Tampereen yliopisto

Ville Aarnio

JULKISEN AVAIMEN INFRASTRUK-TUURI PILVIPALVELUISSA

Informaatioteknologian ja viestinnän tiedekunta
Diplomityö
Maaliskuu 2022

TIIVISTELMÄ

Ville Aarnio: Julkisen avaimen infrastruktuuri pilvipalveluissa
Diplomityö
Tampereen yliopisto
Informaatioteknologian ja viestinnän tiedekunta - tietoturvallisuus
Maaliskuu 2022

Tutkimus käsittelee julkisen avaimen infrastruktuuria pilvipalveluissa. Tutkimuksessa selvitettiin, miten julkisen avaimen infrastruktuuria käytetään pilvipalveluissa, miten avainten säilytys voidaan toteuttaa niitä varten ja miten varmenteita voidaan viedä pilvipalveluihin kolmannen osapuolen varmennepalvelusta.

Tutkimuksen päätavoite oli löytää ratkaisu siihen, miten varmenteita voidaan viedä pilvipalveluihin kolmannen osapuolen varmennepalvelusta. Tämän ratkaisu tuli olla tietoturvallinen sekä automatisoitavissa. Päätavoitteen lisäksi työssä tutkittiin, miten avaimia voidaan säilyttää siten, että ne ovat käytettävissä ja miten julkisen avaimen infrastruktuuria yleisesti käytetään pilvipalveluissa.

Tutkimuksen teoriaosuus koostuu julkisen avaimen infrastruktuurin, varmenteiden ja pilvipalveluiden teoriasta. Tutkimusosuus koostuu työn kannalta oleellisten pilvipalveluratkaisujen kartoituksesta, avaintenhallintaratkaisuiden esittelystä sekä arvioinnista ja kolmannen osapuolen varmennepalvelun mahdollisuuksien arvioinnista pilvipalveluissa.

Tutkimus toteutettiin kirjallisuustutkimuksena sekä kokeellisena tutkimuksena. Kirjallisuustutkimuksessa käytettiin aiheeseen liittyvää kirjallisuutta sekä pilvipalveluiden dokumentaatioita. Kokeellisessa tutkimuksessa käytettiin pilvipalveluita sekä näiden käyttöön liittyviä muita järjestelmiä, kuten rajapintoja.

Tutkimuksen päätavoitteen ratkaisuksi ehdotettiin erillistä ohjelmistoa pilvipalvelun ja varmennepalvelun välille. Tämä ohjelmisto hoitaa muun muassa varmenteiden viennin pilvipalveluun ja niiden elinkaaren hallinnan. Tämä ratkaisu täyttää tutkimuksen tavoitteen vaatimukset turvallisuudesta ja se on automatisoitavissa. Tutkimuksessa käsiteltiin myös pilvipalveluiden jo olemassa olevia valmiita ratkaisuja, mutta ne todettiin riittämättömiksi ilman erillistä ohjelmistoa.

Avaintenhallintaratkaisuiden osalta tutkimuksessa käsiteltiin erilaisia vaihtoehtoja, joista osa oli pilvipalveluiden sisäisiä ratkaisuja ja osa näiden ulkopuolisia. Näistä on-premise-ratkaisu todettiin turvallisimmaksi ja muut ratkaisut ovat ulkopuolisten toimijoiden luottamuksen varassa. On-premise-ratkaisussa oli myös ongelmana sen integroiminen osaksi pilvipalveluita, joten selvää ratkaisua tähän ei löytynyt ja tämä vaatii jatkotutkimusta. Kolmanneksi tulokseksi työssä saatiin lista palveluista, jotka käyttävät julkisen avaimen infrastruktuuria. Tällaisia palveluita olivat muun muassa virtuaalikoneet, API-rajapinnat sekä tiedostojen allekirjoitukset.

Avainsanat: Julkisen avaimen infrastruktuuri, pilvipalvelu, varmenne, Azure, Amazon Web Service, Google Cloud Platform

ABSTRACT

Ville Aarnio: Public key infrastructure in cloud platforms
Master of science thesis
Tampere university
Faculty of Information Technology and Communication Sciences, Cyber security
March 2022

The study focuses on public key infrastructure on cloud platforms. The study looked at how public key infrastructure is used in cloud platforms; how key management can be implemented in a way that it can be used in cloud services and how certificates can be transferred to cloud from a third-party certification service.

The main objective of the study was to find a solution on how certificates can be transferred to cloud platforms from a third-party certification service. This solution had to be secure and automatable. Additionally, the study focused on how encryption keys can be stored in a way that the keys can be used and how public key infrastructure is used in cloud platforms in general.

The theory part of the study consists of public key infrastructure, certificates, and cloud services theories. The research part consists of mapping the relevant cloud services on the platforms, introducing the key management solutions, and valuating them and valuating the possibilities for a third-party certification services in cloud platforms.

The study was done as literature study and experimental study. In the literature study the relevant literature and cloud platform documentations were used. In the experimental study cloud platforms and other systems related to them, such as API's, were used.

The proposed solution to the main objective was a separate software between the cloud platforms and certification service. This software handles for example transfer and lifecycle management of the certificates. This solution fulfils the requirements of the objective; it can be made secure, and it is automatable. Cloud integrated solutions were also researched, but those were found to be insufficient without the separate software.

In the key management part few different solutions were researched. Some of the solutions were inside of the cloud platforms and others outside of the cloud. On-premise solution was found to be most secure and other solutions relied on trust to the other parties. There was also a challenge regarding on-premise solution, where it was found to be hard to integrate to be a part of the cloud platform. Therefore, no one solution was found, and this part requires more research. Third result of the study was a list of services that uses public key infrastructure. These services were for example virtual machines, APIs, and file singing.

Keywords: Public key infrastructure, cloud platform, certificate, Azure, Amazon web service, Google cloud platform

ALKUSANAT

Noin vuosi sitten, kun tämän työn aloitin, en tiennyt aiheesta juuri mitään. En olisi osannut arvata, että vuoden aikana voi oppia niinkin paljon, kuin mitä olen oppinut. Suuri kiitos tästä kuuluu mahtaville työkavereilleni Insta Advancella. Erityiskiitokset Mika Suvannolle ja Antti Lahnaojalle hyvästä ohjauksesta ja työn kommentoinnista sekä Juha Luukkaselle aiheen rajauksesta ja siitä, että hän otti minut tekemään työtä Installe.

Kiitos myös veljelleni Laurille työn lukemisesta ja kommentoinnista. Suuri kiitos myös avopuolisolleni Terhille, joka jaksoi tukea ja kannustaa minua tämän työn kanssa ja koko opiskelujeni ajan.

Tampereella, 15.3.2022

Ville Aarnio

SISÄLLYSLUETTELO

LYHENTEET JA MERKINNÄT

ACM	AWS Certificate Manager
AWS	Amazon Web Services
CA	Certification Authority
CMS	Cryptographic Message Syntax
CRL	Certificate Revocation List
CSR	Certificate Signing Request
DN	Distinguished Name
GCP	Google Cloud Platform
HSM	Hardware Security Module
KMS	Key Management System
OCSP	Online Certificate Status Protocol
PGP	Pretty Good Privacy
PKI	Public Key Infrastructure
RA	Registration Authority
REST API	Representational State Transfer Application Programming Interface
S/MIME	Secure Multipurpose Internet Mail Extension
SPKI	Simple Public Key Infrastructure
SSL	Secure Sockets Layer
TLS	Transport Layer Security
VA	Verification Authority
VPN	Virtual Private Network

1. JOHDANTO

1.1 Kohdeyrityksen esittely

Tämä diplomityö on toteutettu Insta Advance Oy:lle, joka on osa Insta Group Oy:tä. Insta on suomalainen perheyritys, joka toimii teollisuusautomaation, teollisen digitalisaation, kyberturvan sekä puolustusteknologian osa-alueilla. Instan palvelut liittyvät hyvin vahvasti turvalliseen yhteiskuntaan.

Insta Advance Oy toimii kyberturvallisuuden sekä teollisen digitalisaation toimialoilla. Insta Advance Oy tuottaa muun muassa korkean turvatason varmenneratkaisuja omasta palvelukeskuksesta.

1.2 Työn tausta

Pilvipalvelut ovat yleistyneet nopeasti ja yhä useampi palvelu pyörii pilvipalveluissa perinteisten konesalien sijasta. (Synergy Research Group, 2021) Pilvipalvelut ovat usein kustannustehokkaampia, sillä esimerkiksi laitehallintakustannuksia ei ole. Tästä syystä niiden käytön aloittamisen kynnys on pienempi verrattuna oman konesalin perustamiseen. Pilvipalvelut voidaan mieltää tietoturvallisiksi, mutta ongelmakohtana näissä on luottamus.

Pilvipalveluiden turvallisuusratkaisut eivät ole läpinäkyviä ja tästä syystä usein ei pystytä todistamaan, että tietoihin ei pääse käsiksi henkilö, jolla ei tulisi olla oikeuksia niihin. Tätä varten halutaan keksiä ratkaisu siihen, miten julkisen avaimen infrastruktuuri voitaisiin toteuttaa pilvipalvelun ulkopuolella siten, että sitä voitaisiin hyödyntää pilvipalveluissa.

Installa toimii varmennepalvelu, joka sisältää muun muassa varmentajan (eng. Certificate Authorityn, CA) hallinnoinnin sekä varmenteiden elinkaaren hallinnan. Instassa ei kuitenkaan myönnetä varmenteita tällä hetkellä pilvipalveluihin ja näiden yleistymisen takia on tarpeellista selvittää Instan mahdollisuudet julkisen avaimen infrastruktuurin käyttämisestä näissä palveluissa.

Tutkimuksen tarkoitus oli selvittää julkisen avaimen infrastruktuurin hallintaa pilvipalveluja varten. Tutkimuksessa käsiteltiin varmenteiden käyttötapauksia pilvipalveluissa ja miten varmenteiden jakelu sekä hallinta toimii siten, että niitä voidaan käyttää pilvipalveluissa. Lisäksi tutkittiin, onko varmenteita myöntävä varmennepalvelu

integroitavissa osaksi pilvipalvelussa toimivaa ohjelmistoratkaisua. Tämän lisäksi tutkittiin avaintenhallintaratkaisuja pilvipalveluja varten.

Instan varmennepalvelut tuottavat korkean turvatason varmenneratkaisuja ja tätä Instan asiakkaat odottavat myös tulevaisuudessa. Tästä syystä tässä työssä kartoitettiin ratkaisuja varmennepalveluiden integroimiseen pilvipalveluihin siten, että pystytään takaamaan riittävä tietoturvallisuus. Instan asiakkaat käyttävät Instan varmennepalveluja useimmissa tapauksissa automatisoidusti, joten oli oleellista tutkia mahdollisuuksia toteuttaa varmenteiden elinkaaren hallinta tällaisella käyttötapauksella.

Tutkimusongelman perustana oli pilvipalveluiden käytön yleistyminen ja ohjelmistoratkaisujen siirtyminen pilvipalveluihin. Varmennepalvelun toiminnan jatkuvuus pyritään siis takaamaan liiketoimintanäkökulmasta myös tulevaisuudessa.

Pilvipalveluissa on myös käytössä paljon erilaisia palveluja, joten osa tutkimusongelmaa oli selvittää se, millaisissa tapauksissa Instan julkisen avaimen infrastruktuurin ratkaisuja voitaisiin hyödyntää pilvipalveluissa. Nämä ratkaisut rajattiin sen mukaan, mitä Installa on mahdollista toteuttaa ja rajauksesta jätettiin pois palvelut, joihin Insta ei myönnä varmenteita. Tällaisia ovat esimerkiksi internet-selainten oletusarvoisesti luottamat Transport Layer Security-varmenteet (TLS). Kolmannen osapuolen varmenneratkaisujen käyttämisen lisäksi työssä esiteltiin yleisesti pilvipalveluissa toimivia varmenneratkaisuja.

1.3 Tutkimuksen tavoitteet

Tämän diplomityön tavoitteena oli tutkia julkisen avaimen infrastruktuuria pilvipalveluissa ja sen mahdollisuuksia kolmannen osapuolen varmennepalvelulle sekä tähän liittyvien avainten säilytystä. Työn tavoite oli vastata seuraaviin kysymyksiin:

1. Miten kolmannen osapuolen varmenneratkaisuja voidaan hyödyntää pilvipalveluissa?
2. Miten julkisen avaimen infrastruktuuria käytetään pilvipalveluissa?
3. Minkälaisia avaimenhallintaratkaisuja pilvipalvelut tarjoavat?

Näistä ensimmäinen on päätutkimuskysymys ja toinen ja kolmas ovat sitä tukevia kysymyksiä.

1.4 Tutkimusmenetelmät ja tutkimuksen rajaus

Työn tutkimus toteutettiin kirjallisuustutkimuksena ja kokeellisena tutkimuksena. Työssä käytettiin pilvipalveluja sekä tutustuttiin niiden dokumentaatioon. Työn alussa

tunnistettiin ongelma ja sen osat. Tämän jälkeen määriteltiin pilvipalvelut, jotka olivat tutkimuksen kohteena. Näitä käyttämällä sekä dokumentaatioon tutustumalla etsittiin ongelmalle ratkaisu ja määritettiin onko ratkaisu riittävä ja toimiva.

Kirjallisuustutkimuksessa pyrittiin käyttämään vertaisarvioituja lähteitä poislukien pilvipalveluiden dokumentaatiot. Hakusanoja, joita tässä käytettiin olivat muun muassa ”PKI”, ”public key infrastructure”, ”cloud”, ”Azure”, ”Amazon web services”, ”Google cloud platform”, ”certificate” sekä näiden yhdistelmiä. Kokeellinen tutkimus sisälsi muun muassa pilvipalveluiden sekä API-rajapintojen käyttöä. Kokeellisen tutkimuksen apuna käytettiin myös pilvipalveluiden dokumentaatioita.

Työssä tutkittiin kolmea pilvipalveluntarjoajaa; Microsoft Azurea, Amazon Web Serviceä sekä Google Cloud Platformia. Nämä ovat suurimmat pilvipalveluntarjoajat ja tästä syystä valikoituivat työssä käsiteltäviksi.

1.5 Aiheeseen liittyviä tutkimuksia

Pilvipalveluiden turvallisuudesta sekä luottamuksesta on tehty useita tutkimuksia. Nämä tutkimukset kuitenkin keskittyvät yleisellä tasolla pilvipalveluihin, eivätkä tutkimukset käsittele yksittäisiä pilvipalveluita.

Artikkeli ”Addressing cloud computing security issues” käsittelee pilvipalveluiden turvallisuutta ja esittää mallin sen parantamiseen julkisen avaimen infrastruktuurin avulla. Tutkimuksen esittämässä mallissa jokaisella pilven osapuolella on varmenne, jonka on myöntänyt luotettu kolmas osapuoli. Näitä varmenteita käytetään autentikointiin sekä salaamiseen ja niitä käyttämällä voidaan tutkimuksen mukaan parantaa koko palvelun eheyttä, luottamuskellisuutta, autentikointia sekä saatavuutta. Tutkimuksen esittämässä mallissa varmenteita olisi esimerkiksi käyttäjillä, sovelluksilla, palvelimeilla sekä fyysisillä laitteilla. (Zissis & Lekkas, 2012)

Toinen aiheeseen liittyvä artikkeli ”The Future Internet: A World of Secret Shares” keskittyy pilvlipalvelun aiheuttamiin vaikeuksiin avainten säilytyksessä. Julkisen avaimen infrastruktuurin heikkous on tutkimuksen mukaan yksityisen avaimen vaarantuminen. Tämä on ongelma erityisesti julkisissa pilvipalveluissa. Tutkimus esittää erilaisia malleja datan säilytykseen sekä jakamiseen turvallisesti. (Buchanan et al., 2015)

Tutkimuksessa ”Cloud-Trust – a Security Assessment Model for Infrastructure as a Service (IaaS) Clouds” pyrittiin kehittämään ratkaisu pilvipalveluntarjoajien luottamiseen. Tutkimuksen perustana oli se, että pilvipalveluissa käytössä olevat laitteistot voivat olla monen käyttäjän käytössä samaan aikaan ja pilvipalveluntarjoajat voivat hallita tietokoneita. Tutkimuksen tarkoitus oli kehittää malli pilvipalvelujen tietoturvan arviointiin.

Tutkimuksessa kehitetty malli ”Cloud-Trust” osoittautui onnistuneeksi erilaisten tietoturvapoikkeamien havaitsemiseen. Siinä jäi kuitenkin osia palveluntarjoajan mahdollisuuksista pimentoon ja niin sanotut sisäpiirin hyökkäykset voivat olla vielä mahdollisia. (Gonzales et al., 2017)

1.6 Tutkimuksen rakenne

Työ alkaa luvusta kaksi, jossa esitellään julkisen avaimen infrastruktuuri. Tässä luvussa esitellään julkisen avaimen infrastruktuurin perusperiaatteet sekä tarkoitukset. Luvussa kolme esitellään varmenteet, niiden liittyminen julkisen avaimen infrastruktuuriin, niiden ominaisuudet sekä hallinta. Luvussa neljä esitellään pilvipalveluita sekä niiden käyttötarkoituksia.

Luvut kaksi, kolme ja neljä luovat pohjan työn lopuille luvuille. Luvussa viisi esitellään julkisen avaimen infrastruktuurin toimintaa pilvipalveluissa. Luvussa kuusi esitellään avaintenhallintaratkaisuja pilvipalveluita varten. Lopuksi luvussa seitsemän esitellään kolmannen osapuolen varmenneratkaisujen integroimista osaksi pilvipalveluja ja näin pyritään vastaamaan työn päätutkimuskysymykseen.

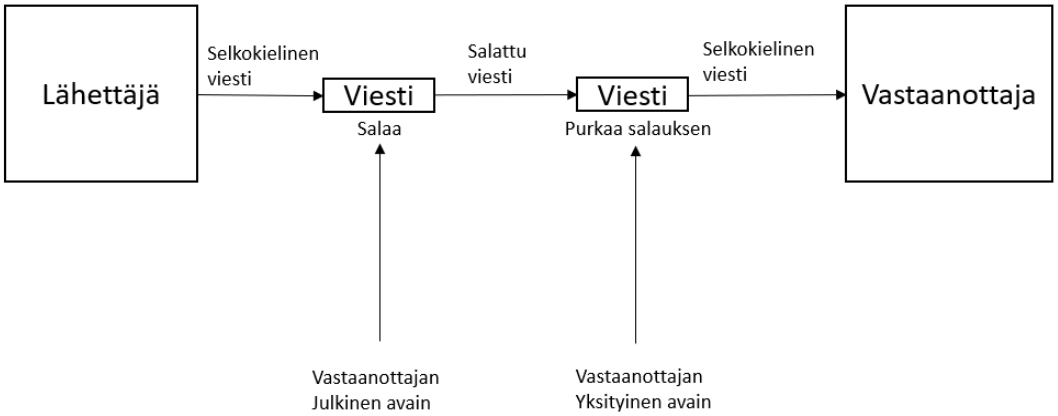
2. JULKISEN AVAIMEN INFRASTRUKTUURI

2.1 Johdatus julkisen avaimen infrastruktuuriin

Ihmiset ovat historiallisesti aina halunneet kommunikoida siten, että kommunikoitu viesti jää vain heidän välisekseen. Tätä varten on kehitetty matemaattisia algoritmeja, joiden tarkoitus on muuttaa kommunikoitu viesti sellaiseksi, että siitä ei voida saada alkuperäistä viestiä luettua. Tämä tapahtuu yhdistäen viesti tietynlaiseen avaimeen. Jotta kuitenkin alkuperäinen viestin saaja pystyy lukemaan viestin, tulee algoritmin olla sellainen, että viesti saadaan muutettua takaisin alkuperäiseen. (Adams et al., 2003)

Tämä salaus onnistuu siten, että molemmilla tahoilla on sama avain ja viestin pystyy salaamaan ja purkamaan samalla avaimella. Tämä on nimeltään symmetrinen kryptografia. Tämä luo kuitenkin ongelmia avaimen jaossa, sillä avain voi vaarantua esimerkiksi joutumalla kolmannen osapuolen haltuun avaimen vaihdon yhteydessä. Tätä ongelmaa ei ole julkisen avaimen kryptografiassa eli niin sanotussa asymmetrisessa kryptografiassa. (Braeken, 2021)

Julkisen avaimen kryptografiassa viestin salaamiseen ja purkamiseen käytetään eri avaimia siten, että nämä avaimet liittyvät toisiinsa. Viesti salataan vastaanottajan julkisella avaimella ja viestin voi purkaa vastaanottajan yksityisellä avaimella kuvan 1 mukaisesti. (Stapleton, 2016)



Kuva 1. Viestin lähetys salattuna julkisen avaimen infrastruktuurilla

Julkinen avain voidaan siis jakaa julkisesti, mutta se tulee tapahtua siten, että varmistetaan avaimen jakajan olevan oikeasti sen omistaja. Avainpari sekä avainten toiminta ja säilytys käsitellään luvussa 2.5. Julkinen avaimen omistajan oikeellisuus voidaan todeta sisällyttämällä se varmenteeseen, joita käsitellään luvussa 3.

Kahden henkilön välisen kommunikaation lisäksi julkisen avaimen kryptografiaa voidaan käyttää käyttäjän tunnistautumisessa esimerkiksi web-sivulle, tai voidaan tunnistaa laitteita tai palveluita toisilleen. Julkisen avaimen kryptografialla voidaan myös esimerkiksi digitaalisesti allekirjoittaa tiedostoja, jolloin voidaan varmistaa tiedoston oikeellisuus. (Stapleton, 2016)

Julkisen avaimen infrastruktuurin perustana on luottamus. Tähän liittyen luottamus varmenteita myöntävään palveluun eli Certificate Authorityyn (CA) on tärkeää. Varmenteisiin luottavan osapuolen tulee olla varma, että CA on yhdistänyt oikean käyttäjän oikeaan julkiseen avaimeen. (Huang & Nicol, 2017 s.245-248) CA:ta käsitellään tarkemmin luvussa 2.2.1.

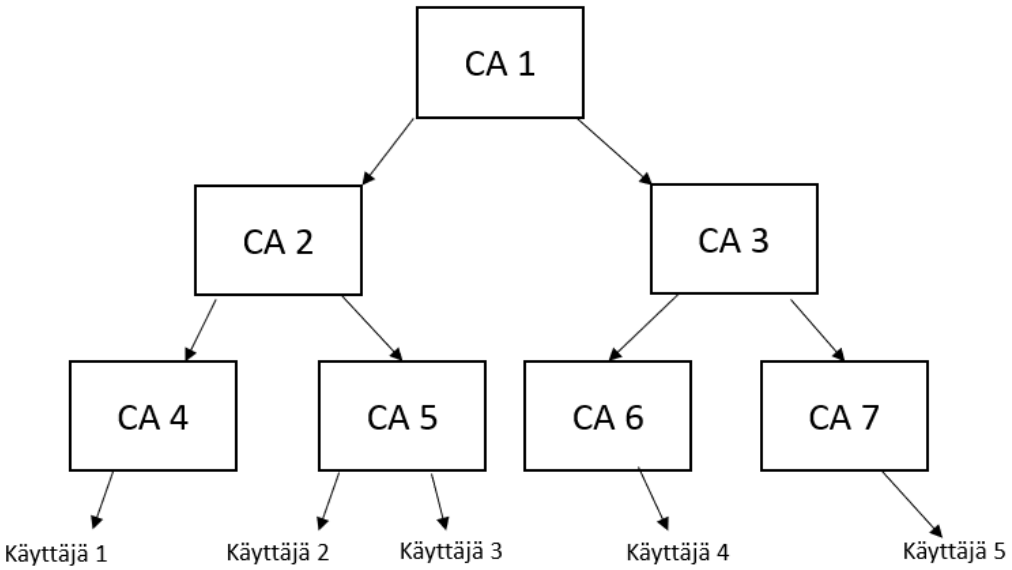
2.2 Osapuolet

PKI:hin liittyy osapuolia, joilla jokaisella on tietty tarkoitus sen osana. Näitä osapuolia ovat CA, loppukäyttäjä ja luottava osapuoli sekä joissain tapauksissa RA. Loppukäyttäjä saa CA:lta varmenteen, jonka tarkoitus on todistaa luottavalle osapuolelle se, että loppukäyttäjä on julkisen avaimen oikea omistaja.

2.2.1 Certification Authority

Certification Authority (CA) on taho, joka myöntää ja hallitsee varmenteita. CA toimii siis loppukäyttäjän ja luottavan osapuolen välissä. Tämä on esitetty kuvassa 4.

CA voi olla joko online tai offline CA. CA:t toimivat hierarkiassa siten, että jokainen CA myöntää hierarkiassa suoraan tästä alemmalla tasolla olevalle CA:lle varmenteen ja hierarkian alin CA myöntää varmenteen loppukäyttäjälle. Tämä täytyy kuitenkin toteuttaa siten, että vaikka käyttäjämäärä kasvaisi suureksi, hierarkia pysyy hyväksyttävän kokoisena. Kuvassa 2 on esitetty kolmitasoinen CA-hierarkia. (Stapleton, 2016)



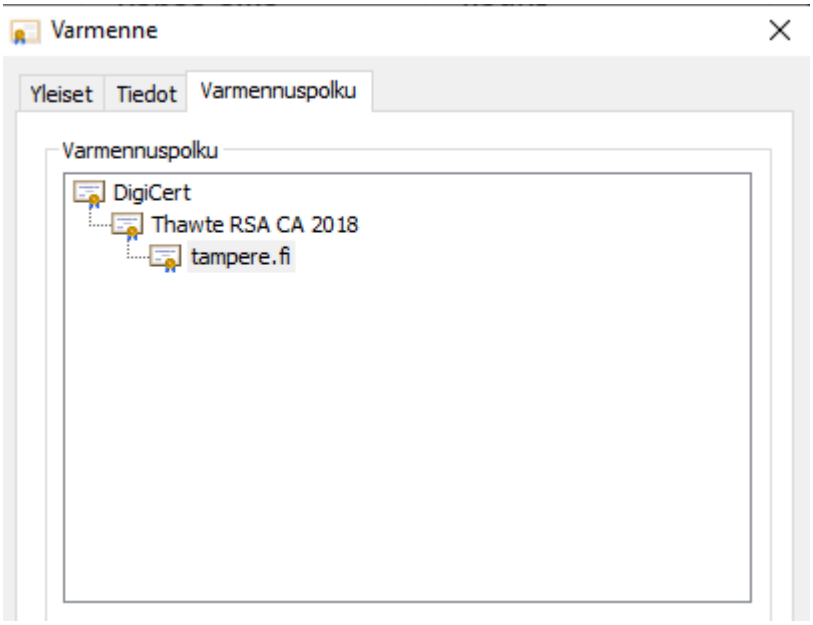
Kuva 2. 3-tasoinen CA-hierarkia.

Hierarkian ylimmällä tasolla (CA 1) on juurivarmentaja. Se pidetään yleensä internetin ulkopuolella ja sitä käytetään vain silloin, kun sillä myönnetään tai uusitaan varmenteita alemman tason alivarmentajille (CA 2 ja CA 3). Tällä tavalla voidaan vähentää riskiä siitä, että juurivarmentaja vaarantuisi, sillä juurivarmentajan vaarantuessa mihinkään alivarmentajaan ei voida enää luottaa. Kuvan 2 tapauksessa jokaisen mukana olevan tahon tulee saada kopio juurivarmentajan julkisesta avaimesta, jotta luottamus voidaan taata. (Stapleton, 2016)

Jos esimerkiksi käyttäjä 1 saa viestin käyttäjältä 4, täytyy sen käydä varmennereitti CA1-CA3-CA6. Tässä tapauksessa käyttäjän 1 tulee luottaa jokaiseen CA:han reitin varrella.

CA voi olla käytössä yrityksen omilla palvelimilla, tai CA:na voi toimia valtio tai erilliset PKI-toimijat. CA ei pelkästään myönnä varmenteita, vaan sen täytyy myös pystyä peruuttamaan myönnettyjä varmenteita tapauksissa, joissa niihin ei voida enää luottaa. CA:n fyysinen turvallisuus ja tietoturvallisuus on pidettävä riittävänä. Jos jonkin CA:n turvallisuus rikkoutuu, ei sen myöntämiin varmenteisiin voida enää luottaa. Riippuen CA:sta, tämä voi tarkoittaa todella suurta määrää varmenteita. (Huang & Nicol, 2017 s.245-248)

Esimerkki tampere.fi-sivulle myönnetyn varmenteen varmennehierarkiasta on esitetty kuvassa 3. Tässä kuvassa DigiCert toimii juurivarmentajana ja Thawte RSA CA 2018 on alivarmentaja, joka on myöntänyt varmenteen sivustolle tampere.fi.



Kuva 3. Esimerkki varmennehierarkiasta sivuston tampere.fi varmenteesta

2.2.2 Loppukäyttäjä

Loppukäyttäjä on se taho, jolle varmenne on myönnetty ja jonka nimi tai muu yksilöivä tieto näkyy varmenteessa. Loppukäyttäjä tekee varmennepyynnön CA:lle ja CA myöntää tämänvarmenteen loppukäyttäjälle. (Stapleton, 2016) chapter 5.4 Tämä tapahtuma on esitetty kuvassa 4.

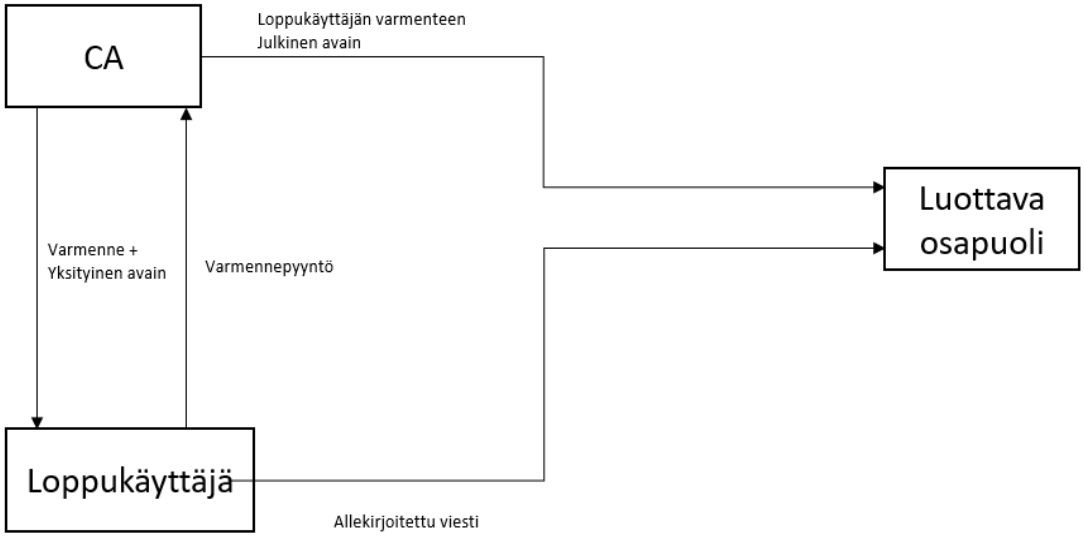
Loppukäyttäjä voi käyttää varmennetta esimerkiksi keskustellessaan luottavan osapuolen kanssa. Loppukäyttäjä allekirjoittaa viestinsä yksityisellä avaimellaan ja lähettää sen luottavalle osapuolelle. Loppukäyttäjä voi olla esimerkiksi henkilö tai tietoliikennelaite. (Stapleton, 2016)

2.2.3 Luottava osapuoli

Luottava osapuoli on taho, joka luottaa varmenteeseen. Tämän osapuolen tulee siis varmistaa varmenteen oikeellisuus ja se, että varmenne ei ole poistettu käytöstä.

Varmenne ei saa olla vanhentunut, suljettu ja sen pitää soveltua käyttötarkoitukseen. (ØLNES & BUENE, 2006)

Luottavan osapuolen tulee verrata myöntävän CA:n ominaisuuksia riskeihin. CA:han liittyviä riskejä voi olla esimerkiksi maa missä CA toimii, CA:ta ylläpitävän tahon taloudellinen tilanne tai sen maine. Luottava osapuoli voi joutua hyväksymään varmenteita monelta eri CA:lta, jolloin riskejä voi olla vaikea hallita. (ØLNES & BUENE, 2006) Luottava osapuoli selvittää varmennepolitiikkojen avulla miten CA lupaa toimia ja voiko CA:han luottaa. (Huang & Nicol, 2017 s.245-248)



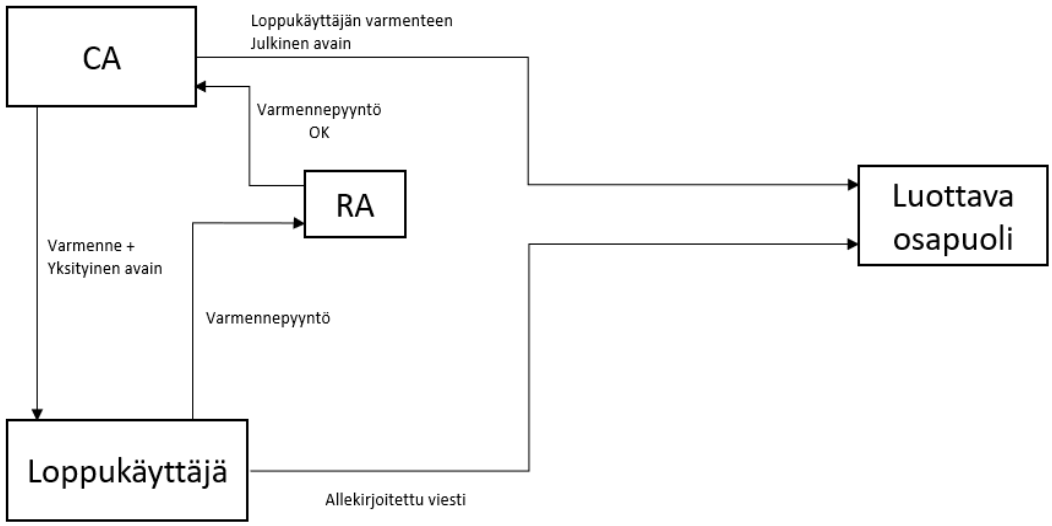
Kuva 4. PKI-järjestelmän osapuolet.

2.2.4 Registration Authority

Registration Authority (RA) on taho, joka tunnistaa käyttäjän ja pyytää varmenteita CA:lta käyttäjän puolesta. RA tunnistautuu CA:lle luotetulla tavalla ja käyttäjä tunnistautuu RA:lle luotetulla tavalla. RA ei ole pakollinen osa PKI:ta, mutta sitä voidaan käyttää helpottamaan PKI:n hallintaa. (Froehlich, 2021)

Erillisen RA:n käyttö tuo PKI:hin lisää suojaa, sillä yksittäinen käyttäjä ei voi pyytää varmenteita. Jos tämä olisi mahdollista ja tämä yksittäinen käyttäjä vaarantuisi, voisi hän pyytää varmenteita haitallisiin tarkoituksiin. Kun käytetään RA:ta, voi sitä varten olla käytössä kokonaan eri osasto yrityksen sisällä, joka ei liity käyttäjään. Täten käyttäjä ei

voi mielivaltaisesti pyytää varmenteita CA:lta. (Hunt, 2001) Toinen etu RA:n käytöstä on, että kun tiettyyn PKI järjestelmään tulee paljon käyttäjiä, niin RA:lla tai monella RA:lla voidaan tehdä prosessista järjestelmällisempää. RA:n toiminta lisänä kuvan 4 PKI-ratkaisua on esitetty kuvassa 5. (Adams et al., 2003)



Kuva 5. RA osana PKI:ta

2.3 Käyttötarkoitus

Tässä luvussa käsitellään julkisen avaimen infrastruktuurin päätarkoituksia jotka ovat: autentikointi, kiistämättömyys ja luottamuksellisuus. (Adams et al., 2003) Nämä käyttötarkoitukset määrittävät turvallisuuden ne täyttymällä voidaan myös PKI todeta turvalliseksi.

2.3.1 Autentikointi

Autentikoinnin tarkoitus on varmentaa käyttäjän, palvelun tai datan identiteetti. Ilman tällaista tunnistautumista ei ole mahdollista tehdä turvallista applikaatiota, mutta se itsessään ei takaa täyttä turvallisuutta. (Adams et al., 2003) Tunnistautuminen on usein kaksisuuntaista siten, että molemmat tunnistamiseen osallistuvat tahot tunnistautuvat toisilleen. Esimerkiksi käyttäjä tunnistautuu web-sivulle salasanalla ja web-sivu tunnistautuu käyttäjälle varmenteella.

Datan alkuperän tunnistautumisessa tietty kohde tunnistetaan datan lähteeksi. Tämän tunnistautumisen tarkoitus on sitoa tietty taho tiettyyn dataan ja täten voidaan varmistaa, että datan lähettäjä on oikea. (Vacca, 2004)

Yksilön tunnistautumisen tarkoitus on muiden aktiviteettien suorittaminen tai turvallisen kommunikoinnin mahdollistaminen. Näitä tunnistautumisia voi olla esimerkiksi paikallinen tunnistautuminen laitteelle, joka ei kommunikoi muiden laitteiden kanssa, tai tunnistautuminen johonkin laitteeseen, joka on osana suurempaa verkkoa. Tällainen tunnistautuminen voidaan tehdä joko yksinkertaisena tai kaksivaiheisena autentikointina. Nämä tunnistautumistavat voidaan jakaa neljään kategoriaan:

1. Jotain mitä omistaa, esimerkiksi smart card
2. jotain mitä tietää, esimerkiksi salasana
3. jotain mitä on, esimerkiksi sormenjälki
4. jotain mitä tekee, esimerkiksi käsiala. (Vacca, 2004)(Adams et al., 2003)

Yksinkertaisessa tunnistautumisessa käytetään näistä vain yhtä tapaa. Tällainen voi olla esimerkiksi salasana tai sormenjälkitunnistus. Kaksivaiheisessa tunnistautumisessa on käytössä kaksi erilaista tunnistautumistapaa. Tämä voi olla esimerkiksi pankkikortti ja pin-koodi, jossa pankkikortti on jotain, mitä käyttäjä omistaa ja pin-koodi on jotain mitä käyttäjä tietää.

PKI:ta harvemmin käytetään paikalliseen tunnistautumiseen, sillä tästä ei saada kovinkaan suurta hyötyä. PKI:ta kuitenkin voidaan käyttää etätunnistautumiseen ja siinä sen hyvä puoli on se, että jos tunnistautumista salakuunnellaan, niin siitä ei saada mitään kriittistä informaatiota irti. Tämä johtuu siitä, että mitään arkaluonteista, kuten salasanoja, ei lähetetä verkon yli. PKI:ta voidaan myös käyttää datan oikeellisuuden tunnistamiseen. (Adams et al., 2003)

2.3.2 Kiistämättömyys

Datan kiistämättömyydellä tarkoitetaan sitä, että data pysyy muuttumattomana eli sitä ei ole muokattu lähetyksen jälkeen. (Adams, Adams & Lloyd, 2003) Tämä on tärkeää esimerkiksi maksutapahtumassa, jotta kumpikaan osapuoli ei voi väittää, että tapahtumaa ei olisi tapahtunut. (Vacca, 2004)

Julkisen avaimen infrastruktuurissa kiistämättömyys voidaan taata käyttämällä digitaalista allekirjoitusta. Tässä datasta lasketaan tiiviste eli hash ja tiiviste salataan varmenteen avulla. Tämän ansiosta dataan tehdyt muutokset näkyvät selvästi datassa,

sillä tiiviste on muuttunut. Datan saaja voi siis laskea saamastaan datasta tiivisteen ja verrata sitä lähettäjän laskemaan tiivisteeseen. Jos nämä ovat samat, niin dataa ei olla muokattu lähetyksessä ja jos ei, niin dataan ei voida enää luottaa. (Adams et al., 2003)

2.3.3 Luottamuksellisuus

Luottamuksellisuus on varmuus datan yksityisyydestä. Tällöin dataa voivat käsitellä ja lukea vain henkilöt, joilla on siihen oikeus. Luottamuksellisuutta tarvitaan, kun data on säilössä tai kun sitä liikutetaan verkon yli. (Adams et al., 2003)

Yksi luottamusta suojeleva protokolla on Transport Layer Security (TLS). Tämä suojaa dataa liikkeessä siten, että sitä ei voida lukea matkalla. Tätä käytetään esimerkiksi internet-selaimissa kirjautuessa sivustolle käyttäen salasanaa. Tämä salasana ei saa näkyä muille, joten luottamuksellisuus on taattava. (Perez, 2014 s.111-124) (Stapleton, 2016 c.3.2)

2.4 Käyttökohteet

PKI:n käyttökohteita ovat esimerkiksi TLS, Secure Multipurpose Internet Mail Extension (S/MIME), Virtual Private Network (VPN) ja Pretty Good Privacy (PGP). (Vacca, 2004) Näiden avulla tietoa voidaan suojata sen liikkuessa verkossa monissa erilaisissa käyttötapauksissa.

TLS:ää voidaan käyttää suojaamaan esimerkiksi autentikointia sekä välttämään kommunikoinnin välissä tapahtuvia hyökkäyksiä. Se käyttää julkisen avaimen infrastruktuuria, jossa käyttäjän selaimen tulee luottaa CA:han. Nykyään miljoonat internetin tapahtumat käyttävät TLS:ää suojaamaan liikennettä. Näitä voivat olla esimerkiksi pankkiliikenne, mikä on hyvin tietoturvakriittistä. (Khan et al., 2020)

S/MIME:ä käytetään salaamaan sähköpostiviestejä käyttäen PKI:ta. Tämän avulla viestijärjestelmät, kuten sähköpostiohjelmat voivat allekirjoittaa tai salata viestejä tai tehdä molempia yhtä aikaa. Tätä käyttämällä voidaan lähettää viestejä turvallisesti siten, että mahdollistetaan autentikoinnin, kiistämättömyyden sekä luottamuksen riittävä taso. (Vacca, 2004)

VPN on yksityinen virtuaalinen verkko, joka käyttää tunneli-protokollaa. VPN vaatii PKI:a tunnelin liitäntäpisteiden autentikointiin esimerkiksi käyttäjän ja yrityksen verkon välillä. VPN:ää käytetään yhdistämään käyttäjä verkkoon, jonka fyysinen sijainti on muualla, kuin käyttäjän luona. (Vacca, 2004)

Yleisesti sähköpostiviesteissä käytettyä PGP:tä käytetään salaamaan viestit. PGP:n käyttäjät hallitsevat itse avainten vaihdot sekä luottamuksen tasot, eikä erillisiä CA:ita tarvita. (Hunt, 2001)

2.5 Avainpari ja avainten säilytys

Julkisen avaimen infrastruktuuri käyttää asymmetristä salausta, jonka perustana on avainpari. Avainparissa on julkinen sekä yksityinen avain. PKI:n tapauksessa julkinen avain lähetetään varmenteen mukana ja yksityinen avain pidetään salassa. Julkisen avaimen voi myös säilyttää julkisessa paikassa kuten avoimessa tietokannassa. Täten kuka tahansa, joka haluaa julkisen avaimen haltijalle lähettää salattuja viestejä, voi hakea julkisen avaimen tietokannasta. (Adams et al., 2003) Esimerkiksi PGP-tapauksessa julkisia avaimia voidaan säilyttää julkisissa avainpalvelimissa.

Avainpari toimii siten, että toinen avain salaa dataa ja toinen avain pystyy purkamaan tämän salauksen. Avainparin toiminta perustuu siihen, että julkisesta avaimesta ei voida päätellä yksityistä avainta ja julkisella avaimella salattu viesti voidaan purkaa ainoastaan siihen liittyvällä yksityisellä avaimella. (Adams et al., 2003)

Avainten säilytys on kriittinen osa julkisen avaimen infrastruktuuria. Jos avaimia säilytetään väärin, ne voivat vaarantua esimerkiksi tietovuodon seurauksena.

2.5.1 Hardware Security Module

Hardware Security Module (HSM) on fyysinen laite, joka on suunniteltu hallitsemaan kryptografisia avaimia. Se voi siis esimerkiksi luoda julkisia ja yksityisiä avaimia ja säilöä näitä turvallisesti. (de Prisco et al., 2018 s.363-372)

HSM toimii siten, että ohjelma joka avaimia käyttää ei pääse niihin käsiksi. Avaimia käytetään API:n kautta, mikä keskustelee HSM:lle. Tästä syystä HSM on yleensä turvallisempi ratkaisu verrattuna avainten säilyttämiseen esimerkiksi tietokannassa. Näiden lisäksi HSM on varusteltu tavoilla suojata laitteen fyysistä turvallisuutta esimerkiksi sinetöidyllä pakkauksella tai erilaisilla sensoreilla. (de Prisco et al., 2018 s.363-372)

HSM-laitteet pystyvät usein takaamaan FIPS 140-2 level 3 suojan. (Google Cloud, n.d.-c) FIPS on standardi, jonka on kehittänyt National Institute of Standards and Technology (NIST). Tämä standardi käsittelee muun muassa salaukseen tarkoitettuja järjestelmiä. Level 3 on suojaluokitus, jonka saavuttamiseksi laitteen tulee olla fyysisesti

tietoturvallinen ja sen tulee käyttää identiteettiin perustuvaa autentikointia. (Annabelle et al., 2001)

2.5.2 Bring Your Own Key

Bring Your Own Key (BYOK) on tapa hallita kryptografisia avaimia. Tässä tavassa avaimia tarvitsevan sovelluksen kehittäjä käyttää omia kryptografisia avaimiaan kehitystyössä. Tämä tapa on käytössä pilvipalveluissa ja voi olla yhdistettynä Bring Your Own Encryption:iin (BYOE) Tässä tapauksessa pilvipalvelun käyttäjä hoitaa kryptografiset operaatiot omalla, tähän tarkoitetulla ohjelmistollaan. (Ulz et al., 2017)

Tämä ratkaisu takaa pilvipalvelun käyttäjälle täydet oikeudet omiin avaimiinsa ja hän voi luottaa omaan HSM:ään tai muuhun avaimien säilytysratkaisuun. Tämä voi luoda pilvipalvelun käyttäjälle turvallisuutta siitä, että avaimet ovat hänellä tallessa, eikä niitä tallenneta pilvipalveluun. (Ulz et al., 2017)

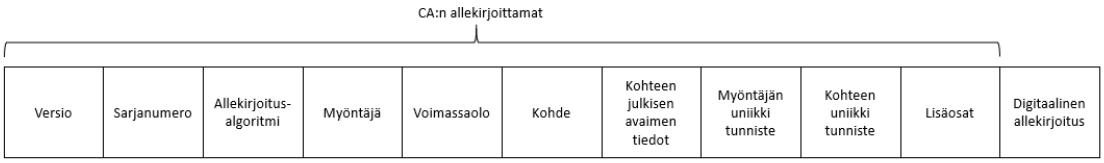
3. VARMENTEET

3.1 Varmenteiden sisältö

Luvussa 2.1 käsiteltiin sitä, että julkisen avaimen voi lähettää kenelle tahansa tai jakaa esimerkiksi julkisessa tietokannassa. Tässä kuitenkin on ongelmana se, että julkisen avaimen omistajuus pitää varmistaa. Tämän lisäksi tulee taata julkisen avaimen eheys. Tätä varten käytetään varmenteita. (Adams et al., 2003)

Varmenteita siis käytetään siihen, että voidaan sitoa tietty taho tiettyyn julkiseen avaimeen. Varmenteita voi olla erityyppisiä, esimerkiksi X.509-varmenne, Simple Public Key Infrastructure (SPKI)-varmenne, PGP-varmenne tai attribuuttivarmenne. Näiden varmenteiden rakenne eroaa toisistaan ja näiden varmennetyyppien sisäisissä versioissa saattaa olla eroja. (Adams et al., 2003)

Esimerkkinä varmenteiden sisällöstä käytetään X.509-varmenteen sisältöä, joka on kuvan 6 mukainen. X.509-varmenne määritellään RFC 5280-standardissa



Kuva 6. X.509-varmenteen sisältö. (Muokattu lähteestä Adams et al., 2003)

Versiolla (version) tarkoitetaan varmenteen versiota, X.509:n tapauksessa se on 1, 2 tai 3. Sarjanumero (serial number) on uniikki numero myöntäjälle, jolla varmenne voidaan tunnistaa. Allekirjoitusalgoritmilla (signature algorithm) nimetään algoritmi, jolla varmenne on allekirjoitettu. Tässä esitetään hash-algoritmi sekä salausalgoritmi. Esimerkiksi allekirjoitusalgoritmi voi varmenteessa olla sha256RSA. Tämä tarkoittaa sitä, että allekirjoitukseen käytetty hash-algoritmi on sha256 ja se on salattu RSA-algoritmilla.

Myöntäjä (issuer) on CA:n Distinguished Name (DN) eli CA:sta käytetty nimitys. DN sisältää muita kenttiä kuten Common Name (CN), Organization (O) ja Country (C). CN on CA:n yksilöivä nimi, O on varmenteen myöntävän organisaation nimi ja C kertoo valtion, missä CA toimii.

Voimassaolo (validity) on se aikaväli, jonka varmenne on voimassa. Tämä ilmaistaan kahdella kentällä; Valid from ja Valid to. Valid from-kenttä kertoo päivämäärän ja

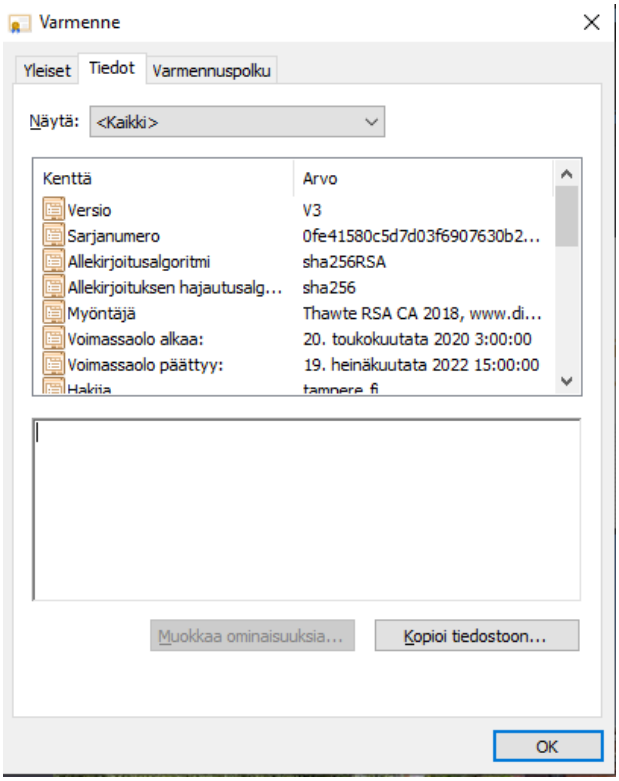
kellonajan, jolloin varmenne on tullut voimaan ja Valid to-kenttä kertoo päivämäärän ja kellonajan, jolloin varmenne vanhenee.

Kohde (subject name) on varmenteen omistajasta käytetty nimitys. Tämän sisällä käytetään myös nimitystä CN, mutta tässä kohtaa se viittaa varmenteen käyttäjään. Se voi olla esimerkiksi domain, jos kyseessä on verkkosivuilla oleva TLS-varmenne.

Kohteen julkisen avaimen tiedot (subject public key info) tai vain julkinen avain (Public key) on kohteen julkinen avain. Myöntäjän uniikki tunniste (unique identifier) ja kohteen uniikki tunniste ovat tunnisteita, joilla voidaan tunnistaa myöntäjä sekä kohde. (Adams et al., 2003) (Boyen et al., 2008)

Lisäosat ovat vapaaehtoisia kenttiä ja voivat olla esimerkiksi avaimen uniikki tunniste, avaimen käytön tunniste, Certificate Revocation List (CRL):n sijainti, yksityisen avaimen käyttöaika tai muita siihen liittyviä kenttiä. Lisäosia voi olla myös yksityisiä, jolloin ne ovat käytössä vain sille käyttötapaukselle. (Adams et al., 2003) (Boyen et al., 2008) Nämä kaikki kentät ovat CA:n allekirjoittamia, mikä suojaa varmenteen sisältöä muokkaamiselta, sillä muokkaaminen on tämän ansiosta helposti havaittavissa. (Vacca, 2004)

Kuvassa 7 on esimerkki varmenteen kentistä. Tämä varmenne on sama, kuin kuvassa 3 ja se on myönnetty sivustolle tampere.fi



Kuva 7. Esimerkki varmenteen sisällöstä sivustolta tampere.fi

3.2 Varmenteiden pyyntö ja myöntäminen

Varmenteiden pyyntö, myöntäminen, hallinta sekä käyttö perustuvat CA:n varmennepolitiikoihin. Nämä politiikat sisältävät yleiset tiedot varmenteen käytöstä, vastuut julkaisusta ja säilytyksestä, autentikointivaatimukset, vaatimukset varmenteen elinkaaresta, varmenteen hallinnan, turvallisuusvaatimukset, varmenteen, CRL:n ja OCSP:n profiilit, auditoinnin noudattamisen sekä muita tietoturvaan liittyviä asioita. Nämä ovat määritelty RFC 3647-viitekehyksessä. (Huang & Nicol, 2017 s.246-258)

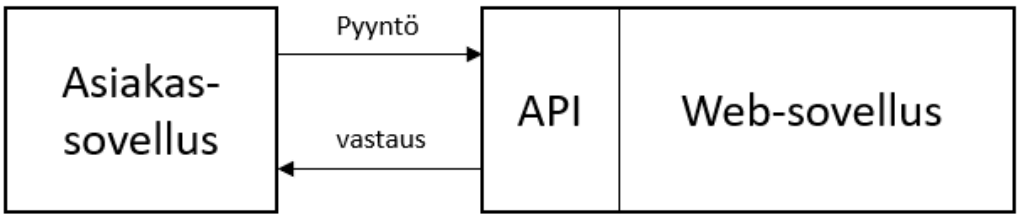
Luvussa 2.2 esitettiin, että usein varmenteiden haussa käyttäjän ja CA:n välillä on RA. RA toimii siis ”välittäjänä” ja tekee varmennepyynnöt käyttäjän puolesta. RA pyytää Varmenteita esimerkiksi web-lomakkeella, Representational State Transfer Application Programming Interface:lla (REST API) tai Certificate Management Protocol:lla (CMP), jotka toimivat yhteydessä CA:han. RA autentikoi käyttäjän, käyttäjä lähettää RA:lle varmennepyynnön joko web-lomakkeella tai API:n kautta. Tämän autentikoinnin onnistuessa ja varmennepyynnön hyväksymisen jälkeen RA lähettää CA:lle varmennepyynnön ja varmenne myönnetään. Tällainen RA-järjestelmä voi olla manuaalinen tai automaattinen. (Vacca, 2004)

Varmennepyyntöjä tehdään yleensä niin sanotuilla Certificate Signing Requesteilla (CSR). Yksi CSR-standardi on PKCS#10, mikä lähetetään tiedostomuodossa käyttäjältä CA:lle allekirjoitettavaksi esimerkiksi selaimen kautta. CA lähettää tämän allekirjoitetun varmenteen takaisin käyttäjälle. Tämä pyyntö sisältää nimen, julkisen avaimen sekä muita attribuutteja, joita on kuvattu luvussa 3.1. (Nystrom & Kaliski, 2000) PKCS#10 varmennepyyntö näyttää kuvan 8 mukaiselta.

```
-----BEGIN CERTIFICATE REQUEST-----
MIIDRjCCAi4CAQAwYjELMAkGA1UEBhMCVVMxEzARBgNVBAoMCkN1c3RvbWVyIFgx
HjAcBgNVBAsMFVRFU1QgQ0JTRCBDZXJ0aWZpY2F0ZTEeMBwGA1UEAwwVMTMyNDM1
NDY1OjAwMTI2MDQ0NjY0MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA
4FbnxmYRrDVlzRwfex5zkdw7vMpGAzVTt0BP3taHeR4YBQswh21egOnlCC3Msy+8
ptCLKZMZpDY4VCrjipemVUxb/hM2zIJzsHISIZ2CGG8QjfKVEvpLd3jURQr9sYa0
u4Icd9bpUgL0YcXeHzAHUGgnkdSbWlNbEKlT/WfwggwOHicAkmk41vxH7CTybTZV
g06D1sXPx94Pfdmtxm4AyOZuD5hpIjcw1+qBu8EhPAQGjYCKueez/LI5g/Cg+nBK
dEtv8m4UsVHERdMrBHNgKLJy0zVTuYRwZ6cALIY3bbaE1r6Mrez82Y5ED2i8U111
5+vL3HSwZYoSxVQ4owFdewIDAQABoIGeMIGbBgkqhkiG9w0BCQ4xgY0wgYowDgYD
VR0PAQH/BAQDAgWgMCYGA1UdIAQfMB0wDAYKKwYBBAGC7BECATANBgsrBgEEAYLs
EQEBAzBQBgNVHREESTBHoBoGCisGAQQBguwRAQSgDAwKRkNDSURfSEVSRaApBgor
BgEEAYLsEQEFoBsMGURFVklDRV9TRVJJQUxfTlVNQkVSX0hFUkUwDQYJKoZIhvcN
AQELBQADggEBAMdI/FDmM3NYvJQsgVxQ5I8yWZX+Rr7B+CdBHdmnjr6SxJQHusJu
pS/b9RXLg9uPVFXncq6oJk5HnCKEuSvH/Ku02mckFeRYn3MsU0rQ+JiL2uYz5bi5
WZqagyGqh+m/TTzvITPPrOnpt5r/sHQaWaiOo3I4MQ9GjRAxxToXkqgKXQfmaWky
mNQGRbT3MjfX3efkhPPZmnrSMw3rZhGyJIufBDLlE2+2Wz2TAWtSvtOKnzpqooAL
/IId15nQSyz97P/yEIoGgpe3SQv6AJKEvl0xaxeVcXBDZmufT7UyXtVr69Uj3fFL
IIcH0H8PkFRv1oS55kcrJlZORVc3tEMVr+Q=
-----END CERTIFICATE REQUEST-----
```


Kuva 8. Esimerkki PKCS#10 varmennepyynnöstä

Toinen tapa varmennepyyntöjen tekoon on (REST API). API on web-sovelluksen rajapinta, jonka kautta asiakas-sovellus voi keskustella tämän web-sovelluksen kanssa. API toimii siis web-sovelluksen ja asiakassovelluksen välissä kuvan 9 mukaisesti. Pyyntö sekä vastaus liikkuvat asiakas-sovelluksen ja web-sovelluksen välillä HTTP:n välityksellä. (Massé, 2011)



Kuva 9. API:n tominta. Muokattu lähteestä (Massé, 2011)

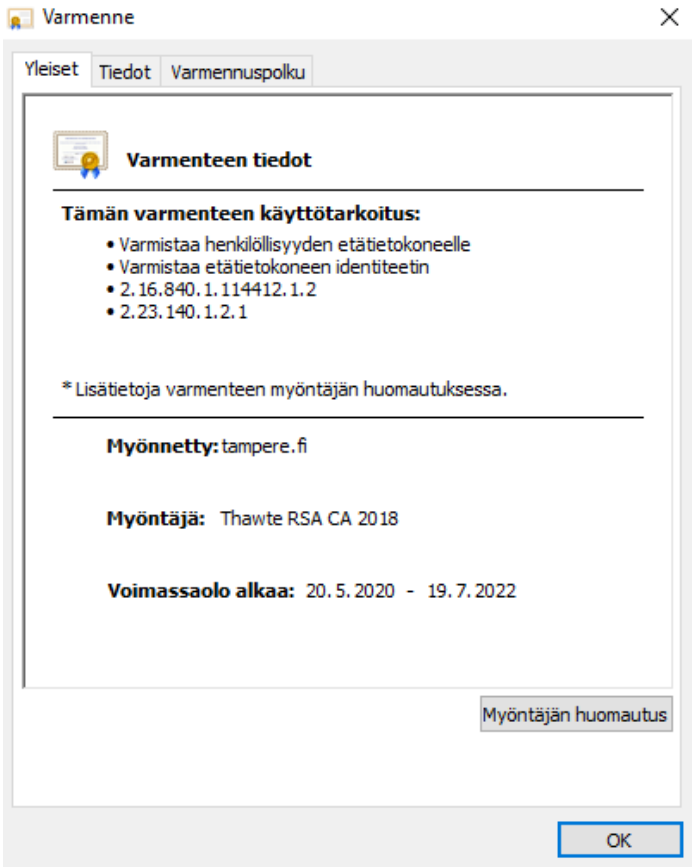
Kolmas tapa on käyttää CMP:tä. Tämä on protokolla, joka määritellään RFC 4210-standardissa. Se käyttää Certificate Request Message Format:ia (CRMF), joka on varmennepyyntöviesti ja se määritellään RFC 4211-standardissa. CMP:llä lähetetään varmennepyyntö CMRF-muodossa HTTP:n tai HTTPS:n yli. (IBM, n.d.) Esimerkiksi Instassa määrällisesti eniten varmennepyyntöjä tehdään REST API:lla tai CMP:llä.

REST on joukko arkkitehtuurisia rajoituksia. Tämä mahdollistaa keskustelun RESTful web-sovellusten kanssa http-protokollan välityksellä. REST:ssä olevien sääntöjen ansiosta se on nopea ja kevyt käyttää. (Red Hat, 2020)

Instan CA:lle tehdyt varmennepyynnöt tehdään joko manuaalisesti syöttämällä varmenteen tiedot erilliseen lomakkeeseen, PKCS#10 (CSR) tiedostoilla tai REST API:a käyttämällä. REST API:lla varmennepyynnöt sekä varmenteen hallinta voidaan toteuttaa automaattisesti tekemällä REST API-pyyntöjä esimerkiksi automatisoidun skriptin avulla. Instan CA:n käyttäjät pääasiassa käyttävät automatisoituja tapoja hallita varmenteita. Manuaaliset varmennepyynnöt ovat harvinaisempia, sillä suuren varmennemäärän hallitseminen manuaalisesti on aikaavievää.

3.3 Varmenteiden hallinta

Varmenteilla on aina tietty voimassaoloaika, jonka jälkeen varmenteeseen ei voida enää luottaa ja se sulkeutuu. Varmenteita voi myös joutua sulkemaan voimassaoloaikana, jos varmenne on esimerkiksi vaarantunut. Tällaiseen tilanteeseen voi johtaa esimerkiksi yksityisen avaimen vuotaminen käyttäjältä. (Vacca, 2004) Varmenteen voimassaoloaika on esitetty varmenteessa kuvassa 10.



Kuva 10. tampere.fi-sivustolle myönnetyn varmenteen yleiset tiedot.

Jos varmenteesta loppuu voimassaoloaika, voidaan sille tehdä jokin seuraavista kolmesta toimenpiteestä:

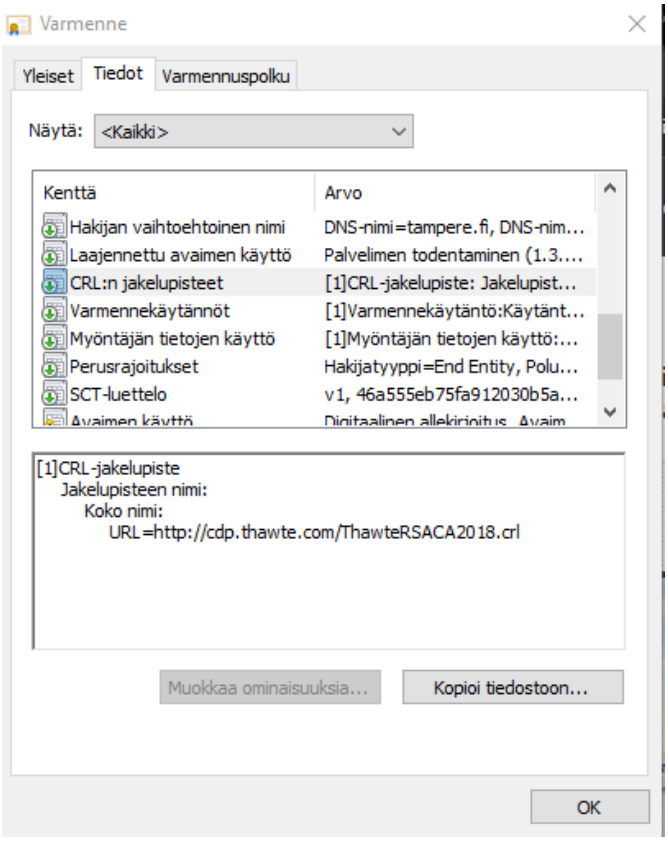
1. varmenne suljetaan, jos käyttäjä ei enää käytä kyseistä PKI-järjestelmää
2. varmenne uusitaan, jolloin käytetään samaa julkista avainta kuin aikaisemmin, mutta uudessa varmenteessa
3. varmenne voidaan päivittää, jolloin luodaan uusi avainpari uuteen varmenteeseen.

Varmenteen uusimista voidaan käyttää silloin, kun varmenteen käyttökohde ei ole muuttunut ja kun avainpari ei ole vaarantunut. Muissa tapauksissa varmenne pitää päivittää. (Adams et al., 2003)

Varmenteiden myöntäminen on myös toteutettu Installa automatisoidusti. Vanhentunut tai vanheneva varmenne uusitaan automaattisesti, jos siihen on tarve, käyttäen REST API-rajapintaa ja varmenteen sulkutiedon tarkastaminen tapahtuu sulkulistan tai OCSP:n kautta. CRL:ää käsitellään luvussa 3.3.1 ja OCSP:tä käsitellään luvussa 3.3.2.

3.3.1 Sulkulista

Certificate Revocation List (CRL) on lista, jossa listataan suljetut varmenteet. Tämä lista on CA:n allekirjoittama ja se on julkisesti saatavissa. Tämän allekirjoittanut CA on yleensä sama CA, joka on alun perin myöntänyt varmenteen. CRL:iä julkaistaan tietyin väliajoin ja seuraava julkaisuaika näkyy CRL:ssä. (Boyen et al., 2008) CRL:n hyvä puoli on se, että se sisältää kaikki suljetut varmenteet yhdessä paikassa, mutta sen huono puoli on se, että nämä listat voivat kasvaa todella suuriksi. Tästä johtuen niiden lataaminen voi viedä aikaa, ja kun käyttäjä haluaa tarkistaa varmenteen tilan, täytyy koko lista ladata. (Berbecaru et al., 2009) CRL:n osoite näkyy varmenteen tiedoissa kuvassa 11.



Kuva 11. CRL:n osoite varmenteen tiedoissa.

Tätä ongelmaa voidaan välttää julkaisemalla niin sanottuja Delta CRL-listoja (DCRL). Näitä voidaan julkaista useammin ja niistä voidaan nähdä pelkät muutokset edelliseen CRL:ään. Näiden lisäksi pitää kuitenkin julkaista alkuperäinen CRL, jota nämä muutokset koskevat. Tämän avulla käyttäjä voi ladata vain tämän ja tarkastaa varmenteen tilan alkuperäisen CRL:n julkaisun jälkeen. (Berbecaru et al., 2009)

Toinen ratkaisu suuriin CRL-tiedostoihin on jakaa tämä lista moneen pienempään listaan. Tämän jaon voi tehdä esimerkiksi myöntökuukauden mukaan. Tämän avulla käyttäjä voi ladata vain sen CRL:n, mikä vastaa hänen varmenteensa myöntöaikaa. (Berbecaru et al., 2009)

3.3.2 OCSP

CA on vastuussa siitä, että käytöstä poistettujen varmenteiden tilatieto on saatavilla. Tämä voidaan toteuttaa Online Certificate Status Protocol:n (OCSP) avulla. (Boyen et al., 2008) Tätä käyttäessä käyttäjä lähettää tilakyselyn OCSP-palvelulle. Tämä responder vastaa kyselyyn varmenteen tai varmenteiden tilalla. Vastaus on allekirjoitettu avaimella, joka kuuluu joko varmenteen myöntäneelle CA:lle, luotetulle responderille tai responderille, jolla on CA:n myöntämä luotettu varmenne. (Berbecaru et al., 2009)

OCSP-vastaus sisältää jonkin kolmesta vastauksesta: hyvä, suljettu tai ei tiedetty. Hyvä tarkoittaa, että varmenne ei ole suljettu ja sitä voidaan käyttää, suljettu tarkoittaa, että varmenne on suljettu, jolloin sitä ei voida käyttää ja ei tiedetty tarkoittaa, että OCSP ei tiedä varmenteen tilaa. Jos OCSP ei tiedä varmenteen tilaa, voidaan päätellä, että varmenne ei ole oikea. OCSP:tä käyttäessä käyttäjän tulee luottaa responderiin, kun taas CRL:iä käyttäessä käyttäjän tulee luottaa vain CA:han, joka julkaiseen listan. (Berbecaru et al., 2009)

4. PILVIPALVELUT

4.1 Yleistä pilvipalveluista

Pilvipalvelut tarjoavat tietojenkäsittelypalveluja internetin välityksellä. Pilvipalveluiden tarkoitus on se, että käyttäjän pääsee käsiksi tietotekniikkaan helposti ilman fyysistä kanssakäymistä palvelimen kanssa ja ilman kanssakäymistä palveluntarjoajan kanssa. (L. Chen et al., 2019 s.4)

Pilvipalveluita voi olla julkinen-, yksityinen- tai hybridipilvipalvelu. Julkinen pilvipalvelu tarkoittaa sitä, että tietojenkäsittelyä varten käytössä olevat resurssit ovat kolmannen osapuolen palvelinkeskuksessa ja nämä ovat pilvipalvelun toimittajan omistuksessa. Yksityinen pilvipalvelu on yleensä yrityksen omassa palvelinkeskuksessa ja se on yhdistetty yksityiseen verkkoon. Hybridipilvipalvelussa nämä kaksi yhdistyvät ja tässä tapauksessa molemmissa palvelinkeskuksissa olevat laitteet keskustelevat keskenään. (Microsoft Azure, n.d.-b) Tässä työssä käsiteltävät suuret pilvipalveluntarjoajat Azure, AWS sekä GCP tarjoavat pääasiassa julkista pilvipalvelua, mutta myös hybridiratkaisu on mahdollista toteuttaa.

Pilvipalveluita on kolmea tyyppiä: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) ja Software as a Service (SaaS). IaaS sisältää vain tietotekniikan perusosat, kuten tietoliikennetekniikat, tietokoneet sekä tallennustilat. Tässä tapauksessa käyttäjän pitää siis rakentaa oma palvelunsa näistä osista pilvipalvelun sisälle. Tämä tuo pilvipalveluun joustavuutta, sillä suurin osa asioista on käyttäjän toteutettavissa. PaaS mahdollistaa hieman helppokäyttöisemmän pilvipalvelun, sillä se sisältää tarvittavia osia, muun muassa käyttöjärjestelmän. Tämä siis mahdollistaa käyttäjän keskittymisen enemmän itse tekemiseen, eli esimerkiksi jonkin ohjelmistoratkaisun toteuttamisen. SaaS on valmis tuote, jota käyttäjä vain käyttää. Sen toteutus ja ylläpito on kokonaan pilvipalvelun tarjoajan vastuulla. Tällainen SaaS-palvelu on esimerkiksi sähköpostiohjelma. (Noor et al., 2013) Tässä diplomityössä keskitytään erityisesti IaaS- sekä PaaS-palveluihin, sillä SaaS-palveluihin ei käyttäjällä ole tarvetta toteuttaa omaa PKI-ratkaisuaan.

Tässä diplomityössä käsitellään kolmea suurinta pilvipalveluidentarjoajaa; Microsoft Azurea, Amazon Web Serviceä sekä Google Cloud Platformia. Kaikki näistä tarjoavat satoja erilaisia palveluita esimerkiksi web-sivujen luomiseen ja ylläpitoon sekä tietokantoihin. Tämän diplomityön kannalta olennaisia palveluita esitellään luvussa 4.2.

Microsoft Azure on Microsoftin tarjoama pilvipalvelu. Se tarjoaa paljon erilaisia palveluita, jotka sopivat niin pienille kuin suurillekin organisaatioille. Azuren avulla voidaan rakentaa ja julkaista ohjelmistoja lähes millä tahansa ohelmointikielellä tai työkalulla. Azuren palvelut on sijoitettu ympäri maapalloa ja ne ovat Microsoftin hallinnan alla. Ohjelmistoja voidaan rajata esimerkiksi laskentaan, tietoliikenteeseen, dataan tai sovelluksiin. (Tulloch, 2013)

Azuren tapaan myös Amazon Web Service (AWS) tarjoaa paljon erilaisia ohjelmistoratkaisuja, kuten verkkosivujen hostausta, datan tallennusta ja sovellusten hostausta. Myös AWS:ssä voidaan rakentaa ja julkaista erilaisia ohjelmistoratkaisuja, joita voidaan kehittää joustavasti erilaisilla työkaluilla. (AWS, n.d.-a)

Google Cloud Platform (GCP) tarjoaa myös paljon erilaisia palveluja ja näissä myös on nähtävissä runsaasti päällekkäisyyksiä muiden pilvipalveluiden kanssa. Sillä on myös toimintaa ympäri maapalloa ja GCP:lle on käytössä monia työkaluja liittyen muun muassa ohjelmistokehitykseen, IoT-tallennustilaan (Internet of Things), tietoliikenteeseen, tietoturvallisuuteen ja moniin muihin.

Kaikki kolme pilvipalveluntarjoajaa mahdollistavat IaaS-, PaaS-, ja SaaS-ratkaisuja. Kaikissa IaaS-ratkaisuja ovat virtuaalikoneet sekä datan tallennusratkaisut.

PaaS-ratkaisuja ovat Azuressa App Services, SQL databases, Cosmos DB. AWS:ssä AWS CodeDeploy sekä AWS Aurora. CodeDeploy on työkalu automatisoimaan ohjelmistokehitystä ja Aurora on SQL-tietokanta. GCP:ssä Google App Engine sekä Cloud Functions.

SaaS-ratkaisuja ovat Azuressa Outlook-sähköposti, -kalenteri sekä Office 365. AWS:ssä nämä ovat kolmannen osapuolen tekemiä, joita voi löytää AWS marketplacesta ja GCP:ssä G Suite-ohjelmat, kuten Gmail. (J. Chen, 2020)

Yllämainitut palvelut jokaisen pilvipalveluntarjoajan kohdalla ovat esimerkkejä ja todellisuudessa ratkaisuja on huomattavasti enemmän. Tämän työn kannalta ei kuitenkaan ole oleellista luetella jokaista mahdollista ratkaisua pilvipalveluiden sisällä. Oleellisia ratkaisuja ovat palvelut, joissa käytetään julkisen avaimen infrastruktuuria. Näitä palveluita käsitellään luvussa 4.2.

Azuren ja GCP:n tarjoamat ratkaisut ovat samankaltaisia ja molemmat tarjoavat ratkaisuja oman infrastruktuurin pystytykseen, kuin myös valmiisiin ratkaisuihin, mitkä ovat suoraan kuluttajien käytössä. Esimerkiksi Gmail sekä Outlook ovat todella suosittuja kuluttajaohjelmistoja. AWS taas eroaa näistä siinä, että sen tarjoamat ratkaisut painottuvat enemmän yrityksille ja ovat luonteeltaan teknisempiä.

Synergy Research Groupin keräämän datan mukaan Azure, AWS ja GCP kattavat yli puolet koko pilvipalvelumarkkinoista. AWS:llä on suurin markkinaosuus, joka oli Q3 2021 33%. Toisena on Azure, jolla oli markkinaosuutta 20% ja kolmantena QCP, jolla tämä oli 10%. Loput pilvipalveluiden markkinaosuuksista jakautuvat usealle pienemmälle yritykselle. (Synergy Research Group, 2021)

4.2 Pilvipalveluiden tarjoamat ratkaisut

Pilvipalvelut tarjoavat satoja erilaisia palveluita ja ratkaisuja. Ratkaisulla tarkoitetaan ongelmaan kehitettyä vastausta. Esimerkiksi tietokanta on ratkaisu siihen, miten web-sivun dataa voidaan tallentaa järkevästi. Kaikkin näihin ratkaisuihin ei ole tarvetta myöntää varmenteita ja useat näistä palveluista eivät käytä niitä. Osa pilvipalveluiden tarjoamista ratkaisuista ei sovellu Instan varmennepaveluihin vaikka niissä käytettäisiinkin varmenteita. Yksi esimerkki tästä on se, että Insta ei myönnä selainten luottamia TLS-varmenteita.

Tässä luvussa käydään läpi tämän työn osalta oleellisimmat ratkaisut. Nämä ratkaisut liittyvät avainten säilytykseen ja palveluihin, joissa voidaan hyödyntää julkisen avaimen infrastruktuuria.

4.2.1 Virtuaalikoneet

Virtuaalikone on ohjelmistokerros tietokoneen sisällä ja sitä voidaan käyttää tavallisen tietokoneen tavalla siten, että fyysinen tietokone sijaitsee fyysisesti muualla kuin käyttäjä. Virtuaalikoneita voi olla joko yhteen prosessiin suunniteltu kone tai kokonaista järjestelmää tukeva kone. Ensimmäinen näistä tekee vain yhtä prosessia ja se luodaan, kun prosessi aloitetaan, ja tuhotaan, kun prosessi loppuu. Jälkimmäisessä on käyttöjärjestelmä, kuten Windowsi tai Linux ja sillä voidaan suorittaa useampaa eri prosessia. (Smith & Nair, 2005)

Pilvipalvelut tarjoavat virtuaalikoneita, joita käyttäjä voi käyttää ilman, että hänen tarvitsee ostaa erikseen fyysisiä laitteistoja. Nämä antavat myös käyttäjälle paljon enemmän mahdollisuuksia toteuttaa omia tarpeita verrattuna pilvipalveluiden valiisiin ratkaisuihin. Pilvessä pyörivät virtuaalikoneet ovat myös helposti skaalautuvia, kun esimerkiksi laskentatehoa tai tallennustilaa tarvitaan enemmän. (Microsoft Azure, 2019b)

Virtuaalikoneissa julkisen avaimen infrastruktuuria voidaan hyödyntää monessa eri tilanteessa. Esimerkiksi varmenteella autentikointi voidaan toteuttaa virtuaalikoneen sisällä olevaan palveluun.

4.2.2 REST API

REST API:n avulla voidaan päästä käsiksi pilvipalveluihin. Sen avulla voidaan muun muassa hallita verkkopalveluita sekä pilvipalveluun tallennettua dataa. Esimerkiksi AWS API Gateway mahdollistaa API:en luonnin sekä niiden käytön. API:n luoja tekee API:n johonkin tiettyyn tarkoitukseen ja sen käyttäjä käyttää sen toiminnallisuuksia API-kutsujen avulla. (AWS, n.d.-e) API:en toiminta määritetään niiden politiikoissa. Näiden politiikkojen avulla voidaan määrittää esimerkiksi API-kutsujen määrää tai estää API-kutsut tietyltä IP:ltä. REST API on käytössä pilvipalveluissa sekä Instan varmennepalvelussa ja tästä syystä tässä työssä käsitellään sitä.

API:n käyttö pilvipalveluissa vaatii tunnistautumista. AWS:ssä tämä tunnistautuminen voidaan tehdä TLS-varmenteen avulla. API:n käyttäjällä tulee olla X.509-varmenne, joka on tässä tapauksessa TLS-protokollan käyttämä varmenne. Tällä varmenteella käyttäjä osoittaa henkilöllisyytensä ja pääsee käyttämään API:a. (AWS, n.d.-e)

4.2.3 Tiedostojen allekirjoitus

Luvussa 2.3.2 esiteltiin digitaalisen allekirjoituksen käsite. Tätä voidaan käyttää tiedostojen allekirjoitukseen. Allekirjoitettava tiedosto voi olla esimerkiksi PDF-tiedosto tai allekirjoitusta voidaan käyttää ohjelmiston allekirjoitukseen (code signing). PDF-tiedoston allekirjoittamisella voidaan varmistua tiedoston oikeellisuudesta ja tiedosto voidaan yhdistää tiettyyn henkilöön. Ohjelmiston allekirjoituksessa voidaan varmistua siitä, että allekirjoitettua ohjelmistoa ei olla muutettu sen allekirjoittamisen jälkeen. Allekirjoituksen validoinnilla voidaan varmistua siitä, että ohjelma on oikea, eikä esimerkiksi haittaohjelma. (Larramo, n.d.) (DocuSign, n.d.)

Azure käyttää tiedostojen allekirjoitukseen Microsoft SignTool:ia. Tällä työkalulla voidaan allekirjoittaa tiedostoja, todentaa allekirjoituksia sekä tehdä aikaleimoja tiedostoihin. Azure käyttää allekirjoitukseen pfx-tiedostoa, joka on säilössä Key Vaultissa. Key Vaultia käsitellään lisää luvussa 6.3.2. PFX-tiedosto on PKCS#12-tiedosto, mikä sisältää varmenteen sekä yksityisen avaimen. (Microsoft Azure, 2021d)

4.2.4 Internet of Things

IoT:ssä tietoteknisiä laitteita yhdistetään internetin välityksellä. Nämä laitteet ”keskustelevat” toistensa kanssa. Laitteiden väliset yhteydet ovat automatisoituja, eivätkä ne vaadi manuaalisia toimenpiteitä. IoT käyttää usein erilaisia sensoreita, mitkä viestivät dataa mikrokontrollerille. Tätä dataa taas käsitellään erilaisten sovellusalustojen avulla. Näiden avulla voidaan myös määritellä IoT-järjestelmälle sääntöjä. (Ramgir, 2019)

Azuressa IoT-palveluna on käytössä IoT hub. Tämän avulla voidaan luoda IoT-järjestelmiä, jotka ovat yhteydessä toisiinsa Azuren välityksellä. Tämän kautta laitteille voidaan asettaa varmenteita, joilla ne autentikoituvat IoT hub:iin. Tällä tavoin voidaan varmistaa se, että kaikki IoT-järjestelmässä olevat laitteet ovat oikeita, eikä järjestelmään pääse haitallisia laitteita. (Microsoft Azure, 2021c)

5. JULKISEN AVAIMEN INFRASTRUKTUURI PILVIPALVELUISSA

5.1 Pilvipalveluissa käytössä olevat varmenneratkaisut

5.1.1 Microsoft Azure

Azuressa julkisen avaimen infrastruktuurin implementointiin on kaksi ratkaisua: käyttää CA:ta, joka on yhteistyössä Azuren kanssa tai käyttää CA:ta, joka ei ole yhteistyössä Azuren kanssa. Jälkimmäistä tapausta käsitellään luvussa 7.2.1. Azuren kanssa yhteistyössä olevat CA:ta ovat DigiCert sekä GlobalSign. (Microsoft Azure, 2020a)

Yhteistyössä olevan CA:n kanssa varmenteen luominen tapahtuu siten, että Azuressa toimiva ohjelma luo avaimen Key Vaultiin. Tämän jälkeen Key Vault lähettää TLS/SSL-varmennepyynnön yhteistyö-CA:lle. Ohjelma lähettää kyselyitä CA:lle ja kun varmenne valmistuu, vastaa CA tällaiseen kyselyyn valmiilla TLS/SSL-varmenteella. (Microsoft Azure, 2020a)

Kun käytetään Azuren kanssa yhteistyössä olevaa CA:ta, niin Key Vault yrittää varmenteen uusimista automaattisesti, kun varmenne on vanhenemassa. Tämän lisäksi käyttäjä saa sähköposti-ilmoituksen uusimisesta. Varmennetta käyttävä Azuren hallinnoima ohjelma lähettää tietyn väliajoin kyselyä Key Vaultille siitä, onko varmenne päivitetty vai ei. Jos varmenne on päivitetty, se otetaan käyttöön ohjelmassa. (Microsoft Azure, 2016)

5.1.2 Amazon Web Services

Amazon Web Servicessä (AWS) on käytössä AWS Certificate Manager-palvelu, jonka tarkoitus on julkisten SSL/TLS X.509-varmenteiden hallinta sekä luonti. Tällä palvelulla voidaan luoda, vahvistaa, uusia, hallita sekä poistaa varmenteita. (AWS, n.d.-h)

Tämän lisäksi AWS:ssä voidaan käyttää yksityistä CA:ta. Tämän palvelun nimi on AWS Certificate Manager Private Certificate Authority. Tämä on CA, joka toimii pilvipalvelun sisällä ja sillä voidaan myöntää niin julkisia, kuin yksityisiä varmenteita. Tämän avulla voidaan luoda CA-hierarkia juurivarmentajasta alivarmentajiin. (AWS, n.d.-c) AWS:n yksityisellä CA:lla on mahdollista käyttää OCSP:ia. Tämän avulla osapuolet voivat tarkistaa varmenteen tilan ilman, että heidän tarvitsee ladata CRL:ää. AWS:n mukaan

tämä mahdollistaa varmenteiden tilan selvityksen tilanteissa, joissa kohteen tallennustila on rajallinen ja suurien CRL-tiedostojen lataaminen ei ole vaihtoehto. (AWS, 2021)

Instan järjestelmät vaativat kuitenkin tietoturvallista ratkaisua, johon voidaan varmasti luottaa eikä tätä voida taata AWS:ssä toimivalta CA:lta. AWS:llä on pääsy fyysisille laitteille, joissa yksityiset avaimet ovat tallennettuna ja tätä kautta pääsy on myös kaikkeen salattuun tietoon. CA tullaan siis tulevaisuudessa pitämään Instan omissa järjestelmissä ja tästä syystä tämä pilven sisällä toimiva CA ei ole vaihtoehto.

AWS:ssä on kuitenkin myös mahdollista käyttää varmenteita, joita ei ole luotu edellä mainitulla AWS:ssä toimivalla CA:lla. AWS Certificate manageriin voidaan tuoda varmenteita kolmannen osapuolen CA:lta. Certificate managerista varmenteita voidaan käyttää AWS:ssä käytössä oleviin muihin ohjelmiin. (AWS, n.d.-h) Tätä käsitellään lisää luvussa 7.

AWS:ssä julkisen avaimen x.509-varmenteita on käytössä ainakin kuudessa palvelussa; Amazon Api Gateway, AWS CloudFormation, Amazon CloudFront, Code Signing for AWS IoT, Elastic Beanstalk ja Elastic Load Balancing. Näissä palveluissa varmenteita käytetään joko SSL/TLS-varmenteina tai koodin allekirjoitusvarmenteina. (AWS, n.d.-g)

Esimerkiksi Amazon Api Gateway käyttää varmennetta tunnistautumiseen. Tähän voidaan käyttää AWS Certificate Managerin (ACM) omaa varmennetta tai ACM:ään tallennettua varmennetta. Jälkimmäinen voi joko olla sinne tuotu varmenne tai ACM:n yksityisen CA:n myöntämä. Tätä varten API:lle tulee asettaa domain-nimi ja varmennetta käytetään tunnistamaan, onko käyttäjällä oikeuksia siihen domainiin. Varmenteen käyttöä varten tulee luoda niin sanottu truststore. Tämän on lista luotetuista varmenteista, joilla on oikeudet API:n käyttöön. (AWS, n.d.-e)

5.1.3 Google Cloud Platform

Google Cloud Platform (GCP) tarjoaa Certificate Authority Service-nimistä palvelua. Tähän palveluun voidaan luoda yksityinen CA, jonka avulla voidaan myöntää varmenteita. Tämä on nopea tapa luoda CA, joka toimii GCP:n sisällä. Tässä tapauksessa päädytään samaan ongelmaan, kuin AWS:n tilanteessa, sillä CA halutaan pitää Instan tiloissa. (Google Cloud, n.d.-a) Muista kolmannen osapuolen CA:n ratkaisuista GCP:n osalta käsitellään luvussa 7.

Certificate Authority Servicen sisällä voidaan luoda CA-hierarkioita juurivarmentajaan asti. Alivarmentajia voi myös olla hierarkiassa useampia erilaisia käyttötarkoituksia

varten. CA:lle voidaan luoda varmennepolitiikkoja ja näiden avulla voidaan esimerkiksi rajoittaa CA myöntämään vain tietynlaisia varmenteita. Näiltä CA:ilta voi pyytää varmenteita, varmenteita voidaan poistaa käytöstä sekä käytössä olevia varmenteita voidaan listata sekä tarkastella. Varmenteen tilaa voidaan myös tarkastella OCSP:n avulla. Certificate Authority Service mahdollistaa myös varmenteiden sekä avainten säilytyksen HSM:ssä. (Google Cloud, n.d.-d)

Certificate Authority Servicen voi myös yhdistää muutaman muun palveluntarjoajan palveluun ja täten tehdä varmenteiden hallinnasta joustavampaa. Näitä on muun muassa Hashicorp Vault, Terraform sekä Cert-Manager. Hashicorp Vaultia käytetään salaisuuksien säilyttämiseen ja sitä voidaan käyttää proxynä siten, että se välittää varmennepyynnöt Certificate Authority Serviceen. Terraformin avulla voidaan kirjoittaa koodia, jolla voidaan hallita Certificate Authority Serviceä. Tämän avulla voidaan esimerkiksi luoda CA:t, luoda CSR-tiedostot sekä julkaista uusia varmenteita luoduilla CA:illa. Cert-Manager on palvelu, jonka avulla voidaan hallita varmenteita Kuberneteksessä. (Google Cloud, n.d.-e)

6. AVAINTEN SÄILYTYS PILVIPALVELUJA VARTEN

Kryptografisten avainten säilytys vaatii hyvää fyysistä turvallisuutta sekä tietoturvallisuutta. Avainten turvallisuuden vaarantuminen saattaa pahimmassa tapauksessa aiheuttaa avainten väärinkäyttöä tai ainakin avaimiin liittyvän prosessin luottamuksen menettämistä.

Tässä luvussa käsitellään avainten säilytystä siten, että niitä voidaan hyödyntää pilvipalveluissa. Avaimia voidaan säilyttää suoraan pilvipalveluissa tai niitä voidaan säilyttää muualla ja käyttää niitä pilvipalveluissa. Luvussa 6.3 esitetään avainten säilytysratkaisuja eri toimijoilla sekä vertaillaan ratkaisuja turvallisuuden näkökulmasta.

6.1 Avainten säilytys

Avainten säilytyksen turvallisuusvaatimukset voidaan jakaa kolmeen kategoriaan; fyysinen turvallisuus, tietoturvallisuus sekä käytön turvallisuus. Fyysisen turvallisuuden suojausmahdollisuuksia ovat esimerkiksi lukkiutuvat ovet sekä valvontakamerat. Tämän tarkoitus on estää luvaton pääsy avaimien säilytysratkaisujen luo. Tietoturvallisuus kattaa avainten suojaamisen virtuaalisesti, esimerkiksi salaamalla ne sekä suojaamalla käytettyjen fyysisten laitteiden ohjelmistot tietoturva-aukoilta. Käytön turvallisuus sisältää esimerkiksi käyttöoikeudet siten, että vain tietyillä henkilöillä on mahdollisuus päästä avaimiin käsiksi. (Turner, 2016)

Tässä luvussa käsitellään avainten säilytystä KMS-sovelluksessa ja HSM-laitteessa. Molemmat näistä mahdollistavat avainten säilytyksen lisäksi niiden hallinnan, esimerkiksi luonnin sekä uusinnan. Avainten käyttöön sekä säilytykseen liittyen KMS- sekä HSM-järjestelmissä voidaan usein luoda politiikkoja, joiden tarkoitus on taata avainten luottamuksellisuus sekä niiden hallinnan autentikointi. (Turner, 2016) Nämä mahdollistavat sen, että avaimet pysyvät muuttumattomina eikä niitä päästä lataamaan tai katsomaan luvattomasti.

Varsinkin suurissa organisaatioissa kryptografisten avainten säilytys on vaikeaa toteuttaa ilman erillistä järjestelmää. Tätä varten on olemassa avaintenhallintajärjestelmiä (Key Management System), joiden avulla avaimia voidaan säilyttää erillisen järjestelmän kautta. (Kuzminykh et al., 2020) Avaintenhallintajärjestelmän tarkoitus on pitää avaimet keskitetyssä paikassa salattuna.

Ilman tällaista järjestelmää avaimet voivat joutua vääriin käsiin esimerkiksi huolimattomuuden takia.

6.2 Haasteet avainten säilytyksessä

Avainten säilytys pilvipalveluja varten voidaan jakaa kahteen kategoriaan: Avainten säilytys On-Premise-järjestelmissä tai kolmannen osapuolen järjestelmissä. Kolmannen osapuolen järjestelmät voivat olla joko pilvipalvelu, joissa avaimia käytetään, tai täysin erillinen toimija.

Avainten säilytyksen suurin haaste on se, että jos yksityinen avain päätyy säilytysratkaisun ulkopuolelle salaamattomana, voi sen haltuun saanut taho käyttää sitä väärin. Yksityisellä avaimella voidaan purkaa salaus ja saada haltuun tietoa ilman, että salatun tiedon hallitsija tietää tätä.

Avainten säilytykseen kolmannen osapuolen järjestelmissä sisältyy riskejä. Avaimet ovat tällöin toisen organisaation hallussa ja niiden käyttöä ei voida itse hallita. Avaimien säilytyksessä ja käytössä täytyy luottaa täysin ulkopuoliseen organisaatioon ja heidän lupaukseen siinä, että he eivät lataa avaimia omista järjestelmistään. Jos kyseessä on avaimia, joiden tietoturvallisuuden vaarantumisella on vakavia seurauksia, täytyy arvioida voidaanko jättää niiden turvallisuus pelkän luottamuksen varaan.

Usein nämä ulkopuoliset toimijat, jotka avaimia säilyttävät, ovat suuria organisaatioita, joiden päätoiminen paikka on Yhdysvalloissa. Tämä vaikeuttaa luottamuksen saamista, sillä näiden toimijoiden ratkaisuja on lähes mahdotonta auditoida suomalaisen yrityksen toimesta. Myöskään muille pilvipalvelun auditoinneille ei voida asettaa vaatimuksia. Tämä asettaa jo rajoituksia tietyissä tapauksissa, joissa avainten säilytysratkaisuihin vaaditaan tiettyä turvallisuutta ja tämä pitää auditoida.

Tässä luvussa mainittujen haasteiden takia pilvipalveluiden avaintenhallintaratkaisut ovat rajoittuneet niiden käyttötarkoituksesta. Nämä ratkaisut eivät sovi korkeimman luokan turvallisuusratkaisuihin, sillä luottamusta ei voi olla näin paljon.

6.3 Vaihtoehtoiset ratkaisut avainten säilyttämiseen pilvipalveluja varten

Avaimia voidaan säilyttää omissa tiloissa esimerkiksi yrityksen omassa palvelinkeskuksessa, erillisten palveluntarjoajien tarjoamissa palveluissa tai pilvipalveluiden sisällä olevissa palveluissa. Tässä luvussa käsitellään muutamaa vaihtoehtoa avainten säilytykseen siten, että niitä voidaan käyttää pilvipalveluissa.

6.3.1 On-Premise

On-Premisellä tarkoitetaan sitä, että avaimen säilytysratkaisut ovat organisaation omissa tiloissa, esimerkiksi omassa palvelinkeskuksessa. KMS on tässä tapauksessa esimerkiksi erillinen ohjelmisto, joka toimii palvelimella ja HSM on erillinen fyysinen laite yrityksen tiloissa. Tällaisessa tapauksessa näiden asennuksesta sekä konfiguroinnista vastaa organisaatio itse.

On-Premisen etuna on mahdollisuus varmistua siitä, että avainten säilytysratkaisut ovat varmasti turvallisia. Tätä ei yleensä voida taata, kun avaimet viedään erillisille palveluntarjoajille. Kun avaimet ovat tallessa yrityksen omilla laitteilla, voidaan itse luoda niille riittävä turvallisuus. Tällaisessa tapauksessa usein käytetään kuitenkin kolmannen osapuolen laitteita, mutta nämäkin ovat usein itse konfiguroitavissa. Fyysinen turvallisuus voidaan taata esimerkiksi tekemällä palvelinsalit turvallisiksi, kahdentamalla säilytysratkaisut sekä tekemällä riittävät varmuuskopiot.

On-Premise-avainten säilytyksessä voidaan hyödyntää BYOK-periaatetta, kun avaimia käytetään pilvipalveluissa. Tällöin avaimet ovat vain käyttäjällä, eivätkä pilvipalvelun tarjoajat pääse niihin käsiksi. Tämä parantaa turvallisuutta avainten käsittelyssä, eikä erillisiin palveluntarjoajiin tarvitse luottaa pitämään niitä riittävässä suojauksessa. On-Premise-ratkaisun etuna on myös se, että avaimia voidaan helposti käyttää myös pilvipalveluiden ulkopuolella.

6.3.2 Azure

Azuressa käytössä oleva KMS on nimeltään Key Vault. Tämän tarkoitus on säilyttää salaisuuksia kuten API-avaimia, salasanoja, varmenteita sekä kryptografisia avaimia. Key Vaultilla voidaan luoda avaimia tai sinne voidaan tuoda avaimia, poistaa avaimia käytöstä, antaa käyttäjille oikeuksia avainten käyttöön ja valvoa sekä konfiguroida avainten käyttöä. (Microsoft Azure, 2019a) Key Vaultin toimintaa osana julkisen avaimen infrastruktuuria on esitelty luvussa 7.2.1.

Key Vault vaatii autentikointia sekä käyttöoikeuksien käyttöä. Autentikoinnilla määritetään kuka palvelua pääsee käyttämään ja käyttöoikeuksilla määritellään mitä operaatioita käyttäjä voi suorittaa. Autentikointiin käytetään Azure Active Directorya ja käyttöoikeuksiin Azure role-based access controlia. Azuren Key Vault voi olla joko ohjelmistolla suojattu tai se voidaan suojata HSM-laitteella. Azuren mukaan Microsoft ei näe eikä voi ottaa dataa Key Vaultin sisältä. (Microsoft Azure, 2021a) Tätä ei voida kuitenkaan varmistaa mitenkään.

Azure tarjoaa käyttäjilleen dedikoituja HSM-laitteita. Näiden laitteiden avulla avainten säilytykseen voidaan saada FIPS 140-2 level 3 -tasoinen suojaus, jolloin niitä voidaan käyttää kryptografisten avainten säilytykseen. Nämä laitteet on sijoitettu maailmanlaajuisesti useaan eri paikkaan ja näin voidaan varmistaa niiden käytön jatkuvuus suurissakin ongelmatilanteissa. (Microsoft Azure, 2020b)

Azuren mukaan asiakkaan on mahdollista saada HSM-laitteisiin järjestelmänvalvojan yksinoikeudet ja Microsoftilla ei ole muita oikeuksia laitteeseen kuin sen monitorointi. Tämänkin voi poistaa käytöstä, mutta tällaisissa tapauksissa laitteen fyysistä kuntoa, kuten lämpötilaa ei monitoroida ja laite voi hajota. (Microsoft Azure, 2021b)

Azuren Key Vaultissa olevia avaimia voidaan hyödyntää esimerkiksi datan salauksessa sekä Azure App Servicessä. Datan salauksessa on mahdollista salata Azuren tallennustilassa olevaa dataa tai erilaisia tietokantoja. Tämän avulla voidaan salata yksittäisiä tiedostoja tai kokonaisia levyjä tai tietokantoja. App Servicen tapauksessa Key Vaultiin voidaan säilöä TLS/SSL:n käyttöön liittyviä asymmetrisiä avaimia. (Microsoft Azure, 2021a)

6.3.3 AWS

AWS:ssä on käytössä Key Management Service (KMS) avainten luontiin sekä säilytykseen. Tätä voidaan käyttää datan suojaamiseen liikkeessä sekä levossa. Avaimia voidaan myös poistaa käytöstä tai poistaa kokonaan. (Kanikathottu, 2020) AWS:n mukaan avaimet eivät koskaan liiku KMS:n ulkopuolella salaamattomina ja niitä varten voidaan luoda politiikkoja, jotka määrittelevät esimerkiksi sen, kuka avaimiin pääsee käsiksi. Tämän lisäksi avainten käyttöä on mahdollista lokittaa. (AWS, n.d.-i)

AWS:n KMS:ään voidaan joko luoda avaimia tai sinne voidaan tuoda omia avaimia (BYOK). Avainten tuonti rajoittuu kuitenkin vain symmetrisiin avaimiin ja asymmetristen avainten tuonti ei ole mahdollista. KMS:ssä olevia avaimia voidaan hyödyntää monissa AWS:n tarjoamissa palveluissa. Näitä ovat muun muassa datan salaaminen eli sen suojaus levossa tai allekirjoitus avainparilla. (AWS, n.d.-d)

AWS KMS käyttää HSM-laitteita, jotka toteuttavat FIPS 140-2 -tason suojausta. Tässä erona Azureen on se, että Azuren Key Vaultissa pitää erikseen määritellä HSM:n käyttö avainten säilytyksessä. AWS:ssä tämä toteutuu aina. Tässä tapauksessa kuitenkin yhtä HSM-laitetta käyttää useampi käyttäjä, eikä KMS-järjestelmällä ole omaa laitetta. Myös FIPS 140-2 suojaus on tasoa 2, eikä 3. (AWS, n.d.-c)

Jos AWS:llä halutaan käyttöön yksityinen HSM-laite sekä FIPS 140-2 level 3-tason suojaus, on mahdollista käyttää CloudHSM:ia. Tässä tapauksessa HSM on vain käyttäjän omassa käytössä ja tämän hallinta on käyttäjän vastuulla. Tämä luo käyttäjälle enemmän kontrollia sekä turvallisuutta, sillä AWS ei osallistu laitteen hallintaan. Tätä CloudHSM:a voidaan käyttää muun muassa yksityisen SSL/TLS-avaimen säilytykseen ja CA:n yksityisen avaimen säilytykseen sekä allekirjoitusten tekoon. (AWS, n.d.-c)

6.3.4 GCP

Google Cloud Platformissa avainten säilytystä varten on Cloud Key Management. Tänne pystytään tallentamaan symmetrisiä sekä asymmetrisiä kryptografisia avaimia ja niitä voidaan luoda sekä tuhota tämän avulla. Tässä palvelussa on mahdollista myös säilyttää avaimia HSM-laitteissa. Tämän lisäksi GCP tarjoaa mahdollisuutta säilyttää avaimia GCP:n kumppanien KMS-järjestelmissä. Näitä on muun muassa Fortanix, Thales ja Unbound Tech. (Google Cloud, n.d.-b)

GCP:ssä pystytään myös käyttämään BYOK-periaatetta, jos avaimia ei haluta säilyttää pilvipalvelussa. Avaimia voi säilyttää kolmannen osapuolen järjestelmissä, jotka voivat olla esimerkiksi GCP:tä käyttävän organisaation hallinnassa. Näitä avaimia voidaan kuitenkin käyttää pilvipalvelussa ja GCP mahdollistaa näkyvyyden niiden käytölle siten, että käyttäjällä on tieto siitä, kuka ja missä avaimia käyttää. (Google Cloud, n.d.-b) Tämä kuitenkin perustuu luottamukseen siitä, että GCP kertoo kaikki avainten käyttöön liittyvät asiat. Ei ole kuitenkaan mahdollista varmistua siitä, että GCP kertoo varmasti kaikki tapahtumat, joissa avaimia on käytetty.

GCP:n HSM-laitteilla on myös mahdollista saavuttaa FIPS 140-2 Level 3 -suojaus. Tämä HSM toimii pilvessä ja sen hallinnasta vastaa GCP. GCP mahdollistaa myös yksityisen HSM:n käytön, jolloin HSM on varattu vain sen käyttäjälle. Tässä tapauksessa käyttäjän tulee kuitenkin hankkia HSM-laitteet itse ja lähettää ne Googlen palvelinsaleihin, jossa Google konfiguroi niihin esimerkiksi internet-yhteyden. Tässä tapauksessa käyttäjällä on mahdollisuus valita hänen tarpeisiinsa sopiva laite ja tämän laitteen turvallisuus voidaan taata paremmin. Käyttäjää vaaditaan myös tässä tilanteessa itse hallitsemaan ja ylläpitämään HSM-laitetta. Googlen mukaan heillä ei ole laitteeseen mitään kontrollia eikä pääsyä avaimiin. (Google Cloud, n.d.-c) Myöskään tätä ei voida varmistaa mitenkään.

6.3.5 Hashicorp Vault

Esimerkki avaintenhallintaratkaisusta pilvipalveluiden ja On-Premisen ulkopuolella on Hashicorpin Vault. Tällä työkalulla on mahdollista säilöä avaimia, salata dataa sekä toteuttaa käyttöoikeuksien hallintaa. Vaultilla on mahdollista säilöä salaisuuksia, kuten API-avaimia, salasanoja tai varmenteita. Vaultin dokumentaation mukaan avaimet saadaan tallennettua turvallisesti ja nämä ratkaisut voidaan auditoida. (HashiCorp, n.d.-a)

Vaultiin voidaan yhdistää niin sanottu Audit Device. Tämä on erillinen komponentti, jonka avulla voidaan pitää lokia siitä, kuka Vaultia on käyttänyt. HashiCorpin mukaan tämä komponentti tallentaa jokaisen pyynnön ja vastauksen, mitä Vaultiin tehdään. Tämä mahdollistaisi siis parempaa luottamusta avainten säilytykseen. Tässä tapauksessa lokien tallennus tulisi kuitenkin pystyä vahvistamaan siten, että tiedetään, että kaikki liikenne varmasti tallentuu lokeihin. (HashiCorp, n.d.-a)

Vaultilla voidaan käyttää Secrets Enginejä. Nämä ovat komponentteja, jotka säilyttävät, luovat sekä salaavat dataa. Näiden toiminnot voivat olla yksinkertaisia, esimerkiksi niihin voidaan tallentaa dataa ja lukea sitä, mutta ne voivat myös esimerkiksi salata dataa tai hallita varmenteita. Näitä Secret Enginejä on tarjolla AWS:een, Azureen sekä GCP:iin. Ne jaetaan autentikointiin liittyviin operaatioihin sekä avainten hallintaan. (HashiCorp, n.d.-a)

AWS, Azure sekä GCP Secrets engine luovat autentikointiin käytettäviä tunnuksia politiikkojen mukaan. Tämä HashiCorpin mukaan helpottaa tunnusten hallintaa, kun tunnukset ovat tallennettuna Vaultiin. Tämän avulla voidaan myös luoda käyttäjäryhmiä ja Vaultin oikeuksia näiden käytössä voidaan hallita politiikkojen avulla. (HashiCorp, n.d.-a)

Secret Enginen avulla avaimia voidaan hallita keskitetysti yhdestä paikasta kuitenkin siten, että kopio avaimista on Secret Enginessä, mutta avaimet ovat säilytyksessä pilvipalveluiden KMS-järjestelmissä. Tämän avulla avainten elinkaaren hallinta on mahdollista toteuttaa Azuren Key Vaultissa, AWS KMS:ssä sekä GCP Cloud KMS:ssä olevien avainten osalta. (HashiCorp, n.d.-a)

Key Vaultin tarkoitus on helpottaa avainten elinkaaren hallintaa tilanteissa, joissa avaimia on monessa eri paikassa. Käyttäjällä voi olla avaimia jokaisessa kolmesta pilvipalvelun tarjoajasta ja tämän avulla näitä kaikkia voidaan hallita suoraan Vaultista.

7. PILVIPALVELUN MAHDOLLISUUDET KOLMANNEN OSAPUOLEN VARMENNEPALVELULLE

7.1 Instan varmennepalvelu pilvipalveluissa

Yksi vaihtoehto kolmannen osapuolen CA:n käyttöön pilvipalveluissa on viedä tämä CA pilvipalveluun. CA-järjestelmä voisi tässä tapauksessa pyöriä esimerkiksi virtuaalikoneella pilvipalvelun sisällä ja se voisi myöntää täältä varmenteita muihin pilvipalvelussa oleviin ohjelmiin.

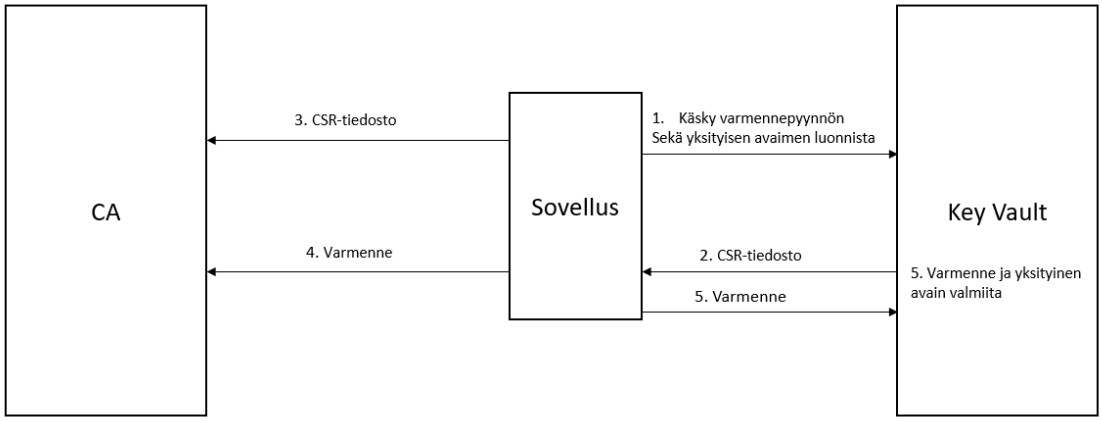
Luvussa 6.2 kerrottiin avainten hallinnan haasteista pilvipalveluissa. Myös tässä tapauksessa on olemassa samat haasteet ja CA:n käyttö perustuu hyvin vahvasti pilvipalveluihin luottamiseen. Instan varmennepalvelu on korkean turvatason ratkaisu ja siinä on huomioitu vahvasti muun muassa fyysinen turvallisuus. Tässä tapauksessa ei kuitenkaan voida taata pilvipalvelun turvallisuutta ja tästä syystä tämä ratkaisu ei ole toimiva.

7.2 Varmenteiden vieminen kolmannen osapuolen CA:lta pilvipalveluihin

Varmenteiden käyttö jakautuu pilvipalveluissa usein kolmeen eri vaihtoehtoon luvussa 5.1 esitellyllä tavalla. Varmenteita voidaan joko myötää pilven sisällä toimivasta yksityisestä varmentajasta, pilvipalveluiden kanssa yhteistyössä olevilta varmentajilta, tai kolmannen osapuolen varmentajalta. Tässä luvussa käsitellään näistä kolmatta vaihtoehtoa ja muut vaihtoehdot on esitetty luvussa 5.

7.2.1 Microsoft Azure

Luvussa 6.3.2 käsiteltiin Azuren Key Vaultia, mikä on käytössä varmenneprosessissa varmenteiden säilömiseen sekä luontiin.



Kuva 12. Varmenteiden käyttö Azuressa. (Muokattu lähteestä Microsoft Azure, 2020a)

Azuressa oleva varmenteiden luonti sekä niiden lähetys kolmannen osapuolen CA:lle allekirjoitettavaksi on esitetty kuvassa 12. Ensimmäisessä vaiheessa sovellus läettää Key Vaultille käskyn aloittaa varmenteen luonti. Tässä kohdassa Key Vault luo yksityisen avaimen sekä varmennepyynnön. Key Vault lähettää sovellukselle varmenteen allekirjoituspyynnön CSR-muodossa (Certificate Signing Request) vaiheessa 2 ja sovellus lähettää tämän CSR-pyynnön vaiheessa 3 CA:lle. CA allekirjoittaa varmenteen ja lähettää sen sovellukselle X509-muotoisena varmenteena vaiheessa 4. Vaiheessa 5 Key Vaultissa on yksityinen avain sekä varmenne käyttövalmiina. (Microsoft Azure, 2020a)

Kolmannen osapuolen varmenteiden luomista tai uusimista Key Vaultissa ei ole mahdollista automatisoida. Uusimisen tapauksessa on mahdollista saada sähköposti-ilmoitus, kun varmenteen vanheneminen lähenee. Uusiminen täytyy tehdä manuaalisesti siten, että Key Vault tekee uuden CSR-tiedoston ja tämä lähetetään manuaalisesti CA:lle. CA myöntää tästä varmenteen ja varmenne pitää viedä Key Vaultiin. (Microsoft Azure, 2020b)

Varmenne, joka on viety Key Vaultiin voidaan viedä edelleen Azuren tarjoamiin muihin palveluihin. Yksi tällainen palvelu on virtuaalikoneet, missä varmennetta ja yksityistä avainta voidaan käyttää samoihin tarkoituksiin kuin tavallisessa tietokoneessa tai palvelimessa. Näitä käyttökohteita on esimerkiksi tietokantasalaus ja erilaiset proxy-toteutukset. Azuren App Serviceen on myös mahdollista käyttää TLS-varmennetta ja

tässä on vaihtoehtona yhdistää Key Vault siihen tai ladata varmenne suoraan sovellukseen App Servicen kautta.

Tämä prosessi Azuressa käyttäen CA:ta, mikä ei ole yhteistyössä Azuren kanssa on siis täysin manuaalinen. Tämä aiheuttaa ongelmia tilanteissa, joissa varmenteita on käytössä suuria määriä. Tällä hetkellä Instan CA-palveluiden käyttäjät käyttävät esimerkiksi REST API:a varmenteiden pyyntöön sekä uusimiseen, joten manuaalinen varmenteiden hallinta ei sovellu Instan käyttötarkoitukseen järkevästi. Tästä syystä voidaan todeta, että Azuren pilvipalvelun sisällä olevat ratkaisut eivät sovellu Instan tarpeisiin. Tähän on kuitenkin esitetty ratkaisu luvussa 7.3, jonka avulla varmenteita voidaan viedä pilvipalveluun REST API-rajapinnan kautta.

7.2.2 Amazon Web Sevices

AWS Certificate Manageriin voidaan tuoda varmenteita, jotka ovat kolmannen osapuolen myöntämiä. Näitä varmenteita voidaan käyttää palveluissa, jotka ovat kytköksissä AWS Certificate Manageriin. Näitä palveluja on lueteltu luvussa 5.1.2. Tässä tapauksessa varmenteen uusiminen tapahtuu tuomalla Certificate Manageriin uusi varmenne entisen vanhetessa. (AWS, n.d.-f)

AWS:ssä varmenteen tuominen täytyy tehdä manuaalisesti käyttäen joko AWS Management Consolia tai AWS CLI:tä. Varmenne täytyy myös uusia täysin manuaalisesti tuomalla uusi varmenne palveluun. Azuren tavalla AWS ei siis myöskään mahdollista varmenteiden automaattista uusintaa, mikä tekee varmenteiden hallinnasta työlästä tilanteissa, joissa varmenteita on käytössä useita. Tähän on kuitenkin mahdollista tehdä CloudWatch-tapahtuma, joka ilmoittaa vanhenevasta varmenteesta. Varmenteen voi kuitenkin uusia tuomalla manuaalisesti uuden varmenteen, jolloin uusi varmenne korvaa vanhan ja toimii samalla tavalla. (AWS, n.d.-h)

AWS asettaa tuoduille varmenteille tiettyjä vaatimuksia. Varmenteen tulee olla version 3 mukainen X.509-varmenne SSL/TLS:ää varten. Varmenne voi olla joko itse allekirjoitettu tai CA:n allekirjoittama. Jos varmenne on CA:n allekirjoittama, niin siinä täytyy käydä ilmi varmennepolku juurivarmentajaan asti. Varmenteen tulee olla PEM-koodattu. (AWS, n.d.-f)

AWS ei tarjoa suoraan sopivaa ratkaisua varmenteiden hallinnan automatisointiin. Kuitenkin luvussa 7.3 esitetään ratkaisu, joka on sopiva myös AWS:n varmenteiden hallinnan automatisointiin.

7.2.3 Google Cloud Platform

GCP:n dokumentaatiosta ei löydy tietoa kolmannen osapuolen varmenteiden käytöstä. Tämä ei siis ainakaan suoraan ole tuettu Googlen Certificate Authority Servicen kautta. Tähän kuitenkin ainakin yksi ratkaisu on käyttää ulkoista KMS-ratkaisua, joka on yhdistettynä GCP:hen.

Tällainen ratkaisu voi olla esimerkiksi HashiCorp Vault, mikä esiteltiin luvussa 6.3.5. Tähän palveluun voidaan viedä varmenteita ja näitä varmenteita voidaan tätä kautta käyttää GCP:ssä.

7.3 Erillinen ohjelmisto varmenteiden käsittelyyn

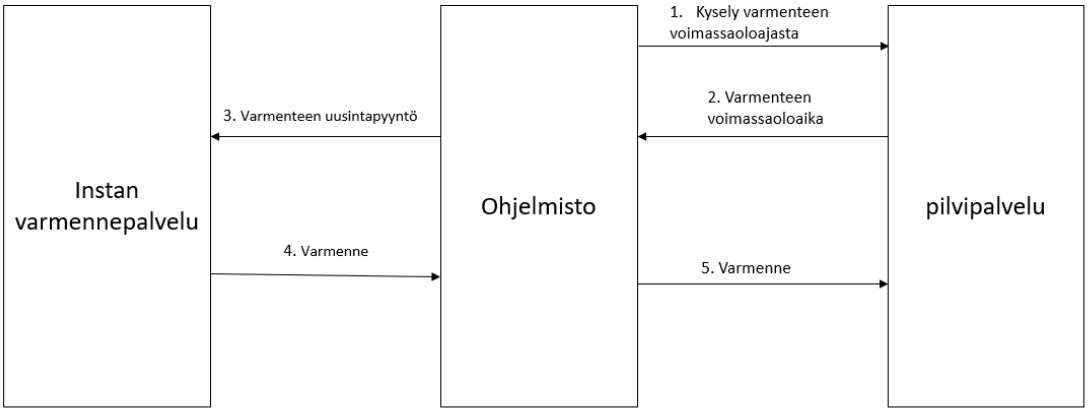
Tässä luvussa esitetyt ratkaisut pilvipalveluiden omissa järjestelmissä ovat olleet riittämättömiä Instan käyttötapauksiin. Näissä ratkaisuissa on ollut yhteistä se, että varmenteita voidaan käyttää pilvipalveluiden omissa järjestelmissä, mutta niiden hallinta on täysin manuaalista. Jos pilvipalveluun on myönnettynä esimerkiksi 100 varmennetta, tarkoittaa tämä sitä, että varmenteiden vanhetessa pilvipalvelun käyttäjän tulee tehdä 100 uutta varmennepyyntöä ja käydä jokainen yksitellen lisäämässä pilvipalveluun. Tämä ei ota huomioon tapauksia, joissa uusi varmenne pitää viedä pilveen, kun varmenne on lisätty CRL:ään ja poistettu käytöstä. Tästä syystä pilvipalvelun valmiit ratkaisut eivät ole soveltuvia suuriin varmennemääriin.

Ratkaisu tähän ongelmaan on kehittää erillinen ohjelmisto, mikä toimii pilvipalveluiden ja Instan varmennepalvelun välissä. Erillinen ohjelmisto käyttää molempien osapuolten REST API-rajapintaa. Tämä ratkaisu on mahdollista automatisoida ja täten se täyttäisi Instan vaatimukset varmenteiden käsittelyn osalta.

Tämä erillinen ohjelmisto sekä CA:t toimivat Instan palvelinsalissa ja vain varmenteet sekä yksityiset avaimet ovat säilössä pilvipalveluiden tai kolmannen osapuolen KMS- tai HSM-ratkaisuissa. Ohjelmiston toiminta perustuu API-kutsuihin siten, että näitä kutsuja lähetetään pilvipalveluihin sekä varmennepalveluun. Pilvipalveluissa varmenteita hallitaan API-kutsuilla muun muassa viemällä uusia varmenteita sekä poistamalla varmenteita käytöstä tarvittaessa.

Instan varmennepalveluun lähetetään API-kutsuja, kun uusia varmenteita täytyy myöntää. Myöntö toteutetaan automaattisesti silloin, kun varmenne on vanhenemassa tai se on poistettu käytöstä. Tämä tieto voidaan saada pilvipalveluista API-kutsuilla tai tieto voidaan nähdä varmennepalvelusta. Kun uusi varmenne myönnetään, niin tämä uusi varmenne ladataan pilvipalveluiden KMS-ratkaisuihin API-rajapinnan avulla.

Kuvassa 13 on esitetty ratkaisu tapauksessa, jossa varmenteen tila kysytään pilvipalvelulta.



Kuva 13. Varmennepyyntö erillisellä ohjelmistolla

Ensimmäisessä vaiheessa tehdään kysely varmenteen voimassaoloajasta. Tämä kysely voidaan tehdä automaattisesti tietyn väliajoin, esimerkiksi kerran päivässä. Tähän kyselyyn pilvipalvelu vastaa kohdassa kaksi varmenteen voimassaoloajalla. Jos varmenne on vanhenemassa, niin siirrytään kohtaan kolme, jossa ohjelmisto lähettää varmennepyynnön varmennepalveluun. Kohdassa neljä varmennepalvelu lähettää myönnetyn varmenteen ohjelmistolle, joka välittää sen kohdassa viisi pilvipalvelulle.

Tällä ratkaisuilla varmenteiden hallinta voidaan toteuttaa pilvipalveluiden ulkopuolella. Tähän on mahdollista käyttää jo olemassa olevia ratkaisuja, kuten CRL sekä OCSP. Varmenteen uusinta voidaan toteuttaa tilanteissa, jossa varmenne on esimerkiksi asetettu CRL:ään.

Azuressa API-kutsuja voidaan lähettää Key Vaultiin pilvipalvelun ulkopuolelta käyttämällä App Register-palvelua. Tämän palvelun avulla luodaan erillinen salaisuus, joka yhdistetään Key Vaultiin ja käytetään sen autentikoitumisessa. Tutkimuksen perusteella havaittiin, että Key Vaultiin pystytään lähettämään GET-request ja näin saamaan sinne tallennettujen salaisuuksien tiedot.

Azuren Key Vaultin REST API-rajapinta mahdollistaa varmenteiden käsittelyn. Tämän avulla varmenteita voidaan viedä pilveen, niitä voidaan poistaa ja niiden tietoja voidaan tarkastella. ”Import Certificate”-toiminnolla varmenteita voidaan viedä tiettyyn Key Vaultiin POST-pyynnön kautta. Varmenteiden poisto tapahtuu ”Delete Certificate”-toiminnolla, jossa lähetetään DELETE-pyyntö Key Vaultiin. Muita mahdollisia operaatioita on esimerkiksi varmenteen päivitys, sekä listaus eri tilassa olevista

varmenteista. Varmenteiden lisäksi tällä REST API-rajapinnalla on mahdollista käsitellä myös muita Key Vaultissa olevia salaisuuksia, kuten avaimia. (Microsoft Azure, n.d.-a)

Myös AWS:ssä on mahdollista käyttää API-rajapintaa varmenteiden hallintaan. Tämä tapahtuu AWS Certificate Managerin kautta, mitä käsiteltiin luvussa 7.2.2. HTTP API-pyyntö tulee allekirjoittaa Signature Version 4:llä. Tämän avulla tapahtuu autentikointi, jossa käytetään pääsynhallinta-avainta. Tämä avain koostuu ID:stä sekä salaisesta pääsynhallinta-avaimesta. (AWS, n.d.-b)

ACM:ssä voidaan tehdä samankaltaisia pyyntöjä, kuin Azuren Key Vaultiin. Varmenteita voidaan ladata palveluun, niitä voidaan poistaa sekä ne voidaan näyttää listana. Varmenteiden lataus tapahtuu ImportCertificate-toiminnolla, jolloin varmenteille on samat vaatimukset, jotka mainittiin luvussa 7.2.2. ”DeleteCertificate”-toiminto poistaa varmenteen sekä sen yksityisen avaimen ja ”ListCertificates”-toiminnon avulla saadaan lista kaikista varmenteista tai varmenteista tietyillä ehdoilla. (AWS, n.d.-b)

Pilvipalveluiden lisäksi HashiCorpin Vaultia voidaan käyttää API-rajapinnan kanssa. HashiCorpin mukaan API-rajapinta mahdollistaa täyden pääsyn Vaultiin käsiksi. Vaultiin autentikointi API:lla vaatii erillisen tokenin ja autentikoinnin avulla voidaan Vaultista hakea esimerkiksi salaisuuksia GET-operaation avulla tai uusia salaisuuksia voidaan luoda POST-operaation avulla. (HashiCorp, n.d.-b)

Tämän ehdotetun ohjelmiston avulla voidaan automatisoidun varmenteiden käsittelyn lisäksi hallita varmenteita sekä salaisuuksia koostetusti yhdestä paikasta. Jos kyseessä on esimerkiksi multi-cloud-käyttötapaus, voi varmenteiden käsittely olla monimutkaista ja tämän avulla se voidaan toteuttaa hallitusti.

Tämä ehdotettu ohjelmisto on ominaisuuksiltaan lähellä RA:ta. Tällä ohjelmistolla pystytään myös saavuttamaan RA:n asema asettamalla sille politiikoita varmennepyyntöjen käsittelyyn sekä käyttäjien autentikointiin liittyen. Esimerkiksi varmennepolitiikoilla voidaan asettaa rajoituksia varmennepyynnön sisältöön liittyen.

8. YHTEENVETO

Työssä tutkittiin julkisen avaimen infrastruktuurin käyttöä pilvipalveluissa. Tarkemmin tutkimus keskittyi avaintenhallintaratkaisuihin, varmenteiden hallintaan sekä varmenteiden viemiseen pilvipalveluihin. Työn kohteena olivat kolme suurinta pilvipalveluntarjoajaa. Työn tavoitteena oli tutkia miten varmenteita käytetään pilvipalveluissa ja onko pilvipalveluissa mahdollisuuksia kolmannen osapuolen varmennepalveluille.

Työssä vertailtiin avaintenhallintaratkaisuja. Vertailun kohteena olivat On-Premise-ratkaisu, pilvipalveluiden omat avaintehallintaratkaisut sekä ulkopuolisen toimijan tarjoamat palvelut. Avaintenhallintaratkaisut olivat pilvipalveluiden osalta riittävät, mutta luottamus pilvipalveluihin voi aiheuttaa rajoituksia. Myös tilanteissa, joissa pilvipalveluita on useampia käytössä, voi avainten hallinta olla vaikeaa. Ulkoisen palveluntarjoajan ratkaisuissa on sama luottamusongelma, mutta avaimet voidaan keskittää yhteen palveluun pilvipalveluja varten. On-Premise-ratkaisussa luottamusongelma saadaan minimoitua, mutta avainten integroiminen pilvipalveluun aiheuttaa vaikeuksia sekä avaintenhallintaratkaisu on toteutettava itse. Tämä vaatii jatkotutkimusta sekä kehitystyötä, jotta optimaalinen ratkaisu löytyisi.

Varmenteiden viemiseen pilvipalveluihin tutkittiin pilvipalveluiden valmiita ratkaisuja. Nämä todettiin soveltumattomaksi Instan tarpeisiin, sillä kolmannen osapuolen varmennepalvelun myöntämien varmenteiden hallinta oli manuaalista, eikä automatisointia pystytty tekemään. Työssä kuitenkin ehdotettiin ratkaisuksi erillinen ohjelmisto, jolla automatisointi pystyttäisiin toteuttamaan.

Tutkimustuloksien pohjalta toimintaehdotus on rakentaa erillinen ohjelmisto varmenteiden käsittelyyn. Tämä ratkaisu soveltuu Instan asiakkaiden tarpeisiin varmenteiden hallinnassa. Tämän ohjelmiston toiminta vaatii erillistä jatkotutkimusta pilvipalveluiden API-rajapinnan käytössä. Rajoituksia voi tulla muun muassa API-kutsujen määrissä. Tämä vaatii myös kehitystyötä, sillä tätä varten pitää kehittää toimiva ohjelmisto. Ohjelma vaatii muun muassa toiminnallisen osan, mahdollisen käyttöliittymän sekä varmenteiden automatisoituun hallintaan liittyvän käyttölogiikan.

Johdannossa mainitussa tutkimuksessa ”Cloud-Trust – a Security Assessment Model for Infrastructure as a Service (IaaS) Clouds” selvisi se, että pilvipalveluntarjoajiin ei voidan täysin luottaa. Tämä myös vaikutti vahvasti tämän tutkimuksen lopputuloksiin, sillä moni tutkimuksen tuloksista jäi luottamuksen varaan.

Työ voidaan todeta onnistuneeksi, sillä tutkimuskysymykseen saatiin vastaus. Työn tulokset voivat kuitenkin muuttua tulevaisuudessa, sillä pilvipalvelut kehittyvät jatkuvasti ja myös tälle osa-alueelle voi tulla kehitystä. Pilvipalveluntarjoajat voivat esimerkiksi alkaa tarjoamaan varmenteiden automaattista uusintaa kolmannen osapuolen CA:n myöntämiin varmenteisiin.

LÄHTEET

Adams, C., Adams, C., & Lloyd, S. (2003). *Understanding PKI : concepts, standards, and deployment considerations* (2nd ed.). Addison-Wesley.

Annabelle, L., Miles, E. S., & Stanley, R. S. (2001). Security Requirements for Cryptographic Modules [includes Change Notices as of 12/3/2002]. In *National Institute of Standards and Technology*. https://www.nist.gov/publications/security-requirements-cryptographic-modules-includes-change-notices-1232002

AWS. (n.d.-a). *About AWS* . https://aws.amazon.com/about-aws/

AWS. (n.d.-b). *AWS Certificate Manager API Reference*. https://docs.aws.amazon.com/acm/latest/APIReference/Welcome.html

AWS. (n.d.-c). *AWS CloudHSM Use Cases* . https://docs.aws.amazon.com/cloudhsm/latest/userguide/use-cases.html

AWS. (n.d.-d). *AWS Key Management Service features* . https://aws.amazon.com/kms/features/

AWS. (n.d.-e). *Configuring mutual TLS authentication for a REST API*. Https://Docs.Aws.Amazon.Com/Apigateway/Latest/Developerguide/Rest-Api-Mutual-Tls.Html.

AWS. (n.d.-f). *Importing certificates into AWS Certificate Manager* . https://docs.aws.amazon.com/acm/latest/userguide/import-certificate.html

AWS. (n.d.-g). *Other AWS services that use X.509 public key certificates* . https://docs.aws.amazon.com/crypto/latest/userguide/awspki-service-other.html

AWS. (n.d.-h). *What Is AWS Certificate Manager?* . https://docs.aws.amazon.com/acm/latest/userguide/acm-overview.html

AWS. (n.d.-i). *When to use AWS Key Management Service (AWS KMS)* . https://docs.aws.amazon.com/crypto/latest/userguide/awscryp-choose-kms.html

AWS. (2021). *ACM Private CA now supports the Online Certificate Status Protocol (OCSP)* . https://aws.amazon.com/about-aws/whats-new/2021/09/acm-private-ca-online-certificate-ocsp/

Berbecaru, D., Desai, A., & Lioy, A. (2009). A unified and flexible solution for integrating CRL and OCSP into PKI applications. *Software, Practice & Experience; Softw: Pract.Exper*, *39*(10), 891–921. https://doi.org/10.1002/spe.918

Boyen, S., Santesson, S., Polk, T., Housley, R., Farrell, S., & Cooper, D. (2008). *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. https://rfc-editor.org/rfc/rfc5280.txt

Braeken, A. (2021). Public key versus symmetric key cryptography in client–server authentication protocols. *International Journal of Information Security*. https://doi.org/10.1007/s10207-021-00543-w

Buchanan, W., Lanc, D., Ukwandu, E., Fan, L., Russell, G., & Lo, O. (2015). The Future Internet: A World of Secret Shares [Article]. *Future Internet*, *7*(4), 445–464. https://doi.org/10.3390/fi7040445

Chen, J. (2020, July 20). *Azure Fundamental: IaaS, PaaS, SaaS*. Https://Medium.Com/Chenjd-Xyz/Azure-Fundamental-Iaas-Paas-Saas-973e0c406de7.

Chen, L., Takabi, H., & Le-Khac, N.-A. (2019). *Security, privacy and digital forensics in the cloud* (L. Chen, H. Takabi, & N.-A. Le-Khac, Eds.; 1st edition) [Book]. Wiley.

de Prisco, R., de Santis, A., & Mannetta, M. (2018). Reducing costs in HSM-based data centers. *Journal of High Speed Networks*, *24*(4), 363–373. https://doi.org/10.3233/JHS-180600

DocuSign. (n.d.). *Understanding digital signatures*. Https://Www.Docusign.Com/How-It-Works/Electronic-Signature/Digital-Signature/Digital-Signature-Faq.

Froehlich, A. (2021). *registration authority (RA)*. TechTarget.

Gonzales, D., Kaplan, J. M., Saltzman, E., Winkelman, Z., & Woods, D. (2017). Cloud-Trust—a Security Assessment Model for Infrastructure as a Service (IaaS) Clouds [Article]. *IEEE Transactions on Cloud Computing*, *5*(3), 523–536. https://doi.org/10.1109/TCC.2015.2415794

Google Cloud. (n.d.-a). *Certificate Authority Service* . https://cloud.google.com/certificate-authority-service

Google Cloud. (n.d.-b). *Cloud Key Management* . https://cloud.google.com/security-key-management#section-10

Google Cloud. (n.d.-c). *Hosted Private HSM* . https://cloud.google.com/kms/docs/hosted-private-hsm

Google Cloud. (n.d.-d). *How-to guides*. https://cloud.google.com/certificate-authority-service/docs/how-to

Google Cloud. (n.d.-e). *Training and tutorials* . https://cloud.google.com/certificate-authority-service/docs#training-and-tutorials

HashiCorp. (n.d.-a). *HashiCorp documentation*. https://www.vaultproject.io/docs

HashiCorp. (n.d.-b). *HTTP API* . https://www.vaultproject.io/api-docs/index

Huang, J., & Nicol, D. M. (2017). An anatomy of trust in public key infrastructure. *International Journal of Critical Infrastructures; Ijcis*, *13*(2–3), 238–258. https://doi.org/10.1504/IJCIS.2017.088234

Hunt, R. (2001). Technological infrastructure for PKI and digital certification. *Computer Communications*, *24*(14), 1460–1471. https://doi.org/10.1016/S0140-3664(01)00293-6

IBM. (n.d.). *z/OS Cryptographic Services PKI Services Guide and Reference*. IBM.

Kanikathottu, H. (2020). *AWS security cookbook : practical solutions for managing security policies, monitoring, auditing, and compliance with AWS* (1st ed.). Packt.

Khan, S., Zhang, Z., Zhu, L., Rahim, M. A., Ahmad, S., & Chen, R. (2020). SCM: Secure and accountable TLS certificate management. *International Journal of Communication Systems*, *33*(15), e4503-n/a. https://doi.org/10.1002/dac.4503

Kuzminykh, I., Ghita, B., & Shiaeles, S. (2020). *Comparative Analysis of Cryptographic Key Management Systems* (Vol. 12526, pp. 80–94). Springer International Publishing. https://doi.org/10.1007/978-3-030-65729-1_8

Larramo, M. (n.d.). *What is Code Signing / Digital Signature / Digital Certificate? (Q&A)*. Https://Www.Samlogic.Net/Articles/Code-Signing.Htm.

Massé, M. (2011). *REST API Design Rulebook* (1st ed.). O'Reilly Media Incorporated.

Microsoft Azure. (n.d.-a). *Azure Key Vault REST API reference*. https://docs.microsoft.com/en-us/rest/api/keyvault/

Microsoft Azure. (n.d.-b). *What is cloud computing?* . https://azure.microsoft.com/en-us/overview/what-is-cloud-computing/

Microsoft Azure. (2016). *Manage certificates via Azure Key Vault* . https://docs.microsoft.com/en-gb/archive/blogs/kv/manage-certificates-via-azure-key-vault

Microsoft Azure. (2019a). *Azure Key Vault basic concepts* . https://docs.microsoft.com/en-us/azure/key-vault/general/basic-concepts

Microsoft Azure. (2019b). *Windows virtual machines in Azure* . https://docs.microsoft.com/en-us/azure/virtual-machines/windows/overview

Microsoft Azure. (2020a). *Get started with Key Vault certificates* . https://docs.microsoft.com/en-us/azure/key-vault/certificates/certificate-scenarios

Microsoft Azure. (2020b). *Renew your Azure Key Vault certificates* . https://docs.microsoft.com/en-us/azure/key-vault/certificates/overview-renew-certificate

Microsoft Azure. (2021a). *About Azure Key Vault* . https://docs.microsoft.com/en-us/azure/key-vault/general/overview

Microsoft Azure. (2021b). *What is Azure Dedicated HSM?* . https://docs.microsoft.com/en-us/azure/dedicated-hsm/overview

Microsoft Azure. (2021c, July 16). *Device Authentication using X.509 CA Certificates*. Https://Docs.Microsoft.Com/En-Us/Azure/Iot-Hub/Iot-Hub-X509ca-Overview.

Microsoft Azure. (2021d, December 30). *Sign packages with Azure Key Vault*. Https://Docs.Microsoft.Com/En-Us/Windows/Msix/Desktop/Sign-with-Akv-Cert.

Noor, T., Sheng, Q., Zeadally, S., & Yu, J. (2013). Trust management of services in cloud environments: Obstacles and solutions. *ACM Computing Surveys*, *46*(1), 1–30. https://doi.org/10.1145/2522968.2522980

Nystrom, M., & Kaliski, B. (2000). *PKCS #10: Certification Request Syntax Specification*. https://datatracker.ietf.org/doc/html/rfc2986#section-1

ØLNES, J., & BUENE, L. (2006). *Use of a validation authority to provide risk management for the PKI relying party* (pp. 1–15). Springer.

Perez, A. (2014). *Network security*. ISTE Limited.

Ramgir, M. (2019). *Internet of Things* (1st edition) [Book]. Pearson Education India.

Red Hat. (2020). *https://www.redhat.com/en/topics/api/what-is-a-rest-api*. https://www.redhat.com/en/topics/api/what-is-a-rest-api

Smith, J. E., & Nair, R. (2005). The architecture of virtual machines. *Computer (Long Beach, Calif.)*, *38*(5), 32–38. https://doi.org/10.1109/MC.2005.173

Stapleton, J. J. (2016). *Security without obscurity : a guide to PKI operations* (W. C. Epstein, Ed.; 1st ed.) [Book]. CRC Press. https://doi.org/10.1201/b19725

Synergy Research Group. (2021, October 28). *Amazon, Microsoft & Google Grab the Big Numbers – But Rest of Cloud Market Still Grows by 27%*. Synergy Research Group. https://www.srgresearch.com/articles/amazon-microsoft-google-grab-the-big-numbers-but-rest-of-cloud-market-still-grows-by-27

Tulloch, M. (2013). *Introducing Windows Azure for It Professionals* (1st ed.). Microsoft Press, c2013.

Turner, D. M. (2016). *What is Key Management? a CISO Perspective* . https://www.cryptomathic.com/news-events/blog/what-is-key-management-a-ciso-perspective

Ulz, T., Pieber, T., Steger, C., Haas, S., Bock, H., & Matischek, R. (2017). Bring your own key for the industrial Internet of Things. In *ICIT* (pp. 1430–1435). IEEE. https://doi.org/10.1109/ICIT.2017.7915575

Vacca, J. R. (2004). *Public Key Infrastructure: Building Trusted Applications and Web Services*. Auerbach Publications. https://doi.org/10.1201/9780203498156

Zissis, D., & Lekkas, D. (2012). Addressing cloud computing security issues [Article]. *Future Generation Computer Systems*, *28*(3), 583–592. https://doi.org/10.1016/j.future.2010.12.006