkittävästi. (Crosignani, Macchiavelli & Silva, 2022).

Muiden laatima ohjelmakoodi viittaa joko avoimen lähdekoodin komponentteihin, kuten esimerkiksi lähdekoodikirjastoihin, tai maksullisina palveluina tarjottaviin palveluihin, kuten esimerkiksi rajapintoihin. Organisaation ohjelmistossa hyödynnettäviä, kolmansien osapuolten laatimia ohjelmistokomponentteja kutsutaan yleisesti riippuvuuksiksi (engl. *dependency*), koska ne laativat yksipuolisen riippuvuussuhteen niiden hyödyntäjän ja julkaisijan välille. Riippuvuudet ovat tietoturvan kannalta ongelmallisia, koska niitä hyödyntävästä organisaatiosta käsin on todella vaikeaa havaita riippuvuuksien päivityksien mukanaan mahdollisesti tuomia haavoittuvuuksia. (Wang et al., 2022; Crosignani, Macchiavelli & Silva, 2022). Organisaation järjestelmissä saattaa olla kymmeniä tuhansia yksittäisiä riippuvuuksia, joiden manuaalinen monitoroiminen on henkilöstöresurssien näkökulmasta mahdotonta. Tietoturvan kannalta heikosti suunniteltu ohjelmistoympäristö ei myöskään sisällä toimintoja, jotka päivittäisivät riippuvuudet automaattisesti niiden julkaisijoiden julkaistessa uusia päivityksiä. Tällöin järjestelmiin voi jäädä merkittäviäkin tietoturvahaavoittuvuuksia, jos ne hyödyntävät vanhoja avoimen lähdekoodin komponentteja, joista on sittemmin löydetty vakavia tietoturvahaavoittuvuuksia (Munsch & Munsch, 2020). Riippuvuuden sisältämän tietoturvahaavoittuvuuden hyödyntämistä tietomurtoon tähtäävään kyberhyökkäykseen kutsutaan supply chain -

hyökkäykseksi. Lähihistoriasta löytyy esimerkkejä kyberhyökkäyksistä, jotka on toteutettu tunkeutumalla ensin ohjelmistoyrityksen järjestelmään, josta käsin on levitetty hyökkäyksen mahdollistavaa ohjelmakoodia yrityksen asiakkaille piilottamalla se yrityksen tuotteen päivityksen yhteyteen (Crosignani, Macchiavelli & Silva, 2022). Supply chain hyökkäyksen toimintaperiaate on havainnollistettu kuviossa 4 sivulla 25.



KUVIO 4 How a Supply Chain Attack works (OpenDataScience.com, 2022).

5 TUNNISTETUILTA TIETOTURVAUHILTA SUOJAUTUMINEN

Tässä luvussa esitellään parhaat käytännöt, joiden avulla kirjallisuuskatsauksessa tunnistetuilta tietoturvauhilta kyetään suojautumaan. Ensimmäisessä alaluvussa määritellään suojautumisen parhaat käytännöt. Toisessa alaluvussa tarkastellaan tietomurroilta suojautumista kumppani-integraatioarkkitehtuurin toimintaympäristössä. Toisen alaluvun alaluvuissa esitellään parhaat käytännöt valtuutusongelmien ennaltaehkäisemiseen ja sisäisiltä uhilta sekä palvelunesto-, Man-in-the-Middle-, Brute force- että Supply chain -hyökkäyksiltä suojautumiseen.

5.1 Suojautumisen parhaat käytännöt

Tietoturvauhilta suojautuminen on haastavaa, koska uusia uhkia ja haavoittuvuuksia ilmenee jatkuvasti eri puolilla maailmaa. Erilaiset parhaiden käytäntöjen viitekehykset (engl. *best practice framework*) helpottavat organisaatioiden ajan tasalla pysymistä ja ehkäisevät lukuisten samankaltaisten ratkaisujen kehittämistä ongelmiin, jotka on jo ratkaistu jossain toisaalla. Parhailla käytännöillä tarkoitetaan toimia, joiden on tutkimusten ja käytännön kokemuksen perusteella todettu tuottavan optimaalisia lopputuloksia, ja jotka soveltuvat laaja-alaiseen käyttöönottoon. (Alhogail, 2021). Tunnetuin esimerkki parhaita käytäntöjä yhteen kokoavasta listauksesta on OWASP Top 10 (Beer & Hassan, 2018; Munsch & Munsch, 2020; Mateus-Coelho, Cruz-Cunha & Ferreita, 2021; Kornienko et al., 2021; Idris, Syarif & Winarno, 2022).

Tietoturvauhkiin varautumisen keskiössä on uhkien tunnistaminen. Tietoturvallisuus on otettava huomioon jo järjestelmiä suunniteltaessa, sen sijaan että sitä alettaisiin miettimään vasta jälkikäteen. Arkkitehtien ja kehittäjien on oltava tietoisia yleisimmistä tietoturvauhista kyetäkseen tunnistamaan järjestelmien osat, jotka ovat niille alttiita. Von Solms ja Von Solms (2004) havaitsivat, että parhaiden käytäntöjen sivuuttaminen organisaation tietoturvapolitiikkaa ja riskienhallintasuunnitelmaa laadittaessa on yksi merkittävimmistä syistä tietoturvapolitiikan epäonnistumiseen. Järjestelmän saatetaan esimerkiksi todeta ole-

van turvallinen tiettynä ajankohtana, jonka jälkeen sen päivitykset ja testaaminen laiminlyödään, eikä uusia haavoittuvuuksia tunnisteta ennen kuin niitä hyödynnetään onnistuneesti kyberhyökkäyksiin.

5.2 Tietomurtojen ennaltaehkäiseminen

Kumppani-integraatioarkkitehtuurin toimintaympäristön suojaaminen tietomurroilta on haastavaa, koska hyökkäyspinta-alan määrä on mahdollisesti eksponentiaalisesti kasvava. Kumppanirajapintoja hyödyntävien sidosryhmien määrää ei ole rajattu, kuten ei myöskään näiden rajapintojen varaan rakennettavien sovellusten määrää. Pääasiallinen toimija ei kykene edes teoreettisesti valvomaan muiden sidosryhmien sovellusympäristöjen tietoturvallisuutta, vaikka ne muodostavat osan tämän omien järjestelmien hyökkäyspinta-alasta. Tämän takia kumppanirajapintojen ylläpitäjän täytyy laatia jokaisen rajapintoja hyödyntävän tahon kanssa käyttöehtosopimus. Käyttöehtosopimuksessa rajataan tarkkaan mihin tarkoituksiin rajapintoja saa hyödyntää, millaisia tietosuojakäytäntöjä allekirjoittaja sitoutuu noudattamaan, ja millaiset sanktiot allekirjoittaja hyväksyy sopimusehtojen rikkomisesta. (Acker & Kreisberg, 2020).

Organisaation näkökulmasta tietomurtojen ennaltaehkäisemisen parhaisiin käytäntöihin lukeutuu kattavan tietoturvapolitiikan (engl. *information security policy*) laatiminen, jossa eritellään organisaation sisäiset roolit ja vastuut tietovarojen suojelussa. Henkilöstö toimii todennäköisemmin tietoturvapolitiikan mukaisesti, jos kokee sen rikkomisesta aiheutuvan merkittävää haittaa itselleen, kuten esimerkiksi työpaikan menetys tai merkittävä rikemaksu. Välinpitämättömyys tietoturvapolitiikan noudattamiseen johtaa esimerkiksi heikkojen salasanojen käyttämiseen. (Chen et al., 2018). Kumppanirajapintojen varaan rakentuvan alustaekosysteemin teknologisen infrastruktuurin ylläpitäjällä on suurin vastuu ekosysteemin kokonaistietoturvallisuudesta. Ylläpitäjän omien järjestelmien ja rajapintojen suojausten on oltava tarpeeksi kattavalla tasolla, jotta vain valtuutetut tahot pääsevät käsiksi tietovaroihin. Tietovarojen on myös oltava sidosryhmien saavutettavissa jatkuvasti, mikä edellyttää varautumista esimerkiksi palvelunestohyökkäyksiin.

Tietosuojapolitiikan ja sen sisältämien prosessien keskiössä tulee olla uusien uhkien tunnistaminen. Tämä edellyttää järjestelmien ja rajapintojen säännöllistä testaamista haavoittuvuuksien varalta. Lisäksi organisaatiossa on muodostettava raportointikäytännöt, jotka velvoittavat henkilöstön jäsenet raportoimaan havaitsemansa tietoturvaongelmat vastuuhenkilöille. (Alhogail, 2021).

5.2.1 Valtuutus- ja todennusongelmien ennaltaehkäiseminen

Valtuutus- ja todennusongelmia ilmenee, kun valtuuttamattomat tai todentamattomat käyttäjät pääsevät käsiksi resursseihin, joihin heille ei ole myönnetty oikeuksia. Kumppani-integraatioarkkitehtuurin rajapintaympäristössä tulisi olla käytössä kattava identiteetin- ja pääsynhallinnan valvontaohjelmisto, joka tarjoaa vain todennetuille tahoille pääsyn ympäristöön, ja tarkistaa tahojen valtuutukset ennen kuin myöntää näille oikeuden suojattuihin resursseihin.

Valtuutusongelmien ennaltaehkäisemisen parhaat käytännöt sisältävät periaatteen pienimmistä käyttöoikeuksista (engl. *principle of least privilege*) noudattamisen, eli käyttöoikeuksien myöntämisen vain, kun käyttäjä niitä tarvitsee. Tällöin valtuutusongelmien ilmeneminen on huomattavasti epätodennäköisempää, koska lähtökohtaisesti käyttäjillä on käyttöoikeudet vain niihin resursseihin, joihin järjestelmävalvoja on myöntänyt heille käyttöoikeudet. Pienimpien käyttöoikeuksien periaatetta tulisi soveltaa jokaisessa sovelluksessa ja rajapinnassa. (Mateus-Coelho, Cruz-Cunha & Ferreita, 2021; Walker-Roberts et al., 2020).

Alan parhaisiin käytäntöihin lukeutuu pääsytunnuspohjainen todentautuminen ja valtuutus, johon yleisimmin käytetyt menetelmät ovat JSON Web Token, OpenID Connect, OAuth 2.0 ja API Gateway (de Almeida & Canedo, 2022). HTTP-pyynnöt tulisi päästää rajapintojen päätepisteisiin vain, jos ne sisältävät voimassa olevan pääsytunnuksen. Pääsytunnuksia tulee muodostaa vain tunnistetuille tahoille, ja niiden on oltava tarpeeksi pitkiä ja monimutkaisia, jotta hyökkääjät eivät kykene arvaamaan niitä brute force -hyökkäyksillä. Pääsytunnuksen tulee myös umpeutua lyhyen ajan sisällä, jotta mahdollisesti vuotaneiden pääsytunnusten muodostamat tietoturvauhat olisivat mahdollisimman lyhytkestoisia.

Hussain et al. (2020) argumentoivat, että rajapintayhdyskäytävän (engl. *API gateway*) sisällyttäminen rajapintaympäristöön lisää sen turvallisuutta. Tällöin kaikki liikenne rajapintoihin kulkee yhdyskäytävän kautta, joka hallinnoi pääsytunnuksien tarkistuksia ja liikenteen suodatusta (engl. *client throttling*), mikä suojaa rajapintoja hajautetuilta palvelunesto- ja brute force -hyökkäyksiltä. Beerin ja Hassanin (2018) mukaan pääsynvalvontaohjelmiston tulisi kirjata lokitiedostoon kaikki kirjautumistapahtumat, ja hälyttää järjestelmävalvojia havaitessaan anomalioita, kuten esimerkiksi lukuisia epäonnistuneita kirjautumisyrityksiä samasta kohteesta. Hussain et al. (2020) mukaan rajapintayhdyskäytävä mahdollistaa myös automatisoidun lokien monitoroinnin.

5.2.2 Sisäisiltä uhilta suojautuminen

Organisaation henkilöstön puutteellinen tietoturvaosaaminen muodostaa suuren osan sisäisistä uhista. Henkilöstön informoiminen erilaisista sosiaalisen manipuloinnin hyökkäyksistä on tehokas tapa näiden uhkien ennaltaehkäisemiseen. (Syafitri et al., 2022).

Toisen merkittävän uhan tietoturvalle muodostaa henkilöstön välipitämättömyys tietoturvapolitiikan noudattamista kohtaan. Aurigemma ja Mattson (2017) havaitsivat, että henkilöstö noudattaa tietoturvapolitiikkaa todennäköisemmin, mikäli he kokevat sekä kiinnijäämisen riskin että rikkeestä seuraavien sanktioiden olevan merkittäviä.

Organisaation henkilöstön suorittamia tietomurtoja voi ennaltaehkäistä noudattamalla kaikkien järjestelmien suunnittelussa pienimpien käyttöoikeuksien periaatetta, jolloin taloudellisilla motiiveilla varustettu vakooja ei pääse käsiksi kuin työnsä kannalta relevantteihin tietovaroihin (Walker-Roberts et al., 2020).

5.2.3 Palvelunesto- ja brute force -hyökkäyksiltä suojautuminen

Palvelunestohyökkäyksiltä suojautuminen edellyttää dataliikenteen määrän monitorointia, jotta anomaliat ja mahdolliset palvelunestohyökkäykset kyetään tunnistamaan. Luvussa 4.2.1 esitelty rajapintayhdyskäytävä mahdollistaa tahokohtaisen kaistansäätelyn (engl. *rate limiting*), jonka avulla rajapintoja liikaa kuormittavien tahojen hyväksyttyjen pyyntöjen määrää kyetään rajoittamaan. Lisäksi rajapintayhdyskäytävä mahdollistaa todentamattomien ja valtuuttamattomien pyyntöjen suoran hylkäämisen ennen kuin ne pääsevät rajapinnan päätepisteisiin saakka. Yhdessä oikeaoppisen pääsytunnuspohjaisen todentamis- ja valtuutusjärjestelmän kanssa rajapintayhdyskäytävä vaikeuttaa hajautettujen palvelunesto- ja brute force -hyökkäysten suorittamista merkittävästi. (Hussain et al., 2020).

Palvelunestohyökkäyksiltä suojautumista edesauttaa myös rajapintaympäristön muodostaminen suorituskyvyltään skaalautuvalle alustalle, joka kykenee takaamaan tietovarojen saatavuuden myös nopeasti nousseen kuormituksen alaisena.

5.2.4 Man-in-the-Middle -hyökkäyksiltä suojautuminen

Conti, Dragoni ja Lesyk (2016) laativat laajan kirjallisuuskatsauksen erilaisista MitM-hyökkäystekniikoista ja parhaista käytännöistä, joiden avulla niitä voidaan ennaltaehkäistä. ARP-spoofing-hyökkäysten havaitsemiseen on kehitetty ARP-Guard, ARPDefender, Arpwatch ja Snort nimiset ohjelmat. DNS-spoofing-hyökkäyksiä voi ennaltaehkäistä Anax ja Cache Poisoning Detection System (CPDS) ratkaisujen avulla. DHCP-spoofing-hyökkäysten torjunnassa käytetään yleisesti DHCP snooping -ominaisuutta, joka muodostaa palomuurimaisen rakenteen tuntemattomien tahojen ja luotettujen DHCP-palvelinten väliin. IP-spoofing-hyökkäysten torjunnassa käytetään yleisimmin IPSec-protokollakehystä ja Ingress filtering -menetelmää. (Conti, Dragoni & Lesyk, 2016).

Beer ja Hassan (2018) suosittelevat HTTP-pyyntöjen ja vastausten sisältöjen kryptaamista keinoksi MitM-hyökkäyksiltä suojautumiseen. Tällöin onnistunutkaan MitM-hyökkäys ei johtaisi tietovarojen vuotamiseen, sillä hyökkääjä

saisi haltuunsa vain salattua tekstiä. Conti, Dragoni ja Lesyk (2016) suosittelevat kaiken dataliikenteen kryptaamista.

5.2.5 Supply chain -hyökkäyksiltä suojautuminen

Supply chain -hyökkäyksiltä suojautuminen on haastavaa, koska hyökkäykset suoritetaan lähtökohtaisesti luotettujen ohjelmistojulkaisijoiden tuotteiden päivitysten kautta. Ohjelmistojen ulkoisten riippuvuuksien tulisi koostua aina mahdollisimman luotettavien ja suurten toimijoiden tuotteista, koska suurempi käyttäjäkunta havaitsee mahdolliset supply chain -hyökkäykset todennäköisesti nopeiten. Organisaation ohjelmistojen riippuvuussuhteiden tulisi olla tiedossa ja ne tulisi päivittää automaattisesti. (National Institute of Standards and Technology, 2021). Maksullisten ohjelmistojen tapauksessa organisaation tulisi myös perehtyä ohjelmiston tarjoajan turvallisuuskäytäntöihin kyetäkseen määrittelemään heidän tuotteensa luotettavuuden ja turvallisuuden tason. Onnistuneisiin supply chain -hyökkäyksiin voi varautua laatimalla etukäteen varautumissuunnitelmia ja säilyttämällä aiemmat, turvallisiksi todetut versiot ohjelmistoista, jotta ne voidaan palauttaa käyttöön tarvittaessa. (National Institute of Standards and Technology, 2021).

6 TULOKSET

Tässä luvussa käydään läpi tutkielman tulokset ja arvioidaan niiden tuottamaa arvoa sekä toimeksiantajalle että tietotekniikan ja kyberturvallisuuden tieteenaloille. Ensimmäisessä alaluvussa käsitellään kumppani-integraatioarkkitehtuurin määritelmää. Toisessa alaluvussa käydään läpi kirjallisuuskatsauksen tulokset. Kolmannessa alaluvussa analysoidaan tutkimuksen reliabiliteettia ja validiteettia. Neljännessä alaluvussa arvioidaan kuinka tutkielma vastasi tutkimuskysymyksiin ja millaista lisäarvoa se tuotti tietotekniikan ja kyberturvallisuuden tieteenaloille, sekä lopuksi esitetään jatkotutkimusaiheita.

6.1 Kumppani-integraatioarkkitehtuurin määritelmä

Tutkimuksen ensimmäinen merkittävä tulos muodostui jo ennen kuin tutkimuskysymyksiin edes alettiin etsiä vastauksia kirjallisuuskatsauksen avulla. Aihealueen aiempaan tutkimuskirjallisuuteen perehtyminen osoitti, ettei kumppani-integraatioarkkitehtuurin käsitettä ollut määritelty aiemmin. Tutkielma tuotti uutta tietoa tietotekniikan tieteenalalle muodostamalla kumppani-integraatioarkkitehtuurin käsitteelle määritelmän aiemman tutkimustiedon pohjalta.

6.2 Kirjallisuuskatsauksen tulokset

Kirjallisuuskatsauksessa tunnistettiin kuusi REST-standardin mukaisille verkkorajapinnoille keskeistä tietoturvauhkaa: valtuutus- ja todennusongelmat, sisäiset uhat sekä palvelunesto-, Man-in-the-Middle-, Brute force- ja Supply chain -hyökkäykset. Aineistoa analysoitaessa tunnistettiin myös parhaita käytäntöjä, joiden avulla havaituilta tietoturvauhilta kyetään suojautumaan. Tietoturvauhat ja niiden suojautumiskeinot on esitelty aiemman tutkimuskirjallisuuden pohjal-

ta tarkkuudella, joka vastaa molempiin tutkimuksen motiivina toimineisiin tutkimuskysymyksiin.

6.3 Tutkimuksen luotettavuus

Hyvän tieteellisen käytännön mukainen tutkimus on tehty huolellisesti, rehellisesti ja tarkkuutta osoittaen tutkimuksen tuloksia esitettäessä ja arvioitaessa. Lisäksi tutkimus on suunniteltu, toteutettu ja raportoitu yksityiskohtaisesti. Tutkimuksessa käytettyjen tutkimus- ja analysointimenetelmien tulee olla tieteellisen tutkimuksen kriteerien mukaisia. (Karjalainen & Vesalo, 2015). Tämän tutkielman teossa on noudatettu edellä kuvattua tieteellistä käytäntöä.

Tutkimusten luotettavuutta arvioidaan reliabiliteetin ja validiteetin perusteella. Reliabiliteetti tarkoittaa tutkimuksen toistettavuutta siten, että saadut tulokset eivät ole sattumanvaraisia, vaan toisiaan vastaavia tutkijasta riippumatta. Validiteetti kuvaa tutkimusmenetelmän kykyä mitata haluttua ilmiötä. (Karjalainen & Vesalo, 2015). Validiteetti jaetaan sisäiseen ja ulkoiseen validiteettiin. Sisäinen validiteetti kuvaa tutkimuksessa käytettyjen käsitteiden ja teorioiden oikeellisuutta. Ulkoisella validiteetilla viitataan tutkimuksen yleistettävyyteen. (Karjalainen & Vesalo, 2015).

Tutkimuksen reliabiliteettia parantaa se, että tutkimuksen toteutus ja tulokset on kuvattu selkeästi ja tarkasti. Reliabiliteettia taasen heikentää se, että kyseessä on laadullinen tutkimus, jota oli laatimassa vain yksi tutkija. Tutkijan subjektiiviset näkemykset aineistojen relevanttiudesta saattavat heikentää tutkimuksen toistettavuutta. Tutkimuksessa käytetyt käsitteet on määritelty aiemman tutkimuskirjallisuuden pohjalta, mikä vahvistaa tutkimuksen sisäistä validiteettia. Ulkoista validiteettia heikentää se, että aineistoa haettaessa suoritetut hakukonehaut kohdistettiin vain kolmeen tietokantaan, ja vain suomen sekä englannin kielistä aineistoa hyväksyttiin analysoitavaksi. Tutkimuksen tulokset olisivat sitä paremmin yleistettävissä, mitä laajempi tutkimusaineiston otanta olisi. Myös käytetyllä aineistonhakumenetelmällä olisi saatu huomattavasti laajempi otanta lähdeaineistoa käyttämällä hakulausekkeita, jotka tuottivat useita tuhansia osumia, kuten esimerkiksi ”api AND security”. Näin suurten aineistomäärien läpikäyminen ei kuitenkaan ollut tarkoituksenmukaista ottaen huomioon tutkimuksen aikarajoituksen ja käytettävissä olleiden henkilöresurssien määrän.

Tutkimuskysymykset laadittiin yhteistyössä toimeksiantajan edustajan kanssa. Vastausten löytäminen oli haastavaa, koska kumppani-integraatioarkkitehtuurista tai sitä vastaavasta arkkitehtuurisesta mallista ei ole parhaan tietoni mukaan laadittu tieteellisiä artikkeleita. Koska toimeksiantajan edustaja rajasi käyttöliittymätason tietoturvahaavoittuvuudet tutkimuksen laajuuden ulkopuolelle, päätyi tutkimus keskittymään pitkälti verkkorajapintojen tietoturvauhkiin. Yleisimmät verkkorajapintastandardit ovat SOAP (Simple Object Access Protocol) ja REST. Koska lähdeaineistossa keskityttiin laajalti vain REST-rajapintoihin, rajattiin pelkästään SOAP-rajapintoja koskevat tietotur-

vauhat tutkimuksen ulkopuolelle. Tästä syystä tutkimuksen tulokset eivät ole yleistettävissä kaikkiin verkkorajapintoihin, vaan lähinnä REST-standardin mukaisiin.

Hyväksyttävälle aineistolle laadittiin tarkat hyväksyntäkriteerit ennen aineistohakujen tekoa, ja niitä noudatettiin tarkasti aineistoa etsittäessä. Kirjallisuuskatsauksen lähdeaineistoa voidaan pitää luotettavana, koska se koostuu Julkaisufoorumin hyväksymissä julkaisuissa julkaistuista, vertaisarvioiduista artikkeleista, jotka ovat julkisesti saatavilla.

Aineiston analysoinnissa käytetty analyysimenetelmä on esitelty, ja keskeisistä tutkimustuloksista on laadittu taulukko, josta ne näkyvät selkeästi ja ymmärrettävästi. Tuloksia arvioitaessa on huomioitava tutkijan aiempi kokemattomuus sisällönanalyysistä, mikä saattaa heikentää tutkimuksen toistettavuutta.

6.4 Pohdinta

Tutkimus onnistui vastaamaan sille asetettuihin tutkimuskysymyksiin, tuottaen näin käytännön hyötyä tutkielman toimeksiantajalle. Tutkimuksen tulokset saavutettiin hyvän tieteellisen käytännön mukaisella tutkimusprosessilla, joka on toistettavissa. Uutta tieteellistä tietoa muodostui sekä tietotekniikan että kyberturvallisuuden tieteenaloille.

Aiemmissa tutkimuksissa oli koottu yhteen REST-standardin mukaisten verkkorajapintojen turvallisuusuhkia (Beer & Hassan, 2018; Chatterjee & Prinz, 2022; Munsch & Munsch, 2020; Kornienko et al., 2021; Modi, Chourasia & Pandey, 2022; Hussain et al., 2020) sekä REST-verkkorajapintoja hyödyntävien Microservice-arkkitehtuurien keskeisiä tietoturvauhkia (Chen, Zhang & Lian, 2021; Mateus-Coelho, Cruz-Cunha & Ferreira, 2021) suojautumiskeinoineen, jotka ovat relevantteja myös kumppani-integraatioarkkitehtuurin kannalta, kun kumppanirajapinnat toteutetaan REST-standardin mukaisesti. Erityisesti kumppani-integraatioarkkitehtuurille ominaisia tietoturvauhkia saattoi jäädä tutkimuksen tulosten ulkopuolelle, koska aiheesta ei ollut kirjoitettu aiempia tutkimuksia. Chen, Zhang ja Lian (2021) totesivat osuvasti, että informaatioteknisen yritysmaailman havainnot ja innovaatiot kulkevat usein akateemisen kirjallisuuden edellä, kuten tässäkin tapauksessa. Tämä tutkielma toimii pioneerina kumppani-integraatioarkkitehtuurin osalta akateemisessa kirjallisuudessa.

Jotta aihealueesta kyettäisiin muodostamaan entistäkin tarkempi kokonaiskuva, tulisi jatkotutkimuksissa kuvata tapaustutkimuksen keinoin kumppani-integraatioarkkitehtuurille ominainen tekninen infrastruktuuri, jota tässä tutkielmassa käsiteltiin vain hyvin pinnallisella tasolla. Kyberturvallisuuden tieteenala hyötyisi tutkimuksesta, joka kokoaisi yhteen kumppani-integraatioarkkitehtuurille ominaiseen tekniseen infrastruktuuriin kohdistuneita tietomurtoja, joiden pohjalta kyettäisiin arvioimaan ovatko OWASP Top 10 listauksen tietoturvauhat keskeisimpiä myös tällaisen toimintaympäristön tapauksessa.

LÄHTEET

Acker, A., & Kreisberg, A. (2020). ”Social media data archives in an API-driven world”. *Archival science, 20*(2), 105-123. https://doi.org/10.1007/s10502-019-09325-9

Alhogail, A. (2021). ”Enhancing information security best practices sharing in virtual knowledge communities”. *VINE journal of information and knowledge management systems, 51*(4), 550-572. https://doi.org/10.1108/VJIKMS-01-2020-0009

Alsene, E. (1999). ”The computer integration of the enterprise”. *IEEE transactions on engineering management, 46*(1), 26-35. https://doi.org/10.1109/17.740033

Andersson, A., Hedström, K., & Karlsson, F. (2022). ”Standardizing information security – a structurational analysis”. *Information & management, 59*(3), 103623. https://doi.org/10.1016/j.im.2022.103623

Aurigemma, S., & Mattson, T. (2017). ”Deterrence and punishment experience impacts on ISP compliance attitudes”. *Information and computer security, 25*(4), 421-436. https://doi.org/10.1108/ICS-11-2016-0089

Beer, M. I., & Hassan, M. F. (2018). ”Adaptive security architecture for protecting RESTful web services in enterprise computing environment.” *Service oriented computing and applications, 12*(2), 111-121. https://doi.org/10.1007/s11761-017-0221-1

Bigelov, S. (2023). ”What are the types of APIs and their differences?” https://www.techtarget.com/searchapparchitecture/tip/What-are-the-types-of-APIs-and-their-differences

Chatterjee, A., & Prinz, A. (2022). ”Applying Spring Security Framework with KeyCloak-Based OAuth2 to Protect Microservice Architecture APIs: A Case Study”. *Sensors (Basel, Switzerland), 22*(5), 1703. https://doi.org/10.3390/s22051703

Ceccagnoli, M., Forman, C., Huang, P., & Wu, D. J. (2012). ”Cocreation of Value in a Platform Ecosystem! The Case of Enterprise Software.” *MIS quarterly, 36*(1), 263-290. https://doi.org/10.2307/41410417

Chen, D., Doumeingts, G., & Vernadat, F. (2008). “Architectures for enterprise integration and interoperability: Past, present and future”. *Computers in industry, 59*(7), 647-659. https://doi.org/10.1016/j.compind.2007.12.016

Chen, F., Zhang, L., & Lian, X. (2021). ”A systematic gray literature review: The technologies and concerns of microservice application programming interfaces”. *Software, practice & experience, 51*(7), 1483-1508. https://doi.org/10.1002/spe.2967

Chen, X., Wu, D., Chen, L., & Teng, J. K. (2018). ”Sanction severity and employees’ information security policy compliance: Investigating mediating, moderating, and control variables”. *Information & management, 55*(8), 1049-1060. https://doi.org/10.1016/j.im.2018.05.011

Conti, M., Dragoni, N., & Lesyk, V. (2016). ”A Survey of Man In The Middle Attacks”. *IEEE Communications surveys and tutorials, 18*(3), 2027-2051. https://doi.org/10.1109/COMST.2016.2548426

Crosignani, M., Macchiavelli, M., & Silva, A. F. (2022). ”Pirates without borders: The propagation of cyberattacks through firms’ supply chains”. *Journal of financial economics, 147*(2), 432-448. https://doi.org/10.1016/j.jfineco.2022.12.002

de Almeida, M. G., & Canedo, E. D. (2022). ”Authentication and Authorization in Microservices Architecture: A Systematic Literature Review”. *Applied sciences, 12*(6), 3023. https://doi.org/10.3390/app12063023

DOAJ – Directory of Open Access Journals. (2023). https://doaj.org/about/

Ehsan, A., Abuhaliqa, M. A. M. E., Catal, C., & Mishra, D. (2022). ”RESTful API Testing Methodologies: Rationale, Challenges, and Solution Directions”. *Applied sciences, 12*(9), 4369. https://doi.org/10.3390/app12094369

Eskola, Jari & Juha Suoranta. (2008). ”Johdatus laadulliseen tutkimukseen” (8. p.). Tampere: Vastapaino.

Gholami, M. F., Daneshgar, F., Beydoun, G., & Rabhi, F. (2017). ”Challenges in migrating legacy software systems to the cloud — an empirical study”. *Information systems (Oxford), 67*, 100-113. https://doi.org/10.1016/j.is.2017.03.008

Gupta, C., Singh, R. K., & Mohapatra, A. K. (2022). ”An Approach for Verification of Secure Access Control Using Security Pattern”. *Wireless communications and mobile computing, 2022*, 1-11. https://doi.org/10.1155/2022/1657627

Heshmatisafa, S., & Seppänen, M. (2023). “Exploring API-driven business models: Lessons learned from Amadeus's digital transformation”. *Digital Business, 3*(1), 100055. https://doi.org/10.1016/j.digbus.2023.100055

Hoque, N., Bhattacharyya, D. K., & Kalita, J. K. (2015). ”Botnet in DDoS Attacks: Trends and Challenges”. *IEEE Communications surveys and tutorials, 17*(4), 2242-2270. https://doi.org/10.1109/COMST.2015.2457491

Hussain, F., Hussain, R., Noye, B., & Sharieh, S. (2020). “Enterprise API Security and GDPR Compliance: Design and Implementation Perspective”. *IT professional, 22*(5), 81-89. https://doi.org/10.1109/MITP.2020.2973852

Idris, M., Syarif, I., & Winarno, I. (2022). ”Web Application Security Education Platform Based on OWASP API Security Project”. *Emitter : International Journal of Engineering Technology, 10*(2), 246-261. https://doi.org/10.24003/emitter.v10i2.705

Julkaisufoorumi. (2023). https://www.julkaisufoorumi.fi/fi

Kaila, U. (2018). ”Information Security Best Practices: First Steps for Startups and SMEs”. *Technology innovation management review, 8*(11), 32-42. https://doi.org/10.22215/timreview/1198

Karjalainen, S. & Vesalo, M. (2015). ”Esimiestyön merkitys organisaatioiden johtamisessa muutosprosesseissa: Integratiivinen kirjallisuuskatsaus”. *Jyväskylän ammattikorkeakoulu*.

Kornienko, D. V., Mishina, S. V., Shcherbatykh, S. V., & Melnikov, M. O. (2021). ”Principles of securing RESTful API web services developed with python frameworks”. *Journal of physics. Conference series, 2094*(3), 32016. https://doi.org/10.1088/1742-6596/2094/3/032016

Kuusisalmi, M. (2019). “Alustaekosysteemin muodostumista ja toimintaa tukevien olosuhteiden rakentaminen”. *Tampereen yliopisto*.

LibGuides. (2023). https://proquest.libguides.com/pqc/content

Lindman, J., Horkoff, J., Hammouda, I., & Knauss, E. (2020). ”Emerging Perspectives of Application Programming Interface Strategy: A Framework to Respond to Business Concerns”. *IEEE software, 37*(2), 52-59. https://doi.org/10.1109/MS.2018.2875964

Manadhata, P. K., & Wing, J. M. (2011). ”An Attack Surface Metric”. *IEEE transactions on software engineering, 37*(3), 371-386. https://doi.org/10.1109/TSE.2010.60

Mateus-Coelho, N., Cruz-Cunha, M., & Ferreira, L. G. (2021). ”Security in Microservices Architectures”. *Procedia computer science, 181*, 1225-1236. https://doi.org/10.1016/j.procs.2021.01.320

Modi, B., Chourasia, U., & Pandey, R. (2022). ”Design and implementation of RESTFUL API based model for vulnerability detection and mitigation”.

IOP conference series. Materials Science and Engineering, 1228(1), 12010. https://doi.org/10.1088/1757-899X/1228/1/012010

Munsch, A., & Munsch, P. (2020). "The Future of API Security: The Adoption of APIs for Digital Communications and the Implications for Cyber Security Vulnerabilities". *Journal of international technology and information management, 29*(3), 25-45.

National Institute of Standards and Technology. (2021). "Defending Against Software Supply Chain Attacks". https://www.cisa.gov/sites/default/files/publications/defending_against_software_supply_chain_attacks_508_1.pdf

Niiniluoto, I. (1996). "Informaatio, tieto ja yhteiskunta: Filosofinen käsiteanalyysi". (5. täyd. p.). *Edita*.

OpenAPIHub. (2022). "3 Major Types of API – Public API, Private API & Partner API". https://blog.openapihub.com/en-us/3-major-types-of-api-public-api-private-api-partner-api/

OpenDataScience. (2022). "3 Ways to Protect Your Code from Software Supply Chain Attacks". https://opendatascience.com/3-ways-to-protect-your-code-from-software-supply-chain-attacks/

Rocha, R. (2020). "7 Layers of API Architecture Maturity." https://www.apiscene.io/lifecycle/7-layers-of-api-architecture-maturity/

Sahoo, K. S., Panda, S. K., Sahoo, S., Sahoo, B., & Dash, R. (2019). "Toward secure software-defined networks against distributed denial of service attack". *The Journal of supercomputing, 75*(8), 4829-4874. https://doi.org/10.1007/s11227-019-02767-z

Salminen, A. (2011). "Mikä kirjallisuuskatsaus?: Johdatus kirjallisuuskatsauksen tyyppeihin ja hallintotieteellisiin sovelluksiin." Vaasan yliopisto.

ScienceDirect. (2023). https://www.sciencedirect.com/

Suomalainen asiasanasto- ja ontologiapalvelu Finto. (2018). "Tieto". https://finto.fi/tt/fi/page/t117

Syafitri, W., Shukur, Z., Mokhtar, U. A., Sulaiman, R., & Ibrahim, M. A. (2022). "Social Engineering Attacks Prevention: A Systematic Literature Review". *IEEE access, 10*, 39325-39343. https://doi.org/10.1109/ACCESS.2022.3162594

von Solms, R., & van Niekerk, J. (2013). "From information security to cyber security". *Computers & security, 38*, 97-102. https://doi.org/10.1016/j.cose.2013.04.004

von Solms, B., & von Solms, R. (2004). ”The 10 deadly sins of information security management”. *Computers & security, 23*(5), 371-376. https://doi.org/10.1016/j.cose.2004.05.002

Torraco, R. J. (2016). ”Writing Integrative Literature Reviews: Using the Past and Present to Explore the Future”. *Human Resource Development Review, 15*(4), 404-428. https://doi.org/10.1177/1534484316671606

Veracode. (2023). ”Man in the Middle (MITM) Attack”. https://www.veracode.com/security/man-middle-attack

Walker-Roberts, S., Hammoudeh, M., Aldabbas, O., Aydin, M., & Dehghantanha, A. (2020). ”Threats on the horizon: Understanding security threats in the era of cyber-physical systems”. *The Journal of supercomputing, 76*(4), 2643-2664. https://doi.org/10.1007/s11227-019-03028-9

Wang, W., Dumont, F., Niu, N., & Horton, G. (2022). “Detecting Software Security Vulnerabilities Via Requirements Dependency Analysis”. *IEEE transactions on software engineering, 48*(5), 1665-1675. https://doi.org/10.1109/TSE.2020.3030745

Whitman, M., & Mattord, H. (2017). “Principles of information security (6th ed.)” *CENGAGE Learning Custom Publishing*.