Tietoturvamenetelmät pilvipalveluissa

Oulun Yliopisto
Tieto- ja sähkötekniikan tiedekunta
Tietojenkäsittelytiede
Kandidaatintutkielma
Valtteri Kinnunen
31.3.2022

Tiivistelmä

Tämän kirjallisuuskatsauksen tarkoituksena on esittää käyttäjälle pilvipalvelutyypit, pilvipalvelutasot, pilvipalveluihin liittyviä käytäntöjä, pilvipalveluiden tietoturvavaatimukset, kolme eri hyökkäystekniikkaa ja ratkaisuja näihin hyökkäyksiin. Pilviteknologia itsessään ei ole mikään varsinaisen uusi käsite, mutta viimeisen viidentoista vuoden aikana niiden käyttö on yleistynyt huomattavasti. Pilvipalveluiden räjähdysmäinen suosion kasvu on pakottanut yritykset ajatelemaan IT-ratkaisujaan pilvipalvelut huomioon ottaen.

Tutkielman toinen luku käytetään tutkimusmenetelmien esittelyyn, jonka jälkeen luvussa kolme käydään läpi pilvipalvelut yleisesti. Aluksi esittelen eri pilvipalvelutyypit, joita on kolmea erilaista, julkinen, yksityinen ja hybridipilvi. Pilvipalvelutyyppien esittämisen jälkeen kerron pilvipalveluiden kolmesta eri pääluokasta, jotka ovat ohjelmistotaso (Software as a Service, SaaS), ohjelmistoalustataso (Platform as a Servicce, PaaS) ja infrastruktuuritaso (Infrastructure as a Service, IaaS). Luvussa neljä esitellään pilvipalveluiden tietoturvavaatimukset sekä yleisimmät tietoturvaongelmat, joita ovat SOAP-viestit, faktoroidut käyttöjärjestelmät ja virtuaalikoneen introspektio.

Tietoturvaongelmien jälkeen luvussa 5 esittelen pilvipalveluihin kohdistuvat yleisimmät hyökkäysmetodit joita ovat wrapping-hyökkäys, haittaohjelmainjektio ja tulvahyökkäys. Lisäksi lopussa esitellään näille kolmelle hyökkäystyypille joitakin mahdollisia ratkaisuja.

Sisällysluettelo

1. Johdanto

Tämä työ perustuu jo aikaisemmin JTT-kurssille palautettuun materiaaliin. Pilviteknologiaa voidaan pitää melko uutena konseptina, jonka voidaan todeta muuttaneen suurta osaa informaatioteknologian kentästä. Pilvipalveluiden yleistyminen on aiheuttanut yritysten IT-toiminnassa huomattavia muutoksia, kun IT-ratkaisuja kehitetään. Tässä tutkielmassa tulen pääasiallisesti käyttämään termiä pilvipalvelu, mutta termi pilvilaskentakin saattaa esiintyä muutamaan kertaan tutkielmassa.

Pilvipalvelulla voidaan esimerkiksi tarkoittaa loppukäyttäjälle tarjottavaa skaalautuvaa IT-infrastruktuuria (Armbrust ym., 2010). Pilvipalveluiden käyttöönotto on nostanut suosiotaan organisaatioissa sen tuomien hyötyjen vuoksi, josta on myös viime vuosina tehty paljon kattavaa tieteellistä tutkimusta. Pilvipalveluiden suuri suosio perustuu nimenomaan tähän vaivattomaan skaalautuvuuteen. Lisäksi suosiota kasvattavat niiden tarjoamat ominaisuudet kuten edellä mainittu resurssien skaalautuvuus ja alhaiset kustannukset verrattuna perinteisiin palvelimiin. Resurssien vaivaton skaalautuvuus ja alhainen kustannushinta lisäävät pilvipalveluiden houkuttelevuutta eri yritysten, organisaatioiden tai muiden yksittäisten loppukäyttäjien näkökulmasta (Mell ja Grance, 2011).

Pilvipalvelun käyttäminen ja ylläpito ei kuitenkaan ole pelkkää ruusuilla tanssimista, sillä yksi pilvipalveluihin kuuluva huonopuoli liittyy käyttäjien yksityisyyteen ja palvelun tietoturvaan. Pilvipalveluissa asiakkaan data lähetetään verkon yli palveluntarjoajalle datavarastoihin säilytettäväksi, jolloin loppukäyttäjän yksityisyys ja tietoturva ovat avainasemassa. Asiakkaan dataa tulisi aina pitää arkaluontoisena, jolloin palveluntarjoajan ensimmäinen prioriteetti tulisi nimenomaan olla asikkaan datan suojaaminen. Myös omistussuhde dataan on herättänyt kysymyksiä pilvilaskennan kentällä. Kenen omistuksessa ja käytettävissä lopulta data on, koska se sijaitsee organisaation ulkopuolisen toimijan hallussa. Myös salakuuntelu ja arkaluontoisen tiedon urkkiminen herättää kysymyksiä pilvipalveluiden käyttöönotossa. (Hayes, 2008.)

Tässä tutkielmassa kerron aluksi mitä pilvipalvelut ovat ja mitä yleisiä käytäntöjä pilvipalveluiden toiminnassa on. Tutkielman pääpainona toimivat luvut kolme, neljä ja viisi, joissa kerron pilvipalveluiden perusteista, pilvipalvelujen tietoturvavaatimuksista, pilvipalvelujen tietoturvaongelmista, yleisimmistä hyökkäysmenetelmistä ja siitä, mitä ratkaisuja näihin hyökkäyksiin voidaan soveltaa.

Tutkimusongelmana tarkastellaan sitä, mitkä asiat vaikuttavat pilvipalveluiden tietoturvallisuuteen, niiden yleiset tietoturvaongelmat ja yleisimmät niihin kohdistuvat hyökkäykset. Tutkimuskysymykset ovat: ”Mitkä ovat pilvipalveluiden yleiset tietoturvaongelmat?” ja ”Mitkä ovat pilvipalveluihin kohdistuvat yleisimmät hyökkäysmenetelmät ja mitä ratkaisuja niihin voidaan soveltaa?”

2. Tutkimusmenetelmät

Tutkimus rajattiin koskemaan pilvipalveluiden tietoturvaongelmia ja niihin kohdistuvia hyökkäyksiä. Kaikkia hyökkäysmenetelmiä ei tutkielmassa kuitenkaan esitellä, vaan nostin esille yleisimmät hyökkäysmenetelmät. Rajasin tutkielman aineistoa julkaisuvuoden perusteella, sillä pilvipalveluiden tietoturvallisuutta parannetaan kokoajan ja uusia hyökkäysmenetelmiä syntyy jatkuvasti. Pyrin pääsääntöisesti pitämään lähteeni vuosien 2010 ja 2021 välillä. En kuitenkaan pitänyt rajauksesta ihan niin tarkasti kiinni, sillä vanhin lähteeni on kuitenkin vuodelta 2006.

Kirjallisuuden ja empiiristen tutkimusten hakeminen toteutettiin pääasiassa Scopuksen (https://www.scopus.com/) avulla. Scopus on tietokanta, joka sisältää viittaustietoja pääosin luonnontieteiden ja lääketieteen alalta. Juurikin tämä luonnontieteisiin keskittyminen on syynä siihen miksi valitsin pääasialliseksi hakukoneeksi nimenomaan Scopuksen. Scopuksesta löytyy pääosin lehtiartikkeleita, joten kirjahakuja tein myös Google Scholarissa (https://scholar.google.com/). Omaa empiiristä tutkimusta ei tässä tutkielmassa tehdä, vaan tarkoituksena on käydä kirjallisuuskatsauksen avulla läpi pilvipalveluiden tietoturvaa ja niihin liittyviä ongelmia.

Scopuksessa hakusanoina käytin pääosin termejä information security, cloud, cloud computing, empirical study, qualitative research, quantitative research. Näiden termien avulla pyrin muodostamaan Scopuksen käytön kannalta hyödyllisiä hakulausekkeita. Ensimmäinen hakulausekkeeni (”information security” AND ”cloud computing”) tuotti yli 1000 hakutulosta, mikä on tämän tuktimuksen kannalta vähän turhan liikaa. Hakua sai kuitenkin nopeasti rajattua kohdistamalla haun vain artikkeleiden otsikoihin jolloin samalla lausekkeella hakutulokset tippuivat muutamaan kymmeneen. Käyttämällä hakulausekkeessa ”cloud computing” -termin sijasta pelkästään termia ”cloud” nosti hakutuloksien määrää noin sadalla. Hakulauseke (”information security” AND ”cloud”) onkin juuri se millä hain suurimman osan lähteistäni.

Kuitenkin tutkielmaan olisi hyvä sisällyttää myös empiiristä tutkimustietoa, jota löydetään lisäämällä hakulausekkeeseen termit ”empirical study”, ”quantitative research” ja ”qualitative search”. Kuitenkin kyseisellä hakulausekkeella löytyi 0 tulosta, joten päädyin muutamaan lausekkeen muotoon (”information security” AND ”cloud” AND ”empirical study”) ja muutin vielä haun rajauksen otsikoiden lisäksi myös ottamaan huomioon avainsanat sekä abstraktin. Tällä hakulausekkeella ja rajauksella hakutuloksia löytyi 41 kappaletta.

Aineiston analysointi tapahtui kätevästi lukemalla hakutuloksien abstraktin. Usein abstraktista sai nopeasti selville sopisiko kyseinen lähde tutkielmaani käytettäväksi. Jos totesin abstraktin lukemisen jälkeen aineiston sopivaksi siirryin tarkastelemaan tutkimusten lopputuloksia ja käytännön implikaatioita. Tutkimusta tehdessä selvisi myös yksi Scopuksen heikkous, nimittäin se, että Scopus ei itsessään sisällä kokonaisia lähdetiedostoja, näin ollen jouduin etsimään alkuperäisen lähdetiedoston usein Google Scholarin avulla. Tämä taas vastaavasti rajasi aineistoa, sillä jokaiseen lähteeseen ei aina löytynyt vastaavaa tiedostoa.

3. Pilvipalvelut yleisesti

Tässä luvussa esitellään eri pilvipalvelutyypit, niiden kolme pääluokkaa sekä pilvipalveluiden toimintaan liittyviä käytäntöjä. Ensiksi olisi kuitenkin hyvä määritellä pilvipalvelu käsitteenä yleisesti. Yksi hyvä määritelmä pilvipalvelulle löytyy Mellin ja Grancen (2011) kirjoittamasta artikkelista, joka löytyy hyvin usein viitteenä pilvipalveluita käsittelevästä materiaalista. Heidän mukaansa pilvipalvelu on aina saatavilla oleva, vaivaton ja laajasti skaalautuva internetin välityksellä saatava palvelu, jonka käyttämiseen ei tarvitse käydä sen kummemin palveluntarjoajan kanssa keskustelua. Mell ja Grance (2011) ovat myös artikkelissaan määritelleet pilvipalveluille viisi ominaispiirrettä, jotka ovat on-demand-itsepalvelu, laaja tavoitettavuus, yhteiskäytettävät resurssit, nopea skaalautuvuus ja palvelun mitattavuus.

Pilvipalvelulla itsellään tarkoitetaan sekä internetin välityksellä asiakkaalle toimitettavia applikaatioita, että sitä laitteistoa ja ohjelmistoa, joka mahdollistaa kyseisen toiminnan. Juurikin näitä internetin välityksellä asiakkaille toimittevia applikaatioita on kutsuttu nimellä Software as a Service (SaaS). (Armbrust ym., 2010.) Tässä tapauksessa siis palveluntarjoaja tarjoaa palvelua internetin välityksellä ja tämän palvelun pystyy ostamaan kuka tahansa, eli kyseessä on silloin julkinen pilvi. Silloin kun kyse on yrityksen sisäisistä pilvipalvelimista, joita ei tarjota julkiseen käyttöön on kyseessä yksityinen pilvi. Hybridi pilvi taas on jonkinlainen julkisen ja yksityisen pilven välimuoto. (Armbrust ym., 2010.) Nämä kolme pilvipalvelutyyppiä käydään tarkemmin läpi luvussa 3.1.

Nykypäivänä pilvipalvelut koostuvat kolmesta eri pääluokasta, joita ovat: ohjelmisto palveluna (Software-as-a-Service, SaaS), ohjelmistoalusta palveluna (Platform-as-a-Service, PaaS) ja ohjelmistoinfrastruktuuri palveluna (Infrastructure-as-a-Service, IaaS) (Viega, 2009). Loppukäyttäjän kannalta näiden kolmen pääluokan välillä ei ole juurikaan eroa. Palveluiden turvallisuus kun on täysin riippumaton käyttäjästä, eikä käyttäjä itse pysty siihen mitenkään vaikuttamaan. Pääluokat käydään tarkemmin läpi luvussa 3.2.

Käytännössä siis pilvipalvelut eivät ole mitenkään monimutkainen käsite. Palveluntarjoaja tarjoaa loppukäyttäjälle internetin yli ohjelmiston, sovellusalustan tai ohjelmistoinfrastruktuurin. Käyttäjä itse ei pilvipalveluiden käyttöön tarvitse muuta kuin laitteen, jolla internetin selaaminen on mahdollista sekä internet selaimen. Internetin välityksellä toimitteva palvelu on myös lähes loputtomasti skaalautuva, mikä tuo erityistä joustavuutta asiakkaan tarpeisiin.

3.1 Eri pilvipalvelutyypit

Turvallisen pilvipalvelun tarjoamisen yhteydessä, on mietittävä, minkä tyyppisen pilvipalvelun haluat toimittaa asiakkaalle. Tällä hetkellä vaihtoehtoina on kolme eri tyyppistä pilvipalvelua, jotka ovat julkinen, yksityinen ja hybridi.

Julkisessa pilvipalvelussa käyttäjät pääsevät pilvipalveluun käsiksi käyttämällä ihan perus internetselainta. Julkinen pilvipalvelu yleensä perustuu, pay-per-use –malliin, joka on samankaltainen, kuin esimerkiksi prepaid sähkön kulutuksen mittausjärjestelmä. Kuitenkin tällainen pilvijärjestelmä mukautuu käyttäjän tarpeiden mukaan. Dillon ym. (2010) mukaan julkinen pilvipalvelu onkin johtavassa asemassa nimenomaan

käyttöönotosta puhuttaessa. Julkinen pilvi ei ole yhtä turvallinen, kuin yksityinen tai hybridipilvi, koska sen julkisuuden vuoksi siinä on enemmän haavoittuvaisuuksia, kuin muissa pilvimalleissa (Zhang ym., 2010). Yksi tapa millä julkisen pilven turvallisuutta voidaan parantaa, on asiakkaan ja tarjoajan välinen sopimus, jossa molemmat lupaavat huolehtivan siitä, että pilven turvallisuutta kunnioitetaan ja että, validointia suoritetaan kaikissa heidän järjestelmissään (Ramgovind ym., 2010).

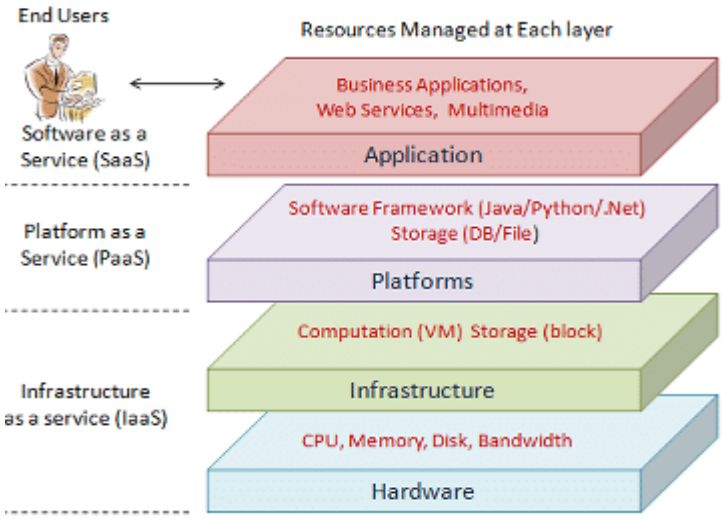
Yksityinen pilvi on pystytetty organisaation omien sisäisten servereiden sisälle. Yksityisen pilven on helpompi mukautua turvallisuusmäärityksiin sekä muihin rajoittaviin sääntöihin (Dillon ym., 2010). Lisäksi se antaa organisaatiolle täydet oikeudet sen käyttämiseen, jolloin organisaatio voi itse päättää miten ja milloin käyttää pilvipalvelua. Yksityisessä pilvessä pilven tarjoajan tarjoamat skaalattavat resurssit ja virtuaaliset sovellukset tuodaan yhteen, joista muodostetaan isompi kokonaisuus, jotka sitten ovat kaikkien pilveä käyttävien saatavilla. Yksityinen pilvi eroaa julkisesta pilvestä siinä, että kaikki pilven resurssit ja sovellukset ovat suoraan kyseisen organisaation hallinnassa. (Dillon ym., 2010.) Pilven hyötykäyttö on myös yksityisessä pilvessä turvallisempaa, kuin julkisessa pilvessä, juurikin tämän sisäisen rakenteen takia. Ainoastaan organisaatiolla ja sen sisällä olevilla valituilla henkilöillä on pääsy pilven kontrolleihin. (Ramgovind ym., 2010.)

Hybridi pilvi tarkoittaa yksityistä pilveä, joka on yhdistetty yhteen tai useampaan ulkoiseen pilvipalveluun (Calheiros ym., 2011). Hybridi pilvi on keskeisesti kontrolloitu, sitä käytetään kuin yhtä pilveä ja sitä suojaa turvallinen verkko. Se tarjoaa IT-ratkaisuja molempien, sekä julkisen, että yksityisen pilven kautta. Hybridipilvi mahdollistaa myös enemmän joustavuutta verrattuna julkiseen tai yksityiseen pilveen. (Zissis & Lekkas 2012.) Hybridi pilvi tarjoaa turvallisempaa datan ja sovellusten kontrollointia ja antaa monelle eri taholle pääsyn informaation internetin ylitse (Ramgovind ym., 2010). Zhang ym. (2010) mainitsevat yhtenä hybridipilven negatiivisena puolena sen, että kun hybridipilveä varten valitaan komponentteja, useasti ongelmaksi muodostuu valinnan vaikeus siitä, että minkälaisella jaolla komponentit jaetaan julkisen ja yksityisen pilven välillä. Dillonin ym. (2010) mukaan yritykset ja organisaatiot käyttävät hybridimallia juurikin silloin, kun ne yrittävät parantaa kilpailukykyään ulkoistamalla toisarvoisia liiketoiminnan osia pilveen samalla, kun kontrolli tärkeimmistä aktiviteeteistä säilyy yksityisessä pilvessä.

Zhang ym. (2010) mukaan näiden mallien lisäksi on olemassa vielä viime vuosina suosiotaan kasvattanut virtuaalinen yksityinen pilvi (Virtual Private Cloud, VPC). Tätä pidetään lähinnä vaihtoehtoisena ratkaisuna yksityisen ja julkisen pilven rajoitteisiin. Käytännössä VPC on pilvialusta, joka toimii julkisen pilven päällä. Zhang ym. (2010) mukaan ero normaaliin julkiseen pilveen on siinä, että VPC hyödyntää toiminnassaan virtuaalista yksityistä tietoverkkoa (Virtual Private Network, VPN). Tämä taas tarjoaa käyttäjälle julkista pilveä enemmän tietoturva-asetuksien konfigurointia, kuten esimerkiksi palomuurin turvallisuusasetuksien määrittelyn. Zhang ym. (2010) toteavatkin pilvipalvelutyypin valinnassa tärkeintä olevan yrityksen tai organisaation tulevaisuuden näkymät. Julkinen pilvi pääsee omilleen nimenomaan tiedeyhteisössä, sillä sen kustannustehokkuus on markkinoiden parasta, eikä perinteistä fyysistä palvelinta pystytä skaalaamaan yhtä vaivattomasti eri tiedeyhteisöjen tarpeisiin nähden. Tulevaisuudessa on ennustettu, että hybridimalli tulee nousemaan suositummaksi käyttöönottomalliksi, sillä se tavallaan sisältää julkisen ja yksityisen pilven hyvät puolet ja negatiiviset puolet hybridipilvessä ovat mitättömät. (Zhang ym., 2010.)

3.2 Pilvipalveluiden pääluokat

Viegan (2009) mukaan pilvipalvelut voidaan jakaa kolmeen pääluokkaan, jotka ovat ohjelmisto palveluna (Software-as-a-Service, SaaS), ohjelmistoalusta palveluna (Platform-as-a-Service, PaaS) ja ohjelmistoinfrastruktuuri palveluna (Infrastructure-as-a-Service, IaaS). Näiden kolmen tason lisäksi Zhang ym. (2010) nostavat esille vielä laitteistokerroksen, joka oikeastaan mahdollistaa koko pilvipalvelun toiminnan. Laitteistokerroksen tehtävänä on huolehtia ja säädellä pilven fyysisiä resursseja, kuten fyysisiä palvelimia, reitittimiä sekä virta- ja jäähdytyslaitteistoa. Laitteistokerros on yleensä implementoitu palvelinkeskuksissa, jotka sisältävät tuhansia servereitä.



Kuvio 1 Pilvipalveluiden yleisrakenne (Zhang ym., 2010).

Pilvipalvelun alin kerros, eli IaaS, tarjoaa asiakkaille prosessointivoimaa, säilytystilaa ja erilaisia verkkoratkaisuja internetin välityksellä (Al Morsy ym., 2010). Tämä toimintamalli perustuu hyvin pitkälti virtualisaatioteknologiaan. Al Morsy ym. (2010) mainitsevat, että yleisin IaaS-palveluita tarjoava taho on Amazon EC2. IaaS-pilviluokassa on yksi pilvikerros, jossa pilvipalvelun tarjoajan omistamat resurssit ovat vain maksavien asiakkaiden käytettävissä, pay-per-use -periaatteen mukaan. Tämä käytäntö vähentää huomattavasti suuren aloitussijoituksen tarvetta tietokonekomponentteihin, kuten servereihin, verkkolaitteisiin tai prosessorivoimaan. Lisäksi IaaS tarjoaa taloudellisia ja funktionaalisia joustavuuksia, joita ei löydetä perinteisistä sisäisistä datakeskuksista, koska komponenttitarvetta pystytään lennosta muuttamaan asiakkaan tarpeiden mukaan. (Ramgovind ym., 2010.)

PaaS toimii pitkälti samalla tavalla kuin IaaS, kuitenkin PaaS tarjoaa vielä yhden tason enemmän ”vuokrattua” funktionaalisuutta (Mell ja Grance, 2011). PaaS on se taso, jonka päälle asiakas pystyy omien resurssien avullaan rakentamaan ohjelmistonsa. Asiakkaalla ei kuitenkaan ole pääsyä kontrolloimaan ohjelmistoalustan alla piilevää infrastruktuuria (Mell ja Grance, 2011). Asiakkaat, jotka käyttävät PaaS:ia, ovat siirtäneet vielä enemmän kuluja pääomasijoituksista suoraan toimintakustannuksiin, kuin IaaS:ia käyttävät, mutta heidän tulisi kuitenkin ottaa edelleen huomioon, että PaaS:in lisätaso tuo mukanaan lock-in -mahdollisuuden. PaaS-palveluiden kulmakivenä toimii virtuaalitietokoneet, jolloin tietoturvanäkökulmasta kysymykseksi nousee virtuaalitietokoneiden turvallisuus. Tämän takia koko järjestelmän eheyttä tulee valvoa jatkuvasti ja tietoturvakäytännöissä tulisi olla tarkat määräykset sen suhteen, miten liikkuvaa dataa tulisi tarkastaa ja miten sen oikeellisuudesta voidaan olla varmoja. (Ramgovind ym., 2010.)

IaaS:n tapaan myös SaaS toimii virtualisoidun pay-per-use -mallin mukaan, mutta SaaS:ssa tarjotaan suoraan ohjelmistoa asiakkaille, eikä infrastruktuuria tai alustaa niin kuin kahdessa aikaisemmassa tasossa (Mell ja Grance, 2011). Käyttäjä pääsee sovelluksiin käsiksi useimmiten ihan perinteisesti verkkoselaimen avulla. Al Morsyn ym. (2010) mukaan sovelluksen funktionaalisuuttaa ollaan useimmiten rajoitettu ja halutessaan käyttäjä pystyy helposti laajentamaan sitä. SaaS-tarjoajat voivat hostata omissa datakeskuksissaan, ulkoisen tarjoajan kautta tai he voivat itse olla ulkoistettu IaaS-tarjoajalle. IaaS-palveluiden saatavuus on yksi SaaS-palveluiden kulmakivistä. Koska, SaaS-palveluihin päästään käsiksi internetin ylitse selaimella, on verkkoselaimen tietoturvallisuus kriittisessä osassa. Tietoturva-ammattilaisten tulee ottaa huomioon monia suunnitellessaan SaaS-palvelun turvallisuutta. Esimerkiksi WS (Web services), XML (Extendable Markup Language) ja SSL (Secure Socket Layer) turvallisuusominaisuuksia olisi hyvä miettiä. (Ramgovind ym., 2010.)

3.3 Pilvipalveluiden toimintaan liittyviä käytäntöjä

Pilvipalvelun toteutus riippuu sekä pilvipalvelua tarjoavan yrityksen ominaisuuksista, että palvelusopimuksessa sovitusta palvelutasosta. Erilaisten toteutusten ja palvelusopimusten monimuotoisuus tekee vaikeaksi antaa yksiselitteisiä vastauksia pilvipalveluihin liittyvissä kysymyksissä. Esimerkiksi pilvipalveluun tallennetun tiedon maantieteellinen sijainti tai pilvipalveluun tallennetun tiedon elinkaari riippuvat palveluntarjoajasta ja palvelusopimuksesta.

Pääsääntö on, että tiedon omistajuus ja siihen liittyvät oikeudet ovat sillä henkilöllä tai organisaatiolla, joka on tuottanut tiedon alun perin esimerkiksi laatimalla pilveen tallennetun asiakirjan. Omistuksen mahdollinen siirtyminen ja käyttöoikeudet määräytyvät sovellettavan lainsäädännön ja sopimusten perusteella. Palveluntarjoajan kanssa on hyvä sopia palveluehdoista tarkasti ja palvelun ehdot on hyvä lukea huolella. Erityisen tärkeää on huomioida tiedon hallintaan ja käsittelyoikeuteen liittyviä seikkoja.

Viestintävirasto (2014) suosittelee, jos yritys säilyttää pilvessä muun tiedon ohella myös liikesalaisuuksiaan, on varmistettava, etteivät henkilöt, joilla ei ole oikeutta käsitellä näitä tietoja, pääse niihin käsiksi. Käsittelyoikeuksien säilyminen vain organisaatiolla itsellään on hyvä varmistaa sopimuksin. Palvelua käyttöönottavan organisaation voi olla järkevää sisällyttää sopimukseen mahdollinen auditointioikeus sen tarkastamiseen, että yrityksen tietoja käsitellään laaditun palvelusopimuksen mukaisesti. Arkaluonteisen tai salassa pidettävän tiedon käsittelyyn tulee kiinnittää erityistä huomiota. Tällöin arvioitavaksi voi tulla myös se, pitääkö yrityksen ja palveluntarjoajan välillä solmia salassapitoon liittyvä oma salassapitosopimus.

Viestintäviraston (2014) määritelmän mukaan tiedon elinkaari tietojärjestelmissä alkaa siitä, kun tieto luodaan ja päättyy siihen, kun se ja kaikki siitä mahdollisesti tehdyt kopiot tuhotaan. Pilvipalvelussa oleva tieto on joko luotu itse pilvipalvelussa tai se on luotu käyttäjän omalla työasemalla tai tuotu siihen muualta ja ladattu pilveen. Pilvipalvelussa tieto on tallennettuna useassa paikassa samaan aikaan eri järjestelmien muistissa, ulkoisissa massamuisteissa tai tietokannoissa. Samaan aikaan se voi olla myös liikkeellä tietovirrassa. Jos tiedolla on useita käsittelijöitä, on mahdollista, että siitä on useampia kopioita käsittelijöiden paikallisissa tietojärjestelmissä tai heidän muissa tallennusvälineissään. Samoin jos tiedosto lähetetään jollekulle esimerkiksi sähköpostin

välityksellä, siitä syntyy uusi kopio, jolla on oma elinkaarensa, joka riippuu sen vastaanottajan toimista.

Viestintäviraston (2014) mukaan pilvipalveluiden toiminta varmistetaan useimmiten varmuuskopioinnin tai palvelun kahdentamisen avulla. Varmuuskopiointi tarkoittaa yleensä sitä, että palveluiden tietosisältö kopioidaan turvalliseen paikkaan. Palvelun kahdentaminen tarkoittaa koko palvelun replikointia toiseen paikkaan siten, että palvelu on ajan tasalla ja käytettävissä eri paikoissa samanaikaisesti. Pilvipalveluiden varmuuskopiot tai kahdennukset voivat sijaita missä vain maapallolla, missä palveluntarjoajalla tai sen käyttämällä alihankkijalla on oma palvelinkeskus tai muuta siihen vaadittavaa kapasiteettia käytössään. Maantieteellistä hajautusta käytetään palvelun toiminnan varmistamiseksi sekä resurssien kohdentamiseksi.

4. Pilvipalveluiden tietoturvaongelmat

Pilvipalveluiden yhtenä merkittävimpänä ongelmana pidetään nimenomaan niiden tietoturvaa ja tähän liittyen tutkimusta on viime vuosina tehty runsaasti. Käyttäjän näkökulmasta epävarmuutta aiheuttaa juurikin se, että palvleuntarjoajalla on täysin rajaton pääsy käyttäjän dataan. Kyseinen data voi olla hyvinkin arkaluontoista materiaalia, joten sen suojaamisen tulisi olla palveluntarjoajan ensimmäinen prioriteetti. Kuitenkin pilvessä toimiva verkkopalvelu on korkeintaan yhtä luotettava kuin itse pilvipalvelukin missä se toimii, joten tämäkin aiheuttaa huolia liittyen palvelun turvallisuuteen.

Tässä luvussa esittelen aluksi pilvipalveluiden yleiset tietoturvavaatimukset, jonka jälkeen kuvailen pilvipalveluiden yleisimmät tietoturvaongelmat, jotka ovat SOAP-viestit, faktoroidut käyttöjärjestelmät ja virtuaalikoneen introspektio.

4.1 Pilvipalveluiden tietoturvavaatimukset

Tässä alaluvussa käydään läpi yleisellä tasolla vaatimukset, jotka jokaisen palvelun tulisi saavuttaa tietoturvalliselta kannalta. Monet kyseisistä vaatimuksista koskevat etenkin IaaS(Infrastructure as a Service) -palveluntarjoajaa ja näiden palveluiden käyttäjää. Zissisin ja Lekkasin (2012) mukaan eräs hyvinkin tavanomainen tapaus pilvipalveluissa on se, että asiakas haluaa siirtää fyysisen laitteiston ylläpidon pilveen palveluntarjoajalle. Asiakkaan näkökulmasta katsottuna tämä onkin yksi pilvipalveluiden tarjoamista eduista ja kuitenkin samalla myös yksi suurimmista huolenaiheista. Kun asiakas siirtää laitteiston ylläpidon jonkin toisen tahon hoidettavaksi, menettää asiakas kaiken laitteiston fyysisen hallinnan ja samalla siirtyy myös asiakkaan data sekä yksityisyys palveluntarjoajan harteille (Cachin ym., 2009). Pilvipalveluiden tietoturvan ja niissä esiintyvien riskien ymmärtäminen on tärkeässä osassa, kun pilvipalveluille kehitetään tehokkaita ja tepsiviä ratkaisuja (Takabi ym., 2010). Takabin ym. (2010) toteavat, että vaikka pilvipalvelut tarjoavat asiakkailleen sekä edullisia että joustavia resursseja heti käytettäväksi, tarjoaa niiden uniikki infrastruktuuri käyttäjälleen ihan uudenlaisia tietoturvaongelmia, mitä perinteisistä palvelininfrastruktuureista ei löydy. Perinteiseen tietoturvamalliin verrattuna pilvipalveluiden eriävät ominaisuudet luovat siis poikkeavat vaatimukset tietoturvaan.

Pilvipalveluiden tietoturvan takaaminen ei ole pelkästään palveluntarjoajan vastuulla, vaan asiakkaankin harteille lankeutuu jonkinlainen vastuu, riippuen siitä onko kyseessä SaaS, PaaS vai IaaS -asiakas (Takabi ym., 2010). Takabin ym. (2010) mukaan SaaSissa suurin osa tietoturvavastuusta on palveluntarjoajalla, sillä asiakas on ostanut käytettäväkseen vain ohjelmiston ja tämän ohjelmiston alapuolella toimiva ohjelmistoalusta sekä ohjelmistoinfrastruktuuri on täysin asiakkaan oikeuksien ulkopuolella, eikä asiakas pysty niihin vaikuttamaan millään tavalla. PaaSissa taas asiakas on itse rakentanut ohjelmiston ostamansa alustan päälle, joten kyseisen ohjelman käyttäjien tietoturvan takaaminen on taas asiakkaan harteilla (Takabi ym., 2010). Tietenkin PaaSissa kaikki infrastruktuuri- ja laitteisto -ongelmat ovat palveluntarjoajan harteilla, sillä asiakkaalla ei ole niihin pääsyä. Takabi ym. (2010) mukaan koko infrastruktuurin oltaessa asiakkaan käytössä, eli siis IaaS-palveluissa, makaa lähes koko tietoturvavastuu heidän harteillaan. Palveluntarjoajalla on silti edelleen tarjottava edes jonkinlaista alhaisen tason tietojensuojausmenetelmiä, sillä kaikki asiakkaan lähettämä ja vastaanottama data kulkee internetin välityksellä palveluntarjoajan palvelimiin.

Saatavuus on yksi verkkopalveluiden tärkeimmistä elementeistä, sillä kaikkien verkossa toimivien palveluiden saatavuus on lähes mahdotonta pitää sadassa prosentissa. Saatavuus on myös erittäin tärkeässä roolissa silloin kun pohditaan kannattaisiko minun valita yksityinen, julkinen vai hybridipilvi. (Ramgovind ym., 2010.) Yleensä kuitenkin palveluntarjoajan ja asiakkaan välillä tehdyssä sopimuksessa on määritelty jonkin asteinen saatavuusprosentti, joka on poikkeuksetta määritelty mahdollisimman lähelle sataa prosenttia, yleensä yli 99.9% (Zissis & Lekkas, 2012). Asiakkaan on hyvä itse olla perillä sopimusehdoista, sillä on hyvin mahdollista, että esimerkiksi maksun myöhästyessä palveluntarjoaja sulkee palvelun ja asiakas menettää datansa (Cachin ym., 2009).

Eheyttä voidaan pitää saatavuuden rinnalla yhtä tärkeänä elementtinä pilvipalveluiden käytössä. Ramgovind ym. (2010) käyttävät termiä ACID (atomicity, consistency, isolation and durability) puhuttaessa pilvipalveluiden eheysvaatimuksista. On tärkeää, että data pysyy oikeellisena ja paikkansa pitävänä kun se siirtyy asiakkaalta palveluntarjoajalle (Takabi ym., 2010). Oletusarvoisesti kuitenkin data siirtyy käyttäjältä palveluntarjoajan tietokantoihin aina epäluotettavaa verkkoa pitkin. Siirtovaiheessa yleisimmät ongelmat datan virheellisyyteen ovatkin juuri epäluotettava verkkoyhteys tai erinäiset ohjelmistovirheet. (Cachin ym., 2009.) Lisäksi palveluntarjoajan tulee taata asiakkaalle, ettei kukaan kolmas osapuli pääse oikeudettomasti käsittelemään asiakkaan siirtämää dataa. Yleisimpiä ratkaisuja näihin eheysongelmiin ovat digitaaliset allekirjoitukset ja tiivistealgoritmit (Cachin ym., 2009). Palveluntarjoajan palvelimet ovat myös haavoittuvaisia sekä ulkoa, että organisaation sisältä tuleville hyökkäyksille. Cachin ym. (2009) mainitsevat esimerkkinä Red Hat Linuxin palvelimiin kohdistuneen hyökkäyksen, jossa hyökkääjä pääsi lisäämään lähdekoodiin useita uusia tietoturvareikiä.

Zissisin ja Lekkasin (2012) mukaan luottamuksellisuudella käsitetään, että asiakkaan lähettämää dataa ja tietoja pidetään salassa tallennuksen ja siirron aikana. Luottamuksellisuuden puutetta on pidetty pilvipalveluiden aihealueella yhtenä merkittävimmistä huolenaiheista. Palveluntarjoajan on helppo päästä käsiksi asiakkaan tallentamaan dataan, joten asiakkaan on tärkeää pystyä luottamaan palveluntarjoajaan. Myös kolmansien osapuolten hyökkäykset on otettava huomioon. Asiakkaan pitää pystyä luottamaan palveluntarjoajaan, ettei kolmas osapuoli pääse missään tilanteessa käsiksi asiakkaan tallentamaan dataan (Takabi ym., 2010). Palveluntarjoaja ei kuitenkaan ole yksin täysin vastuussa kolmansien osapuolten urkinnasta. Asiakas voi myös itse omalla toiminnallaan esimerkiksi paljastaa salasanansa, jolloin mahdollisesti syntyvästä tietoturvauhasta on vastuussa asiakas itse, eikä palveluntarjoaja. Cachin ym. (2009) mukaan luottamuksellisuutta on mahdollista parantaa esimerkiksi käyttämällä erilaisia salausmenetelmiä datan tiedonsiirrossa ja varastoinnissa. Näitä salausmenetelmiä kutsutaan kryptografisiksi avaimiksia ja niitä ei tule säilyttää samassa palvelussa, missä käyttäjän data sijaitsee.

Identifikaatio eli tunnistus tarkoittaa, että palveluntarjoajan palveluiden käyttäjiä ja heidän käyttöoikeuksiaan voidaan hallinnoida ja että käyttäjät voidaan tunnistaa. Pääsynvalvontaa yleensä kontrolloidaan käyttäjätunnuksen ja salasanan tai erilaisten kryptografisten avainten avulla. (Ramgovind ym., 2010.) Pääsynvalvonnan kontrollointi pilvipalveluissa on tärkeää, koska käyttäjillä palvelussa voi olla erilaisia käyttöoikeuksia ja myös palvelu voi sisältää eri käyttäjillä henkilökohtaisia tiedostoja ja tietoja (Cachin ym., 2009).

Authorisaatiolla eli hallinnalla tarkoitetaan, että käyttäjän täytyy pystyä hallinnoimaan infrastruktuurin mahdollistamia resursseja reaaliajassa. Käyttäjällä täytyy siis olla oikeus

palvelun tilan ja näkyvyyden kontrollointiin. (Zissis & Lekas, 2012.) Palvelun hallinnointi tapahtuu yleensä jonkinlaisen rajapinnan kautta. Hallinta tarkoittaa käytännössä esimerkiksi sitä, että käyttäjän täytyy pystyä poistamaan tiedostoja palvelimelta tai tarvittaessa ajamaan verkkopalvelu alas. (Cachin ym., 2009.) Ramgovindin ym. (2010) mukaan myös valtuuttaminen on tärkeässä osassa hallinnasta puhuttaessa. Ainoastaan palvelun ylläpitäjällä tulisi olla valtuuttamisoikeudet, jonka kautta pystytään kontrolloimaan sitä, että kuka palveluun pääsee käsiksi.

Kiistämättömyys on myös tärkeää kaikessa verkossa toteutettavassa liiketoiminnassa. Ramgovindin ym. (2010) mukaan kiistämättömyydellä tarkoitetaan sitä, ettei kukaan liiketoimintaan osallistunut osapuoli pysty kiistämään osallistumistaan tai peittelemään sitä. Tämä pystytään takaamaan erilaisten digitaalisten allekirjoitusten, aikaleimojen ja varmistuskuittien avulla (Ramgovind ym., 2010).

4.2 SOAP-viestit

Zunnurhain ja Vrbskyn (2011) mukaan Web service (WS) -teknologia on kaikista yleisin käytetty menetelmä SOA (Service-Oriented Architecture) puolella pilvessä. WS-suojauksen tulisi olla tarpeeksi vahva estääkseen perushyökkäykset monelta eri hyökkääjältä. SOAP on XML-pohjainen viestintäarkkitehtuuri, jolla vaihdetaan koodattuja viestejä (kuten web servicen pyyntö- ja vastausviestit) erilaisten protokollien avulla (esim. HTTP, SMTP, MIME). Tällä hetkellä kaksi yleisintä SOAP-hyökkäyksiä ovat palvelunestohyökkäys (Denial of Service, DoS) ja ”käärehyökkäys” (Wrapping attack). (Zunnurhain & Vrbsky, 2011.) Erityisesti näistä jälkimmäinen on erittäin vaarallinen ja yleinen hyökkäys isoille datakeskuksille, kuten pilvijärjestelmille. Vaikkakin jotkin yritykset ovat onnistuneet pitämään järjestelmänsä suojassa näiltä hyökkäyksiltä, voi jopa Amazonin kokoinen yritys joutua tällaisen hyökkäyksen uhriksi, kuten heidän EC2 (Elastic Compute Cloud) -palvelun tapauksessa kävikin. (Zunnurhain & Vrbsky, 2011.) Seuraavassa kappaleessa voisin esittää esimerkin käyttäen Amazon Web Service (AWS) -teknologiaa ja sen tietoturvaominaisuuksia.

Prosessin alussa, kun henkilö on rekisteröitymässä AWS-palveluun, täytyy hänen syöttää tietoihin ”Self-Signed Certificate” (SSC). SSC on satunnaisesti generoitu RSA AWS:ää varten. Jos henkilöllä ei kuitenkaan ole tällaista, täytyy hänen lähettää nimikirjoituksella varustettu julkisesti määritelty sertifikaatti AWS:lle. (Rahaman ym., 2006.) Tässä tapauksessa AWS toimittaa joitakin komentorivityökaluja, joiden avulla pystytään etsimään virtuaalikonekuvia (Amazon Machine Images, AMI). Nämä kuvat suoritetaan, niitä monitoroidaan ja lopuksi terminoidaan jotkin AMI:t. (Rahaman ym., 2006.) Näitä SOAP-viestejä pystyy jokainen kehittäjä muokkaamaan. SOAP-headeri sisältää kaksi elementtiä, joista toinen on BinarySecurityToken, joka sisältää edellä mainitsemani sertifikaatin. Toinen elementti on TimeStamp, joka nimensä mukaisesti sisältää kyseisen SOAP-viestin luomis- ja loppumispäivämäärän. (Rahaman ym., 2006.) Jos SOAP-viesti välitetään suojaamattoman kanavan läpi, täytyy SOAP body (joka sijaitsee SOAP: Envelopen sisällä) ja SOAP-headerin sisällä oleva TimeStamp allekirjoittaa (Zunnurhain & Vrbsky, 2011). Koska jokaisen kanavan suojauksessa on oletusarvona olemassa SSL/TLS -suojaus, on suojaamattoman kanavan hyökkäykset suurimmassa osassa tapauksia täysin tehottomia.

4.3 Faktoroidut käyttöjärjestelmät ja virtuaalikoneen introspektio

Faktoroidut käyttöjärjestelmät (fos) on suunniteltu hoitamaan erinäköiset haasteet, joita järjestelmissä yleensä esiintyy, kuten pilvipalvelut ja monen ytimen järjestelmät. Fos:it voivat myös tarjota rungon pilvipalvelun tietoturvalle (Zunnurhain & Vrbsky, 2011). Käytännössä kuitenkin on olemassa monia samaan luokkaan kuuluvia järjestelmiä kuin fos. Muun muassa perinteiset mikrokernelit, jaetut käyttöjärjestelmät ja pilvipalveluinfrastruktuurit. Perinteisen palvelinten välisen rinnakkaisuuden sijasta fos-järjestelmä pyrkii jakautumaan ja rinnakkaistumaan yhden palvelimen sisällä mahdollistaen yhden korkean tason funktion. Päämotiivit fosin kehittämiseen olivat skaalautuvuus, elastisuus, virheiden sietokyky ja suurien järjestelmien ohjelmoinnissa ilmenevien vaikeuksien ohittaminen. (Zunnurhain & Vrsbky, 2011.) Isolle järjestelmälle, kuten pilvipalvelulle, fos-pohjainen käyttöjärjestelmä on täydellinen vastaus yllä mainittuihin ongelmiin. Monissa usean ytimen järjestelmässä, käyttöjärjestelmä hallitsee sekä tarkkailee järjestelmän resursseja, että aikatauluttaa tehtävät. Skaalautuvuuden tapauksessa, sovellus muutetaan palveluksi, jonka jälkeen se vielä jalostetaan usemmassa muussa palvelussa, jotta se voidaan jakaa palvelukohtaisten palvelinten välille.

Zunnurhain ja Vrbskyn (2011) mukaan järjestelmän resurssien hallitsemisesta vastaa käyttöjärjestelmä, jonka takia monen ytimen järjestelmien ytimet on dynaamisesti jaoteltu jokaisen tehtävän kohdalla palvelinten välillä. Jaksollisten viestien avulla voidaan monitoroida, toimiiko jokainen palvelin, niin kuin pitäisi. Jos yksi jaksollinen viesti puuttuu, voidaan siitä päätellä, että yhdessä palvelimessa on vika. Toisin kuin aikaisissa pilvi- tai klusterijärjestelmissä, fos tarjoaa yhden järjestelmäkuvan jokaista sovellusta kohden. Tämä tarkoittaa käytännössä sitä, että sovelluksen käyttöliittymä on yhden tietokoneen kautta saatu kuva, mutta samalla käyttöjärjestelmä implementoi tämän saman käyttöliittymän monen eri tietokoneen välillä pilvessä. (Zunnurhain & Vrbsky, 2011.)
Jokaiselle pilven käyttäjälle toimitetaan oma ilmeentymä yhdestä virtuaalikoneesta, esimerkiksi käyttöjärjestelmä tai jokin sovellus. Virtual Machine Introspection (VMI) ehdotettiin virtuaalitietokoneiden monitorointia varten Livewire-ohjelman avulla (Zunnurhain & Vrbsky, 2011). Monitorointikirjasto nimeltään XenAccess on hyvä työkalu monitorointia varten. XenAccess hyödyntää toiminnassaan VMI-tekniikkaa, jonka avulla se pystyy pääsemään käsiksi muistin ja levyn käyttötiloihin halutun käyttöjärjestelmän sisällä. Tällaisen lähtökohdan yhtenä haittapuolena on se, että kohde järjestelmän tulee olla puhdas silloin kun monitorointi aloitetaan. ”Lares” on runko, jonka avulla voidaan hallita luottamattoman käyttäjän virtuaalikoneella pyörivää ohjelmaa, injektoimalla käyttäjän virtuaalikoneeseen suojattuja koukkuja (Payne ym., 2008).

Näiden suojattujen koukkujen asettaminen vaatii virtuaalikoneen muokkaamista lennosta, joka ei välttämättä ole mahdollista jokaisessa käyttöjärjestelmässä. Jokaisessa luettelemastani turvallisuusmenetelmästä löytyy siis joku heikkous, jonka takia jokaisessa pilvipalvelussa tulisi harjoittaa kapsulointia. (Payne ym., 2008.)

5. Pilvipalveluihin kohdistuvat hyökkäykset

Seuraavassa osiossa määrittelen erilaisia pilvipalveluihin kohdistuvia hyökkäysmenetelmiä. Käsittelyssä tulee olemaan Wrapping-hyökkäys, haittaohjelmainjektio ja tulvahyökkäys.

5.1 Wrapping-hyökkäys

Kun pilvipalvelua käyttävä henkilö tekee pyynnön virtuaalikoneelle nettiselaimen kautta, menee pyyntö ensin nettipalvelimelle. Palvelimen sisällä generoidaan SOAP-viesti, joka sisältää pyynnön rakenteen sekä sen minkälaisesta nettiselaimen ja palvelimen välisesti vuorovaikutuksesta on oikein kyse. (Zunnurhain & Vrbsky, 2011.) Viestin ”Body” sisältää kaiken pyyntöön liittyvän tiedon, lisäksi body tarvitsee allekirjoituksen, jotta se menee palvelimella eteenpäin. Wrapping-hyökkäyksessä hyökkääjä tekee hyökkäyksensä ennen kuin SOAP-viesti kerkeää TLS-tasoon kääntämistä varten. Jos SOAP headerin sisälle bodyn rinnalle on ilmestynyt uusi ”Wrapper” elementti ei sen prosessointiin tarvita muuta kuin yksinkertainen validointi. (Rahaman ym., 2006.) Tämän turvallisuusaukon takia hyökkääjä pystyy tekemään käyttäjän lähettämästä SOAP-viestistä kopion ja lähettämään tekemänsä haittaohjelmakopion palvelimelle. Yksinkertaisuudessaan hyökkääjä ottaa oikeutetun käyttäjän lähettämän SOAP-viestin, tekee siitä kopion ja lähettää viestin uuden headerin alaisuudessa palvelimelle. Zunnurhain ja Vrbskyn (2011) mukaan kun viesti pääsee palvelimelle, palvelin tarkastaa viestin ja koska viesti sisältää oikean käyttäjän allekirjoituksen, palvelin päästää viestin järjestelmäänsä, vaikka sen headeri onkin tuntematon. On kuitenkin olemassa keinoja, joilla tällaisen wrapping-hyökkäyksen voimme tunnistaa. Rahaman ym. (2006) löytämä ”inline approach” -keino on yksi hyvistä tavoista tunnistaa ja välttää wrapping-hyökkäykset. Jos hyökkääjä muuttaa viestin rakennetta ja yhtä viestin rakenteista, esimerkiksi Bodya, muokataan, voidaan wrapping-hyökkäys huomata melkeinpä heti ja se pystytään estämään.

5.1.1 Haittaohjelmainjektio

Kun käyttäjän pyyntö suoritetaan pilvijärjestelmässä, on olemassa hyvin suuri todennäköisyys, että käyttäjän käyttämä nettiselain ja pilvipalvelun nettipalvelin vaihtavat keskenään metadataa (Zunnurhain & Vrbsky, 2011). Hyökkääjä pystyy käyttämään tätä metadatan vaihtoon kuluvaa aikaa hyväkseen. Hyökkääjä pystyy tekemään oman instanssin metadatasta tai hyökkääjä pystyy yrittämään injektoimaan jo olemassa olemaan metadataan haitallista koodia. Tässä tapauksessa, joko injektoitu haittaohjelmisto tai koodi esiintyy pilvipalvelussa ihan validin näköisenä palveluna, eikä sitä erota muista palveluista. (Zunnurhain & Vrbsky, 2011.) Jos hyökkääjä on onnistunut hyökkäyksessään, tulee pilvipalvelu kärsimään salakuuntelusta ja lukkiutumista. (Zunnurhain & Vrbsky, 2011). Lukkiutumiset aiheuttavat palvelun oikeille käyttäjille sen, että he joutuvat odottamaan, että haittaohjelmisto palvelu on tehnyt suoritettavat prosessinsa, jonka jälkeen he pystyvät tekemään omia oikeita juttujaan. (Zunnurhain & Vrbsky, 2011.)

5.1.2 Tulvahyökkäys

Jokaisessa pilvipalvelujärjestelmässä, kaikki tietokonepalvelimet työskentelevät omissa palveluissaan niille määritetyllä tavalla, samalla palvelimet kommunikoivat toistensa kesken. Jos jokin palvelimista ylikuormittuu tai on ylittänyt oman suorituskynnyksensä, kyseinen palvelin siirtää joitakin sen suorituksessa olevia tehtäviä muille samanlaisia tehtäviä suorittaville palvelimille tai poikkeustapauksessa kaikista lähimpänä olevalle palvelimelle. (Zunnurhain & Vrbsky, 2011.) Tällainen jakaminen tekee pilvestä itsestään paljon tehokkaamman ja nopeamman vaikeitten pyyntöjen käsittelyssä. Kuitenkin, jos hyökkääjä on saanut itselleen oikeudet lähettää pyyntöjä pilveen, pystyy hän tällä tavalla lähettämään palvelimelle valepyyntöjä, joka kuormittaa palvelinta. Kun palvelin käsittelee pyyntöjä pilvessä, tarkistaa palvelin aina ensimmäisenä, että pyynnön lähettänyt henkilö on todennettu luotettavaksi käyttäjäksi (Zunnurhain & Vrbsky, 2011).

Kun palvelin tarkastaa ei oikeutettuja pyyntöjä, eli hyökkääjän lähettämiä pyyntöjä, käyttää palvelin omaa prosessorivoimaansa, muistiaan ja lisäksi se työllistää palvelun IaaSia suurella mittakaavalla (Zunnurhain & Vrbsky, 2011). Tästä taas seuraa se, että koska palvelin on saavuttanut prosessointikynnyksensä, lähettää se näitä kutsuja suoritettavaksi muille palvelimille. Muut palvelimet sitten taas suorittavat näitä hyökkääjän lähettämiä valepyyntöjä, josta seuraa taas se, että nämä palvelimet taas vuorostaan lähettävät työtaakkaansa muille palvelimille. (Zunnurhain & Vrbsky, 2011.) Tällä tavalla hyökkääjä saa koko pilvipalvelun jumitettua, ihan vain lähettämällä yhdelle palvelimelle valepyyntöjä.

5.2 Wrapping-hyökkäykset ja inline approach

Wrapping-hyökkäyksen tapauksessa jo ennen kuin tällainen hyökkäys on ajankohtainen olisi hyvä miettiä erilaisia tietoturvamenetelmiä SOAP-viesteille. Tässä on oikeastaan kaksi hyvää vaihtoehtoa, jolla wrapping-hyökkäys voidaan välttää. Ensimmäinen on SSC (self-signed signature) ja RSA-avaimen generointi ihan vain jokaista SOAP-viestiä kohden. (Zunnurhain & Vrbsky, 2011.) Toinen tapa on rekisteröidä julkinen sertifikaatti pilvipalvelun tarjoajan kanssa. Itse wrapping-hyökkäyksen välttämiseen suosittelen sitä, että SOAP-viestin turvallisuusheaderi pitää allekirjoittaa, joka kerta kun se lähetetään suojaamattoman kuljetuskanavan kautta (Zunnurhain & Vrbsky, 2011). Kun viesti vastaanotetaan määränpäässä, viestin kelpoisuus tarkistetaan heti ensimmäisenä. Jos viestin TimeStamp ei vastaa TimeStampille määrättyjä parametrejä, voidaan olettaa viestin olevan saastunut.

Kun SOAP-viestejä mietitään WS-suojauksen kannalta, huomaamme, että SOAP-viesteille on mahdollista luoda huomattava määrä SOAP-lisäosia (Rahaman ym., 2006). SOAP-header kun ei koskaan mieti SOAP-viestien rakennetta, joka on nimenomaan wrapping-hyökkäyksessä se reitti mitä kautta hyökkäys tehdään (Rahaman ym., 2006). Rahaman ym. (2006) viittaavat dynaamisiin SOAP-viesteihin käyttämällä termiä SOAP-tili. Rahaman ym. (2006) toteavatkin, että lisäämällä SOAPiin tämän SOAP-tilin sisältämän tiedon, XML-uudelleenkirjoitushyökkäysten havaitseminen tapahtui aikaisin validointiprosessin aikana, kun ennen SOAP-tilin lisäämistä nämä uudelleenkirjoitukset menivät validoinnin läpi. Koko ”inline approachin” ideana on turvata SOAP-viestien eheys lisäämällä niihin SOAP-tilin sisältämä informaatio.

5.3 Haittaohjelmainjektion ratkaisut

Asiakkaan virtuaalikone on luotu ja sitä säilytetään pilvipalvelun kuva-arkistossa (image repository). Näitä sovelluksia pidetään aina korkean tason sovelluksina ja niiden eheyttä pidetään ykkösprioriteettina. Zunnurhain ja Vrbsky (2011) suosittelevatkin, että näiden virtuaalikoneiden turvallisuutta katsottaisiin ihan hardware-tasolta, koska hyökkääjällä on erittäin vaikea päästä hyökkäämään suoraan palvelimen hardwareen. Zunnurhain ja Vrbskyn (2011) mukaan tässä olisikin hyvä käyttää FATiin (File Allocation Table) pohjautuvaa järjestelmäarkkitehtuuria, FATin suoraviivaisen tekniikan takia, sekä siksi, että FATiä tukee lähestulkoon jokainen käyttöjärjestelmä. Tällaisesta FAT-taulusta löytäisimme haluamamme sovelluksen, joka on juuri sillä hetkellä jollakin käyttäjällä päällä. Hypervisorin (Virtuaalikonemonitori) avulla tarjoaja pystyisi pyörittämään montaa eri virtuaalikonetta yhdestä lähdekoneesta. Hypervisori olisi vastuussa tehtävien aikataulutuksesta, mutta ennen kuin se aikatauluttaa tehtäviä, sen tulisi tarkastaa istunnon eheys FAT-taulukosta. (Zunnurhain & Vrbsky, 2011.)

Eräs toinen ratkaisu injektiohyökkäykseen olisi antaa käyttäjän tehdä käyttäjätilin pilveen, jonka palveluntarjoaja kopioisi virtuaalikoneen varaustaulukkoon (Kumar, 2016). Varaustaulukon käytön kautta pystymme varmistamaan, että koodi jota käyttäjä suorittaa on säilyttänyt eheytensä, eikä siihen ole injektoitu haittaohjelmia (Kumar, 2016). Tavallisesti, kun asiakas avaa tilin pilvessä, palveluntarjoaja tekee tästä tilistä kuvan, joka säilytetään kuvavarastossa (Image Repository). Asiakkaan suorittamat ohjelmat siis ajetaan korkean tehokkuuden ja eheyden alaisina. (Kumar, 2016.) Kumarin (2016) mukaan erityisesti koodin ja datan eheyttä tulisi tarkkailla laitteisto-tasolla, sillä hyökkäykset IaaS-tasolla ovat erittäin vaikeita. Kumarin (2016) mukaan myös vaihtoehtoinen tapa varmistaa käyttäjän datan eheys, on tallentaa asiakkaan käyttöjärjestelmätiedot, asiakkaan avatessa käyttäjätilin. Pilvipalveluiden ollessa täysin käyttöjärjestelmäriippuvaisia, pystytään eheys varmistamaan tarkistamalla käyttäjän käyttöjärjestelmätiedot, ennen kuin instanssi avataan pilvessä (Kumar, 2016).

5.4 Tulvahyökkäyksen ratkaisut

Tulvahyökkäyksen estämisessä tärkeää on ajatella jokaista yksittäistä pilvipalvelinta yhtenä isona palvelimena tai joukkona palvelimia (Zunnurhain & Vrbsky, 2011). Jokaiselle palvelinjoukolle tulisi määrittää jokin tietty tehtäväkategoria, jota juurikin se kyseessä oleva palvelinjoukko hoitaisi. Tällä tavalla kaikki joukon palvelimet kommunikoivat keskenään viestien avulla. Kun yksi palvelin ylikuormittuu, voidaan joukkoon lisätä uusi palvelin, joka saa joukon kautta kaikki sen joukon palvelinten tiedot ja tiedon siitä minkälaisia tehtäviä kyseinen palvelinjoukko käsittelee (Kumar, 2016). Tässä tapauksessa ylikuormituksen siirtoviestiin liitettäisiin PID (proportional–integral–derivative controller), PIDit on encryptattu joko RSA (Rivest–Shamir–Adleman) tai hash -suojauksella, jonka avulla datan eheydestä voitaisiin olla varmoja (Kumar, 2016). Zunnurhain ja Vrbskyn (2011) aikaisemmin mainitseman hypervisorin avulla voidaan tehtäviä aikatauluttaa näiden joukkojen sisällä. Hypervisor suorittaisi validoinnin jokaiselle uudelle koodin pätkälle joka joukkoon tulee. Jos hypervisor tunnistaa koodista jonkin ei-oikeudellisen koodin pätkän joka keskeyttää palvelin normaalia työskentelyä, pystytään tämä instanssi eristää introspektion avulla. (Zunnurhain & Vrbsky, 2011.) Tällä tavalla tulvahyökkäystä voidaan edes jonkin verran mitigoida. Kuitenkin jos hypervisor saastuu itse joukon sisältä, tämä vaatisi kuitenkin sen, että joku henkilö jolla on joukkoon järjestelmänvalvojan oikeudet saastuttaisi sen. Tämä on siten erittäin epätodennäköinen

skenaario, mutta jos näin kävisi, pitäisi hypervisoria analysoida lisää ja implementoida sille uudet turvaominaisuudet. (Zunnurhain & Vrbsky, 2011.)

6. Yhteenveto

Tutkimuksessa käsiteltiin pilvipalveluita ihan yleisellä tasolla, niiden ongelmia, kolmea eri hyökkäystekniikka ja ratkaisuja näihin hyökkäyksiin. Tutkimus aloitettiin esittelemällä pilvipalvelutyypit ja mallit. Yleisesti ottaen kirjallisuudessa pilvipalvelutyyppejä olivat julkinen, yksityinen ja hybridipilvi. Kaikki kolme tyyppiä käyttävät pay-per-use -mallia, mutta niillä oli muutamia eriäväisyyksiä, kuten esimerkiksi se, että julkiseen pilveen pääsee käsiksi ihan pelkästään internetselaimella. Yksityinen pilvi sen sijaan sijaitsee organisaation tai yrityksen omien servereiden sisällä, jolloin pääsy sinne on rajoitettu organisaation tai yrityksen työntekijöille ja henkilöstölle. Yksityinen pilvi eroaa julkisesta pilvestä myös sillä tavalla, että kaikki pilven resurssit ja sovellukset ovat suoraan kyseisen organisaation hallinnassa.

Pilvipalvelumalleja taas ovat sovellustaso (SaaS), alustataso (PaaS) ja infrastruktuuritaso (IaaS). Sovellustaso on pilvipalveluiden ylin taso ja samalla myös suosituin pilvipalvelumalli, esimerkkinä tästä olisi vaikka google drive, jota itsekin käytän päivittäin. Alustataso on hyvin pitkälti samanlainen, kuin infrastruktuuritaso, mutta se tuo mukanaan vielä yhden tason enemmän vuokrattua funktionaalisuutta. Asiakkaalla ei kuitenkaan ole pääsyä kontrolloimaan ohjelmistoalustan alla piilevää infrastruktuuria.

Luku neljä aloitettiin esittelemällä aluksi pilvipalveluiden tietoturvavaatimukset, jonka jälkeen annettiin vastaus ensimmäiseen tutkimuskysymykseen ”Mitkä ovat pilvipalveluiden yleiset tietoturvaongelmat?”. SOAP-viestit, faktoroidut käyttöjärjestelmät sekä virtuaalikoneen introspektio muodostavat pilvipalveluiden yleisimmät tietoturvaongelmat.

Luku viisi antoi vastauksen toiseen tutkimuskysymykseen ”Mitkä ovat pilvipalveluihin kohdistuvat yleisimmät hyökkäysmenetelmät ja mitä ratkaisuja niihin voidaan soveltaa?”. Wrapping-hyökkäys, haittaohjelmainjektio ja tulvahyökkäys ovat pilvipalveluihin kohdistuvat yleisimmät hyökkäysmenetelmät. Wrapping-hyökkäyksen ratkaisuun voidaan käyttää julkista sertifikaattia. Haittaohjelmainjektion ja tulvahyökkäyksen ratkaisemiseen voidaan käyttää virtuaalikonemonitoria eli hypervisoria.

Lähdeluettelo

Al Morsy, M., Grundy, J. & Müller, I. (2010). An Analysis of the Cloud Computing Security Problem. *Asia-Pacific Software Engineering Conference, APSEC-2010, November 30, 2010, Proceedings.* https://doi.org/10.48550/arXiv.1609.01107

Armbrust, M. & Fox, A. (2010). A View of Cloud Computing. https://doi.org/10.1145/1721654.1721672

Cachin, C., Keidar, I. & Shraer, A. (2009). Trusting the cloud. *Acm Sigact News, 40(2), 81-86.* https://doi.org/10.1145/1556154.1556173

Calheiros, R.N., Ranjan, R., Beloglazov, A., De Rose, C. A. & Buyya, R. (2011). CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience, 41(1), 23-50.* https://doi.org/10.1002/spe.995

Dillon, T., Wu, C. & Chang, E. (2010). Cloud Computing: Issues and Challenges. *International Conference on Advanced Information Networking and Applications, 20-23 April, 2010, Proceedings.* https://doi.org/10.1109/AINA.2010.187

Google Scholar. https://scholar.google.com/

Hayes, B. (2008). Cloud computing. *Communications of the ACM, 51(7), 9-11.* https://doi.org/10.1145/1364782.1364786

Kumar, P. (2016). Cloud Computing: Threats, Attacks and Solutions. *International Journal of Emerging Technologies in Engineering Research, 4(8), 6-11.* https://ijeter.everscience.org/Manuscripts/Volume-4/Issue-8/Vol-4-issue-8-M-06.pdf

Mell, P. & Grance, T. (2011). The NIST Definition of Cloud Computing. *National Institute of Standards and Technology Special Publication 800-145.* https://doi.org/10.6028/NIST.SP.800-145

Payne, B.D., Carbone, M., Sharif, M. & Lee, W. (2008). Lares: An Architecture for Secure Active Monitoring Using Virtualization. *IEEE Symposium on Security and Privacy, 2008, 233-247.* https://doi.org/10.1109/SP.2008.24

Rahaman, M., Rits, M. & Schaad, A. (2006). An Inline Approach for Secure SOAP Requests and Early Validation. *The Open Web Application Security Project Europe Conference, OWASP-2006, May 30-31, 2006, Proceedings.* https://owasp.org/www-pdf-archive/AnInlineSOAPValidationApproach-MohammadAshiqurRahaman.pdf

Ramgovind, S., Eloff, MM. & Smith, E. (2010). The Management of Security in Cloud Computing. *Information Security for South Africa, 1-7.* https://doi.org/10.1109/ISSA.2010.5588290

Scopus. https://www.scopus.com/

Takabi, H., Joshi, J. & Ahn, G. (2010). Cloud Computing: Security and Privacy challenges in Cloud Computing Environments. *IEEE Security & Privacy, 8(6), 24-31.* https://doi.org/10.1109/MSP.2010.186

Viestintävirasto. (2014). Pilvipalveluiden turvallisuus. Haettu 7.3.2022 osoitteesta https://www.kyberturvallisuuskeskus.fi/sites/default/files/media/file/Pilvipalveluiden_tietoturva_organisaatioille.pdf

Viega, J. (2009). Cloud Computing and the Common Man. *Computer, 42(8), 106-108*. https://doi.org/10.1109/MC.2009.252

Zhang, Q., Cheng, L. & Boutaba, R. (2010). Cloud computing: State-of-the-art and research challenges. *Journal of Internet Services and Applications, 1(1), 7-18.* https://doi.org/10.1007/s13174-010-0007-6

Zissis, D. & Lekkas, D. (2012). Addressing Cloud Computing Security Issues. *Future Generation Computer Systems, 28, 583-592* https://doi.org/10.1016/j.future.2010.12.006

Zunnurhain, K. & Vrbsky, S. (2011). Security in Cloud Computing. *International Conference on Security and Management, SAM-2011, Proceedings*. https://www.researchgate.net/publication/267710616_Security_in_Cloud_Computing