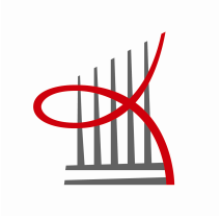


PIENYRITYKSEN SISÄVERKON PALVELUIDEN TIETO-TURVALLINEN ETÄKÄYTTÖ ASIAKASSERTIFIKAATEIL-LA.

Taneli Mäenpää

Opinnäytetyö
Toukokuu 2011
Tietojenkäsittely
Tietojenkäsittelyn koulutusohjelma

TAMPEREEN AMMATTIKORKEAKOULU
Tampere University of Applied Sciences

Tampereen ammattikorkeakoulu
Tietojenkäsittelyn koulutusohjelma

MÄENPÄÄ, TANELI: Pienyrityksen sisäverkon palveluiden tietoturvallinen etäkäyttö asiakassertifikaateilla.

Opinnäytetyö 42 s.
Toukokuu 2011

TIIVISTELMÄ

Opinnäytetyö käsittelee tilaajayrityksen työntekijöiden tarvetta käyttää yrityksen sisäverkon palveluita ulkoverkosta käsin sekä sitä, miten kyseiset palvelut voidaan toteuttaa tietoturvallisesti. Työn tarkoituksena on suunnitella, toteuttaa ja dokumentoida kaikki eri vaiheet etäkäyttöä mahdollistettaessa. Lisäksi tavoitteena on tuottaa helposti ylläpidettävä kokonaisuus etäkäytön myöhempää hallintaa varten.

Lähtötilanteessa tilaajayrityksen työntekijät ovat tehneet raportointia ja muuta työntekoon liittyvää dokumentointia ainoastaan fyysisesti yrityksen tiloissa tai sähköpostitse. Sähköpostin käyttö aiheuttaa saman tiedon kirjaamista useaan kertaan, sillä tiedot joudutaan dokumentoimaan lisäksi kohdejärjestelmään. Sisäverkossa on käytössä useita tapoja, joiden avulla pidetään kirjaa tehdyistä asioista.

Sisäverkon palveluiden etäkäyttö ja palveluiden yhtenäistäminen mahdollistaa nopeamman informaatiokulun yrityksen henkilöiden välillä sekä reaaliaikaisen raportoinnin suoraan kohdeohjelmistoon. Tilaajayrityksen asiakkaina on yrityksiä kouluja sairaaloita ja kuntia ympäri Suomea. Mahdollisuus seurata toimistolta, mitä henkilöstö tekee asiakasyrityksen tiloissa, sekä mahdollisuus muuttuvien piirustusten päivittämiseen etänä tekee siitä entistä kilpailukykyisemmän yrityksen tiukassa markkinatilanteessa.

Työhön sisältyi runsaasti dokumentointia, toteutusvalintoja sekä asetustiedostojen muokkausta. Tämä työ pitää sisällään esittelyn käytetyistä alustoista, keskeisimmät käytetyt ohjelmistot, kuvauksen konfiguroinnin kulusta sekä pohdinnan yrityksen kehityksestä tulevaisuudesta.

Asiasanat: PKI, sertifikaatti, Apache, salaus, palvelin

Tampere University of Applied Sciences

Degree programme in Computer Science

MÄENPÄÄ, TANELI: Secure Remote Access to Small Business Intranet Services with Client Certificates.

Bachelor's thesis 42 pages
May 2011

ABSTRACT

This thesis covers the need for employees working for client organization be able to use intranet services from an outside network, and the possibilities for providing that function as securely as possible. The main purpose of this work is to plan, execute and document the different stages when enabling remote access. Another aspect is to produce an easily manageable entity for later administrational needs.

At starting point, the employees have been doing their reports or other work-related documentation only when they are physically in the company's premises or via e-mail. The use of e-mail causes the same information to be documented more than once, because of the need to insert that information also to the target system. There are multiple different services and programs provided in the LAN that are used in documentation process.

Securing the local services to be used from outside network, and unifying the services used enables faster information movement between workers and real-time reporting to the target system. The client organization has companies, schools, hospitals and local communities as its customers. The ability to follow the progress of an employee doing installation in customer's premises, or the possibility to update existing blueprints, makes the client a more competitive company in the tight market situation.

This thesis involves documentation, implementation choices and modifying configuration files. This thesis contains an introduction of used platforms, fundamental software, description of configuration and a pondering of the future development.

Keywords: PKI, certificate, Apache, encryption, server

SISÄLLYS

1 LYHENTEET

AJP	Apache JServ Protocol. Protokolla, jolla välitetään pyyntöjä web-palvelimelta sovelluspalvelimelle
CA	Certificate authority. Sertifikaatteja myöntävä luotettu taho, varmentaja
CRL	Certificate revocation list. Lista, johon kerätään tietyn varmentajan eväämät sertifikaatit, sulkulista
CSR	Certificate signing request. Allekirjoituspyyntö, jonka sertifikaatin pyytäjä lähettää varmentajalle
IRC	Internet relay chat. Pikaviestintäpalvelu
LDAP	Lightweight directory access protocol. Hakemistopalvelun käyttöön tarkoitettu protokolla
PKI	Public key infrastructure. Hallintajärjestelmä, jonka avulla julkisen avaimen aitouteen voidaan luottaa
RA	Revocation authority. Sertifikaattien kumoamiseen oikeutettu taho
RFC	Request for comments. IETF:n julkaisema muistio, joka kuvaa internet-järjestelmien käyttämiä metodeja
SMTP	Simple mail transfer protocol. Sähköpostin lähetykseen käytetty protokolla
SSL	Secure sockets layer. Protokolla, jonka avulla tietoliikennettä voidaan suojata

2 JOHDANTO

Nyky-yhteiskunnassa verkostoituminen on missä tahansa työpaikassa tärkeää. Koska tilaajayrityksen toimialana on yritysten verkkopohjaiset viestintä- ja hälytysratkaisut, joudutaan asennustöiden merkeissä viettämään asiakkaan tiloissa paljon aikaa. Aika, jolloin asennushenkilöllä ei ole suoraa pääsyä työssä käytettäviin resursseihin, on tuottamatonta aikaa.

Tilaajayritys työllistää noin 15 henkilöä, joiden työnkuvat vaihtelevat huomattavasti. Yhteistä kaikille näille töille on kuitenkin se, että niissä käytetään tietokonetta muokkaamaan, säilyttämään, siirtämään, tulostamaan sekä luomaan asiakkaisiin liittyvää informaatiota. Tämä informaatio voi olla esimerkiksi tarjouspyyntö, teknінen piirustus, muokattu ohjelmisto tai vaikkapa raportti tehdystä työstä.

Yrityksen käytäntöjen yhtenäistämiseksi on aloitettu sisäinen prosessi wiki-alustan käyttöönottamiseksi. Confluence-yrityswiki on hyvää vauhtia ottamassa tärkeimmän ohjelmiston paikkaa sisäisen informaation säilyttämisessä ja jakamisessa. Runsas valikoima sisäisiä ominaisuuksia ja valinnaisesti asennettavia liitännäisiä luovat Confluencesta monipuolisen alustan, mutta vain sisäverkkoon. Tämänkaltaiset resurssit tuovat työntekijöille lisäarvoa vain silloin, kun ne ovat käytettävissä mistä tahansa, sillä myös asiakkaat ovat ympäri Suomea.

Asiakaskohtaiset suunnitelmat juuri kyseiseen projektiin on valmisteltu, tarkastettu ja arkistoitu toimistolla. Kesken asiakkaan tiloissa tapahtuvan työn voidaan kuitenkin joutua muuttamaan piirustuksia, vikaraportteja, asiakastietoja, omien ohjelmistojen päivityksiä sekä muita dokumentteja. Tilanne on tuttu kaikille toimiston ulkopuolella töitä tekevälle. Tähän epäkohtaan tilaajayrityksen järjestelmässä ei ole kunnolla valmistauduttu, mutta siihen on haluttu muutosta jo vuosia. Tämä työ on tilattu mainitun epäkohdan korjaamiseksi.

3 TAUSTAT JA TAVOITTEET

Toimeksiantajana opinnäytetyössäni on yritys, jonka ydinosaamisaluetta ovat verkkopohjaiset viestintäratkaisut, kuten ohjausjärjestelmät, viestintäohjelmistot ja hälytysjärjestelmät. Osa työntekijöistä, ensisijaisesti asentajat, viettävät runsaasti aikaa toimiston ulkopuolella asennustöissä asiakkaiden tiloissa. Varsinaisten työtilojen ulkopuolella toimiminen tuottaa tarpeen päästä käsiksi asiakasyrityksen tietoihin myös sen sisäverkon ulkopuolelta.

Aloitin yrityksessä ylläpitäjänä syksyllä 2009. Yrityksessä ei ole aikaisemmin ollut pelkästään ylläpitoon keskittyvää henkilöä, vaan ylläpitotoimet on hoitanut joku muu työntekijä silloin, kun aikaa on sattunut löytymään. Käytössä olevien dokumentaatio-ohjelmistojen seuraajaksi oli jo ennen minun aikaani asennettu Confluence-niminen yrityswiki, joka kaikkien työntekijöiden olisi tarkoitus ottaa tulevaisuudessa käyttöön yhteisenä tiedonkeruualustana. Etäkäyttömahdollisuuden puuttuessa tätä ei kuitenkaan voitu toteuttaa. Sama ongelma oli myös yrityksen sähköpostin kanssa, johon ei ollut pääsyä ulkoverkosta käsin.

Tavoitteena oli ottaa käyttöön joko VPN- tai sertifikaattipohjainen etäkäyttöratkaisu siten, että tietomurron mahdollisuus olisi pieni. VPN-järjestelmät tuovat etäkäyttötapauksissa runsaammat mahdollisuudet vaikuttaa kohdekoneen tietoihin, joten päädyimme yhteistuumin sertifikaateilla turvattuun vaihtoehtoon. Ohjelmisto- ja toteutustapavalintojen sekä konfiguroinnin lisäksi koko prosessi oli tarkoitus dokumentoida yrityksen sisäiseen wikiin. Tämä selkeyttää toimenpiteitä vikatilanteissa ja antaa järjestelmää myöhemmin ylläpitäville henkilöille mahdollisuuden tutkia perusasetuksiin tehtyjä muutoksia.

Työssä esitellään perusteita salauksesta, käydään läpi palveluiden rakennetta ennen muutoksia, kerrotaan tehdyt toimenpiteet tietoturvallisen palvelun mahdollistamiseksi ja tutkitaan vaiheita ylläpitäjän näkökulmasta.

4 PERUSTEET

4.1 Julkisen avaimen infrastruktuuri

Internetissä sijaitsevien erinäisten standardien kehityksessä mukana oleva taho IETF (Internet engineering task force) tuottaa laadukkaita dokumentteja, joiden avulla se haluaa saada itse internetistä toimivamma paikan kaikille (IETF 2011). Julkisen avaimen infrastruktuuri pohjautuu IETF:n muotoileman RFC 5280 -standardiin. Kyseinen standardi määrittelee yksityiskohtaisesti X.509 PKI-sertifikaattien sekä sulkulistojen profiilien sisällön ja syntaksin.

4.1.1 Toimintatapa

Varmentaja on taho, joka on vastuussa varmenteen hakijan tietojen oikeellisuudesta. Tähän perustuu julkisen avaimen infrastruktuuri (Weise 2001). Yksinkertaisin PKI-tilanne pitää sisällään yhden ylimmän tason varmentajan ja tämän itselleen myöntämän sertifikaatin, eli varmenteen. Kun joku taho tarvitsee sertifikaattia käyttöönsä, sitä anotaan varmentajalta. Varmentaja myöntää luottamalleen taholle tämän anoman sertifikaatin, jolloin sitä voidaan käyttää sille myönnettyyn tarkoitukseen. Täten eri palvelut voivat käyttää eri sertifikaatteja, vaikka palveluiden tarjoaja olisi sama. Tärkein merkitsevä tekijä varmenteita haettaessa on hakutietojen kohta, joka määrittelee kenelle kyseistä varmennetta haetaan.

4.1.2 Salausalgoritmit

Salaustapoja on kahta perustyyppiä: symmetrisiä ja epäsymmetrisiä. Symmetrisessä salauksessa tiedon salaamiseen ja purkamiseen käytetään samaa avainta. Epäsymmetrisessä salauksessa käytetään kahta avainta: julkista ja yksityistä.

Oleellista symmetrisessä salauksessa on avaintenvaihtoprosessi, joka täytyy toistaa jokaisen kommunikointikerran yhteydessä. Epäsymmetrinen salaus vaatii symmetristä enemmän resursseja, mutta mahdollisuutta purkuavaimen saamiseen siepatuksi liikenteen seasta ei ole. Simon Singh käyttää epäsymmetrisen salauksen toimintaa kuvatessaan riippulukkovertauskuvaa, mikä helpottaa puolestaan aiheen käsittämistä:

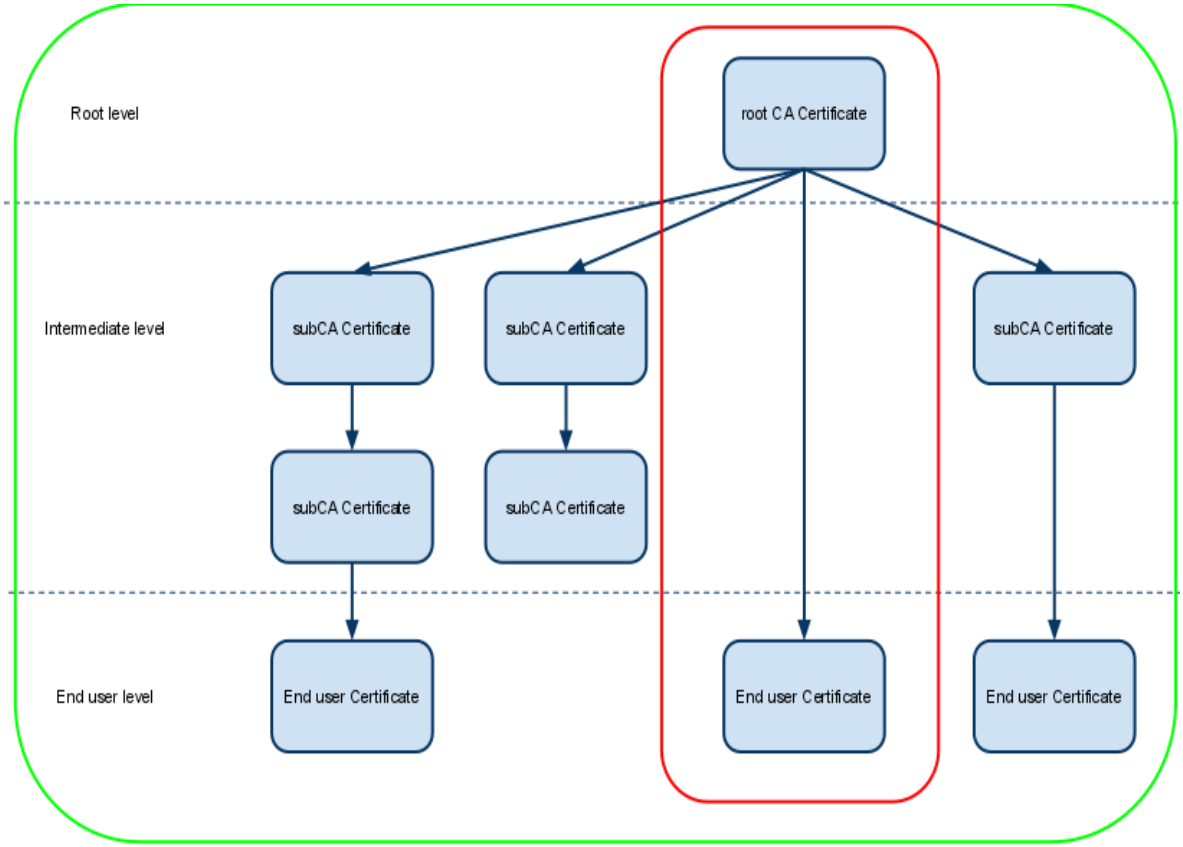
> *"Jos palataan riippulukkovertaukseen, epäsymmetristä salakirjoitusta voidaan kuvata seuraavasti. Kuka tahansa voi sulkea riippulukon vain napsauttamalla sen kiinni, mutta vain avaimen haltija voi avata sen. Lukitseminen (koodaus) on helppoa, siihen pystyvät kaikki, mutta avaamiseen (tulkitsemiseen) pystyy vain avaimen haltija. Jokapäiväinen tieto siitä, kuinka riippulukko suljetaan, ei kerro, kuinka se avataan. Jos viedään vertausta vielä pidemmälle, kuvitellaan, että Liisa suunnittelee riippulukon ja avaimen. Hän pitää avaimen itsellään, mutta valmistaa tuhansia samanlaisia riippulukkoja ja jakaa ne postitoimistoihin ympäri maailman. Jos Pekka haluaa lähettää sanoman, hän panee sen laatikkoon, menee paikalliseen postitoimistoon, pyytää "Liisa-riippulukkoa" ja lukitsee laatikon. Nyt hän ei pysty avaamaan laatikkoa, mutta kun Liisa saa laatikon, hän voi avata lukon omalla ainutlaatuisella avaimellaan. Riippulukko ja sen kiinni napsauttaminen vastaavat julkista salausavainta, koska kaikilla on mahdollisuus hankkia riippulukko ja kaikki osaavat lukita riippulukolla viestin laatikkoon. Riippulukon avain vastaa yksityistä avausavainta, koska sellainen on vain Liisalla, vain hän voi avata riippulukon ja saada käsiinsä laatikossa olevan viestin." (Singh 1999, 364)*

Nykyaikaiset tietoliikenteessä käytettävät salausjärjestelmät käyttävät hyödyksi kumpaakin salaustapaa. Yhteydenoton alussa käytetään epäsymmetristä salausta symmetrisen salausavaimen koodaamiseen, jonka jälkeen siirrytään käyttämään symmetristä salausta itse sisällölle. Tämä käytäntö varmistaa avaimenvaihtoprosessin turvallisuuden sekä salauksen nopeuden itse sisältöä siirrettäessä.

4.1.3 PKI-hierarkia

Hierarkia alkaa juuritasolta. Tällä tasolla toimii juurivarmenne. Juurivarmenne on aina itse allekirjoitettu, sillä korkeampaa allekirjoittavaa tahoa ei ole. Tärkeää juurivarmenteessa on se, että tiedetään varmasti kuka sen on tehnyt. Oletuksena selaimien tiedoissa on aina luotettuina varmentajina suuret julkiset tahot, kuten Thawte, VeriSign ja GeoTrust (Mozilla 2011).

Varmenteissa käytetty hierarkia voi olla vain yhden juurisertifikaatin sisältävä (Kuva 1, punainen), tai tarvittaessa lukuisiakin välivarmentajia sisältävä (Kuva 1, vihreä). Perusperiaate on kuitenkin aina sama - jos luotat johonkin juuritason varmentajaan, luotat samalla kaikkiin tämän varmentajan myöntämiin sertifikaatteihin.



KUVA 1: Esimerkki PKI-hierarkiasta.

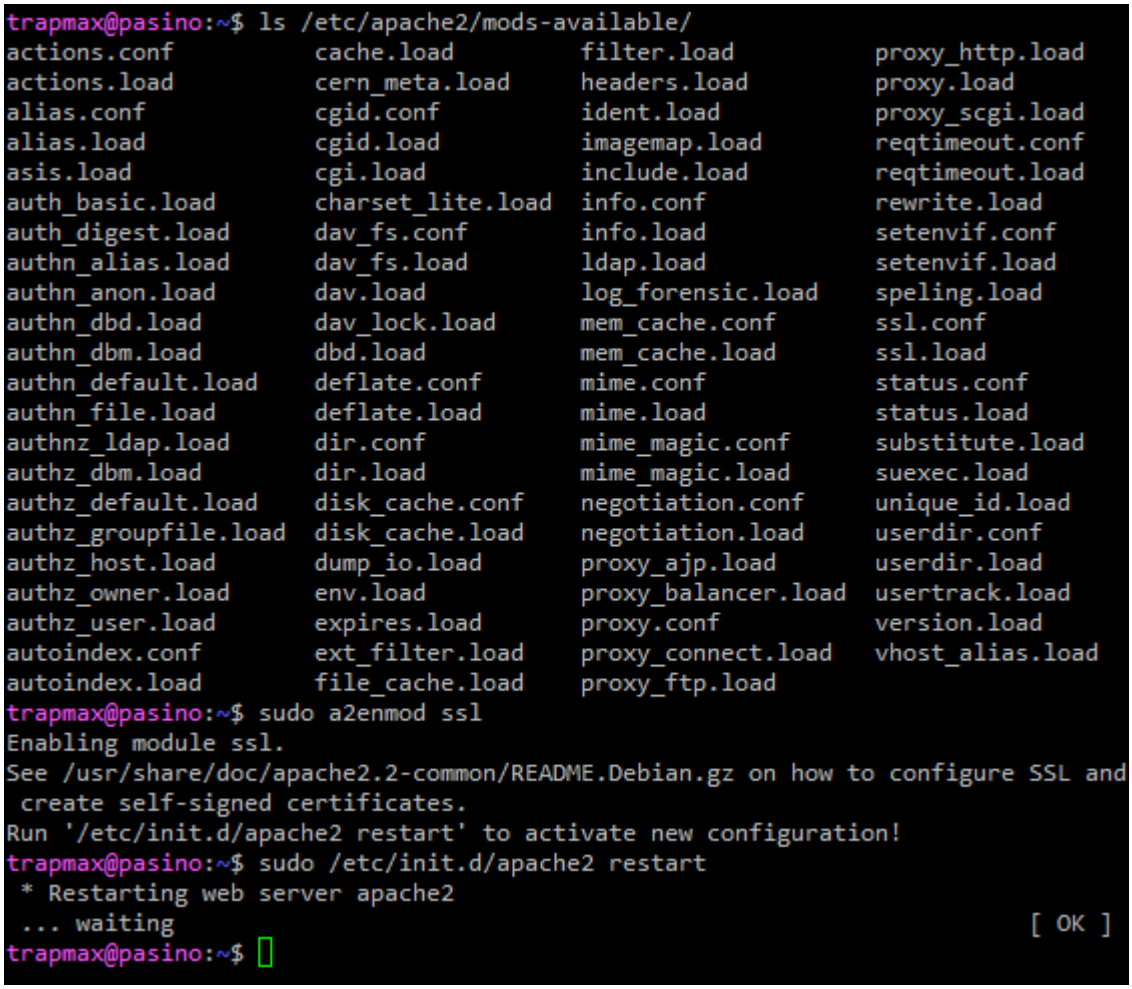
4.1.4 Varmenteet

Varmenteet pitävät sisällään informaatiota tietystä tahosta (Viestintävirasto 2011) - käytäntöä voidaan verrata passijärjestelmään. Suomessa poliisi on passeja myöntävä, eli luotettu taho, ja passia voidaan käyttää henkilön tunnistamiseen luotettavasti. Sama periaate pätee varmenteissa. Varmenteen myöntävä taho, varmentaja, on varmistanut, että sitä käyttävä taho on validi. Varmentaja voi myös määritellä myöntämänsä varmenteen käyttökohteen, joita voi yhteen varmenteeseen olla määritetty useitakin. Yleisin käyttökohde on WWW-palvelimelle myönnetty varmenne, jota käytetään salatun http-liikenteen (https) yhteydessä. Varmenteita voidaan käyttää tietysti myös henkilön tunnistamiseen monissa eri palveluissa - sähköpostin salaaminen ollee näistä käyttökohteista tärkeimpiä.

4.2 WWW-liikenteen suojaus

4.2.1 Salattu liikenne

Yleisesti internet-selaimet käyttävät tiedon siirtämiseen http-protokollaa. Huono puoli tämän protokollan käyttämisessä on se, että tietoa ei salata millään tavalla sen kulkiessa asiakaskoneelta palvelimelle ja päinvastoin. Salaamattoman liikenteen käyttö mahdollistaa monia metodeja tiedon urkintaan. Ratkaisuna voidaan käyttää SSL-protokollaa, joka siirtää tiedon salattuna. SSL-protokolla on helppo ottaa käyttöön, kun käytössä on Apache http -palvelin ja Ubuntu server -linux-jakelu (Kuva 2). Moduulin käyttöönoton jälkeen on tietysti luotava SSL-yhteyttä varten omat konfigurointinsa.

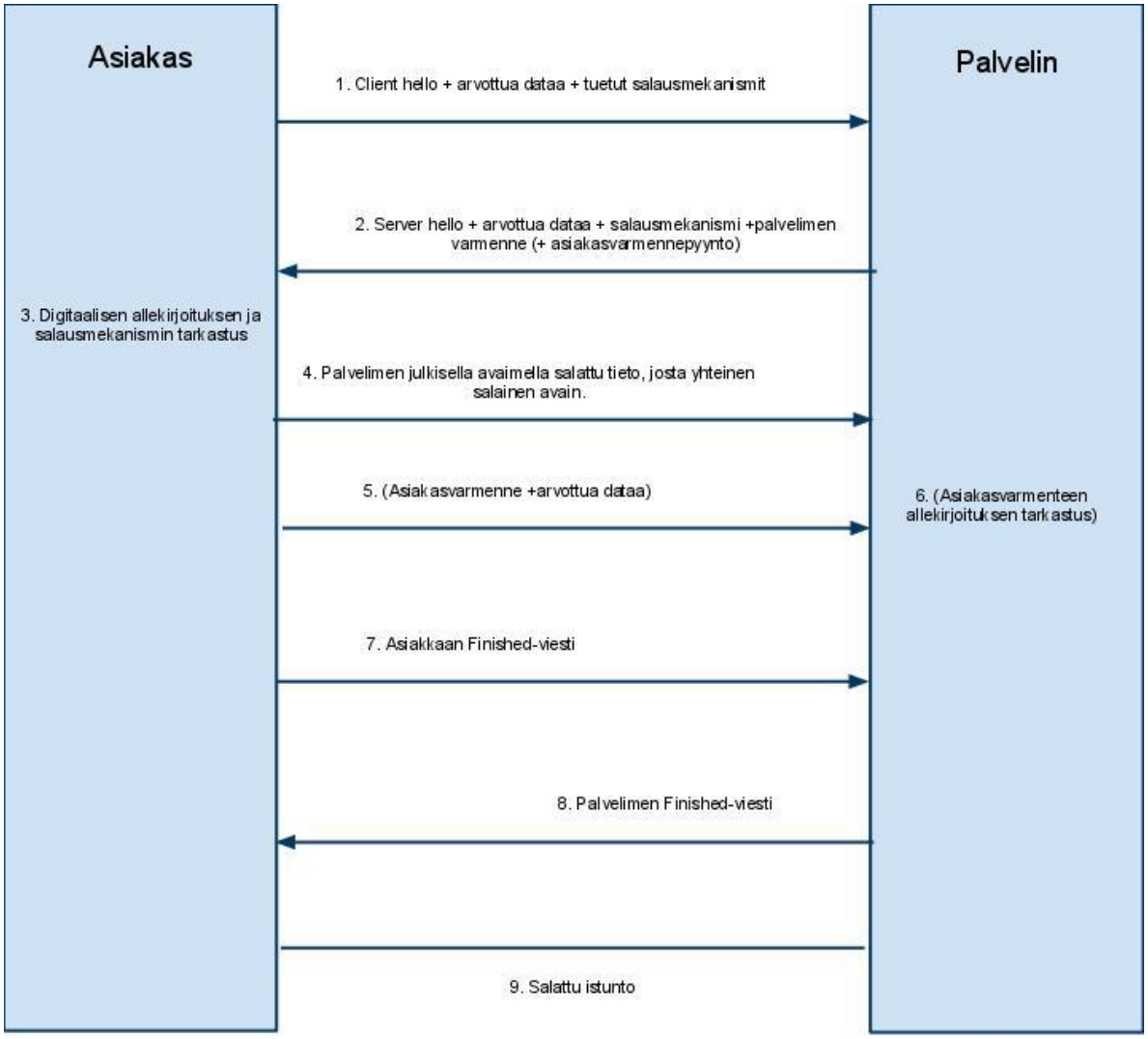


KUVA 2: Apachen ssl-moduulin enablointi.

Salattua yhteyttä muodostettaessa asiakkaan ja palvelimen välillä tapahtuu seuraavaa (Microsoft 2011):

1) Asiakas lähettää "client hello" -viestin palvelimelle. Tämä viesti pitää sisällään tiedon asiakkaan tukemista salausmekanismeista sekä arvotuilla arvoilla täytetyn tavun dataa, jota käytetään myöhemmissä vaiheissa tätä kahdenkeskistä keskustelua.
2) Palvelin vastaa "server hello" -viestillä. Viestissä on palvelimen valitsema salausmekanismi asiakkaan antamista vaihtoehdoista, toinen tavu arvottua dataa sekä "client certificate request" -pyyntö, jos sellainen on asetettu.

3) Asiakas tarkastaa digitaalisen allekirjoituksen sekä tarkastaa, että palvelimen valitsema salausmekanismi on hyväksyttävä.
4) Asiakas lähettää palvelimen julkisella avaimella salatun tiedon, joka on arvottua dataa, palvelimelle - tästä datasta kumpikin osapuoli laskee yhteisen salaisen avaimen (katso Julkisen avaimen salaus)
5) Jos palvelin vaatii asiakassertifikaattia, niin asiakas lähettää omalla salaisella avaimellaan kryptatun tiedon, joka on arvottua dataa ja asiakkaan käyttämä asiakassertifikaatti, palvelimelle.
6) Palvelin varmistaa asiakassertifikaatin digitaalisen allekirjoituksen.
7) Asiakas lähettää palvelimelle "finished" -viestin, joka on kryptattu arvotusta datasta muodostetulla salaisella avaimella, ja ilmoittaa kättelyn olevan asiakkaan osalta valmis.
8) Palvelin lähettää "finished"-viestin, joka on myös kryptattu arvotusta datasta muodostetulla salaisella avaimella, ja ilmoittaa kättelyn olevan palvelimen osalta valmis.
9) Istunto alkaa, ja sen aikainen asiakkaan ja palvelimen viestienvaihto on kryptattu yhteisellä salaisella avaimella.



KUVA 3: Salatun yhteyden muodostus.

4.2.2 Apache http -palvelinohjelmisto

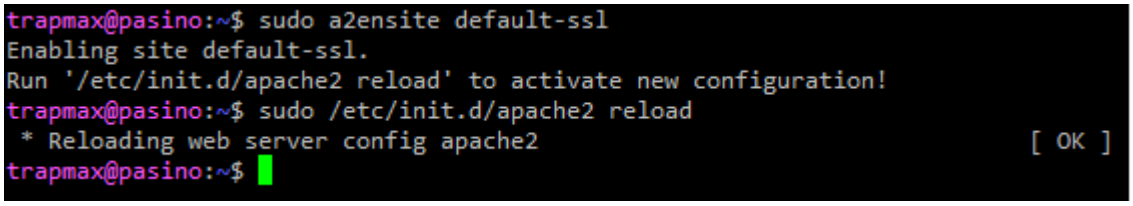
Tilaajayritys käyttää http-palvelimena Apache Software Foundationin Apache HTTP Server -ohjelmistoa, joka tarjoaa helposti asennettavan ja ylläpidettävän, avoimeen lähdekoodiin perustuvan ratkaisun. Perusominaisuuksiensa lisäksi Apache HTTP -palvelimeen on olemassa moduuleita, joita voi tarvittaessa etsiä, asentaa ja aktivoida erittäin helposti, kuten python-moduulin etsiminen ja asennus havainnollistaa (KUVA 4).

```
trapmax@pasino:~$ aptitude search libapache2-mod-python
p   libapache2-mod-python           - Python-embedding module for Apache 2
p   libapache2-mod-python-doc       - Python-embedding module for Apache 2 - doc
v   libapache2-mod-python2.6        -
trapmax@pasino:~$ sudo aptitude install libapache2-mod-python
Reading package lists... Done
Building dependency tree
Reading state information... Done
Reading extended state information
Initializing package states... Done
The following NEW packages will be installed:
  libapache2-mod-python
0 packages upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 124kB of archives. After unpacking 541kB will be used.
Writing extended state information... Done
Get:1 http://fi.archive.ubuntu.com/ubuntu/ lucid/main libapache2-mod-python 3.3.
1-8ubuntu2 [124kB]
Fetched 124kB in 0s (344kB/s)
Selecting previously deselected package libapache2-mod-python.
(Reading database ... 68650 files and directories currently installed.)
Unpacking libapache2-mod-python (from .../libapache2-mod-python_3.3.1-8ubuntu2_i
386.deb) ...
Setting up libapache2-mod-python (3.3.1-8ubuntu2) ...
 * Reloading web server config apache2                                 [ OK ]

Processing triggers for python-central ...
Reading package lists... Done
Building dependency tree
Reading state information... Done
Reading extended state information
Initializing package states... Done
Writing extended state information... Done

trapmax@pasino:~$
trapmax@pasino:~$ ls /etc/apache2/mods-enabled/
alias.conf            authz_user.load  dir.load          reqtimeout.load
alias.load            autoindex.conf   env.load          setenvif.conf
auth_basic.load       autoindex.load   mime.conf         setenvif.load
authn_file.load       cgid.conf        mime.load         ssl.conf
auth_pam.load         cgid.load        negotiation.conf  ssl.load
authz_default.load    deflate.conf     negotiation.load  status.conf
authz_groupfile.load  deflate.load     python.load       status.load
authz_host.load       dir.conf         reqtimeout.conf
trapmax@pasino:~$
```

KUVA 4: libapache2-mod-python -paketin etsiminen ja asennus Ubuntu-palvelimella.

Tärkein moduuli tietoturvallisen varmennekäytännön luonnissa on SSL-moduuli, joka mahdollista SSL v2/v3 ja TLS v1 -protokollatuet, ja täten myös salatun yhteydenoton käyttämisen halutuissa tapauksissa (Apache 2011). Pelkkä moduuli ei kuitenkaan riitä, sillä oletuksena Apache tarjoaa ulkoverkkoon vain tavallisen salaamattoman sivun. Sivustoon liittyvät konfiguroinnit täytyy muokata siten, että salausta käytetään.

Ubuntussa Apacheen liittyvät konfiguraatiot tehdään oletusarvoisesti hakemistossa /etc/apache2/ ja sen alihakemistoissa. Hakemistossa /etc/apache2/sites-available/ sijaitsee perusasennuksen jälkeen kaksi tiedostoa: default ja default-ssl, joista default-ssl pitää sisällään esimerkkiasetukset liittyen SSL-protokollan käyttöön. Default-ssl -sivu (site) on myös helppo ottaa käyttöön (Kuva 5).



KUVA 5: Default-ssl -konfiguraatiotiedoston käyttöönotto.

4.3 Wiki

4.3.1 Wiki käsitteenä

Kun tietojärjestelmää luodaan tyhjästä yrityksen tarpeisiin, on hyvä olla olemassa alusta, jonne luotu tieto voidaan yhteisesti koota. Tähän ongelmaan on ratkaisuna syntynyt verkkosivustoilla ylläpidettävät wikit. Wikin toimintaperiaate on yksinkertainen - jokainen voi halutessaan lisätä, muokata ja poistaa tietoa. Tämä tieto on kaikkien asianosaisten saatavilla. Kun työyhteisö kokoaa ohjeita, dokumentoi asiakkaiden tietoja tai luo työpäiväkirjoja, ne ovat helposti kaikkien yhteisöön kuuluvien ulottuvilla, ja tarpeen vaatiessa myös muokattavissa. Hyviä puolia wikin käytöstä voisi keksiä lähes loputtomiin. Vanhanaikaisiin toimintatapoihin verrattuna wikin

kanssa työskentely tuo työryhmälle keskitetyn alustan, mikä helpottaa koordinointia ja järjestelmällisyyttä.

4.3.2 Confluence wikin pohjana

Tilaajayrityksen käyttämä wikialusta on tällä hetkellä (26.4.2011) australialaisen ohjelmistoalayritys Atlassianin Confluence-yrityswiki. Confluence on nimenomaan yrityksille suunnattu wiki, joka omaa laajan markkina-alueen, joten toiminnot ovat täten varmasti suomalaiselle pienyritykselle riittäviä.

Confluencessa on monia erinomaisia piirteitä. Perustoimintoihin ja viimeiseksi päivitettyihin sivuihin pääsee helposti käsiksi, joka nopeuttaa wikin käyttöä. Erityisen maininnan ansaitsee "watch"-ominaisuus, jonka avulla mitä tahansa Confluencessa olevaa sisältöä (sivua, henkilöä, yms.) pystyy seuraamaan helposti. Seuranta toimii tilaajayrityksen tapauksessa sähköpostitse, eli muutosten tapahtuessa Confluence lähettää sähköpostiisi siitä tiedon. Muita tärkeitä ominaisuuksia ovat: versionhallinta, haku, blogisivut sekä "spaces"-ominaisuus.

Confluence päivittyy suhteellisen usein, joka varmistaa ajan tasalla olevien ominaisuuksien ilmestymisen. Dokumentaatio on hoidettu tietysti Confluencen kotisivuilla, ja se on jaoteltu versioiden mukaan (Atlassian 2011). Dokumentaatiot ovat erittäin kattavat, mutta monipuolisuutta lisää Atlassianin nopeasti toimiva raportointisysteemi, jonka kautta voi tehdä vikaraportteja tai vaikkapa toiminnallisuuspyynnön (feature request) (Atlassian 2011).

4.4 Webmail-ohjelmisto

Webmail-ohjelmisto on ohjelma, jonka avulla sähköpostipalvelimen tarjoamaa sisältöä voidaan käyttää selaimesta käsin. Selainkäyttö saattaa tulla kyseeseen silloin, kun itse sähköpostipalvelinta ei haluta laittaa julkiseen verkkoon, vaan se halutaan pitää poissa roskapostittajien sekä muiden haitantekijöiden ulottuvilta. Se-

lainkäyttö mahdollistaa yhtenäisemmän autentikointikäytännön ainakin niissä tapauksissa, joissa yrityksellä on jo turvalliseksi konfiguroitu WWW-palvelin.

4.5 IRC apuvälineenä

Normaalisti ongelmatilanteen sattuessa saatat kysyä asiaa työkaverilta tai tuttavalta. Nykypäivän tietointensiivisessä yhteiskunnassa ihmisillä ei ole tarpeeksi tarkkaa tietoa kaikilta aloilta. Tästä johtuen pitäisi löytää monien joukosta se, jolla sitä tietoa on. Ongelmaan on ratkaisu. Sen nimi on IRC.

IRC on pikaviestintäpalvelu, joka jakautuu eri verkkoihin (network), jotka puolestaan tarjoavat pääsyä eri kanaville (channel). Verkkoon päästäkseen on käytettävä jotain IRC-asiakasohjelmaa (IRC client), jolla voidaan ottaa yhteys IRC-verkkoon. Kun yhteys on luotu, voi kyseisessä verkossa liittyä kanavalle.

Apuvälineeksi IRC-kanavilla olevat ihmiset muuttuvat siinä vaiheessa, kun samanhenkiset, apua tarvitsevat ihmiset ovat samalla kanavalla. Kun runsaasti saman alan ihmisiä on kysymyksineen yhdellä kanavalla, he pystyvät auttamaan toisiansa - vieläpä reaaliajassa, joka ei onnistu esimerkiksi sähköpostikeskustelua käytäessä.

Aivan kuten opinnäytetyön aihe, pitää kysymys ensin suunnitella mahdollisimman yksityiskohtaiseksi. Kun kysymys on hiottu, se voidaan kysyä kanavalla. Sen jälkeen odotetaan kenen tahansa vastausta siihen. Eri vuorokauden aikaan ihmiset saattavat olla aktiivisia eri tavalla. Periaatteena kuitenkin se, että vastaus tulee jossain vaiheessa.

#ubuntu-server -niminen kanava irc.freenode.org -palvelimella on runsas avunlähde, kun käytetään Ubuntu-palvelinta. Muille palvelimille, erillisille ohjelmille, kuten myös ohjelmointikielille on omat kanavansa. Pienellä vaivalla on Googlen haulla helppo löytää kanava ja verkko, josta saa apua omaan ongelmaansa.

5 ALKUTILANNE

5.1 Verkon rakenne

Tilaajayrityksen verkko muodostuu palvelimista ja käyttäjien tietokoneista. Nämä on yhdistetty toisiinsa eri huoneissa sijaitsevien kytkimien avulla. Tarpeettomia portteja ei ole auki julkiselta puolelta.

Laitteet saavat IP-asetuksensa dnsmasq-ohjelman avulla. Tietojenkäsittelyyn liittyvä liikenne tapahtuu sisäverkossa, jonne on toteutettu lukuisia palveluja. Tärkeimmät palvelut ovat Confluence:lla toteutettu wiki ja IMAP-sähköpostipalvelin.

5.2 Palvelimet

Sisäverkossa on käytössä omia fyysisiä palvelimia. Käyttöjärjestelminä kyseisissä palvelimissa on Linux. Nämä palvelimet päivittyvät usein, joka on tietoturvan näkökulmasta tärkeä ominaisuus, sillä mahdollisiin tietoturva-aukkoihin tulee korjaavia päivityksiä nopeasti. Myös muita uusia ominaisuuksia on mahdollista ottaa käyttöön tarvittaessa käytössä olevan paketinhallintaohjelman avulla.

5.3 Sähköposti

Tilaajayrityksessä käytetään sähköpostipalvelimena avoimen lähdekoodin IMAP-palvelinta. Jotta tällä palvelimella sijaitsevia sähköposteja voidaan lukea, on käytettävä siihen tarkoitettua asiakasohjelmistoa. Työntekijöillä on käytössä Thunderbird-sähköpostiohjelma, jonka avulla yhteys IMAP-palvelimeen voidaan ottaa. Kuten useat muutkin palvelut, myös sähköposti tunnistaa käyttäjät LDAP-järjestelmän avulla.

Sähköpostin lähetykseen käytetään niin ikään avoimen lähdekoodin ohjelmaa. Saapuvan postin palvelimen sijaitessa sisäverkossa, täytyy reunapalvelimella olla sähköpostin välityspalvelu.

5.4 Wiki

Käytössä oleva ohjelmisto on Confluence-ohjelmisto. Palvelu ylläpidetään yrityksen sisäverkossa, ja se käyttää myös LDAP-järjestelmää. Dnsmasq-ohjelma ohjaa sisäverkossa tapahtuvat, wikiosoitteeseen tulevat kyselyt kyseisen ohjelman sivuille.

Aloittaessa työtä, ei sisäinen wiki ollut kaikkien työntekijöiden käytössä, vaan sitä käytettiin satunnaisesti eri henkilöiden toimesta. Itse olin työsuhteeni alusta lähtien käyttänyt wikiä erinäisiin toimiin: työpäiväkirja blogimuodossa, ohjeistusten tekeminen sille kuuluvan kategorian alle sekä valvonnassa käytettyjen sivujen seuraaminen olivat näistä tärkeimmät.

6 TYÖN KULKU

6.1 Toteutus- ja ohjelmistovalinnat

Jotta yrityksen työntekijöillä olisi mahdollisuus käyttää sisäverkossa sijaitsevia palveluita ulkoverkosta käsin, on tälle toiminnallisuudelle ensin määriteltävä kriteerit. Käsiteltävät tiedot pitävät sisällään arkaluontoisia, asiakkaisiin kohdistuvia tietoja, joten tiedon on ehdottomasti kuljettava turvallisesti myös etäkäyttötapauksessa. Tiedon turvallisen kuljettamisen lisäksi on taattava sitä käyttävän henkilön oikeellisuus - ominaisuus, joka voidaan saavuttaa varmenteiden avulla.

Suurin päätös työn alkuvaiheessa oli etäkäyttötavan valitseminen VPN-tekniikoiden ja sertifikaatteihin perustuvan tunnistautumisen välillä. Päädyimme ratkaisuun, joka antaa käyttäjälle rajatummat oikeudet yrityksen resursseihin etäkäytön aikana - sertifikaatteihin.

Tärkein etäkäytettävä resurssi on yrityksen sisäinen wiki, joka toimii Apache Tomcat -websovelluksena. Tämä oli sinänsä hyvä asia, sillä pystyisin käyttämään Apachea välityspalvelimen (proxy server) ominaisuudessa vain asentamalla proxy-moduulin välittämään wikisivuja. Tämä valinta johti siihen, että etäkäytettävien palvelujen tuli olla resursseja tai sovelluksia, jotka on toteutettu WWW-muodossa. Tietoturvallisuuden näkökulmasta tämä on vain hyvä asia, sillä pienempi määrä palveluja tarkoittaa pienempää määrää mahdollisia heikkouksia.

Perustekniikan ollessa valittuna, tuli kysymykseen toisen työntekijöille tärkeän resurssin, sähköpostin, etäkäyttömahdollisuus. Mitä vähemmän reunapalvelimen portteja on auki, sitä vähemmän hyökkääjillä on mahdollisuuksia löytää haavoittuvuuksia järjestelmästämme ulkoapäin. Tähän ajatukseen perustuen tuli webmail-ohjelman valinta seuraavana eteen. Mikä ilmainen ohjelma olisi järkevin valinta? IRC-konsultoinnin jälkeen, ja tutkittuani muutaman ohjelman ominaisuuksia, ehdo-

tin Roundcubea sen ominaisuuksien (Roundcube 2010) takia. Ohjelman ulkoasu ja ominaisuudet vastasivat nykystandardeja, joten ehdotukseni menikin läpi. Isompia perustavan tason valintoja ei ollut enää edessä.

6.2 OpenSSL palveluiden turvaratkaisuna

OpenSSL on yhteistyöprojektina kehitettävä ohjelma, joka pitää sisällään kokoelman työkaluja SSL ja TLS-tekniikoiden implementoinniksi, sekä vapaaehtoisten ylläpitämän yleiskäyttöisen kryptografiakirjaston (OpenSSL 2011). Näitä työkaluja hyväksikäyttämällä toteutin ensimmäisen varmennehierarkian.

6.2.1 CA-sertifikaatin luominen

Kun varmentajatahoa luodaan, se tarvitsee oman sertifikaatin, mikä pitää sisällään tiedot kyseisestä varmentajasta. Linux-järjestelmässä CA-tiedot on järkevä sijoittaa oletushakemistoon /etc/ssl/ -hakemiston alle, koska tämä hakemisto pitää sisällään jo valmiiksi sertifikaatteihin liittyvää dataa, kuten luotetut juurivarmentajat /etc/ssl/certs/ -hakemistossa. OpenSSL:n konfigurointitiedostoa muokkaamalla korjataan oletusasennuksen parametrien sisältö kohdejärjestelmään sopivaksi. Kun tiedot on muokattu oikein, voidaan tätä konfigurointitiedostoa käyttää OpenSSL-komennon kanssa muodostamaan itse allekirjoitettu ca-sertifikaatti. Tämä luominen OpenSSL-komennolla tapahtuu seuraavien vaiheiden kautta (Kuva 6):

```
trapmax@pasino:~$ sudo mkdir -p /etc/ssl/CA
trapmax@pasino:~$ sudo sh -c "echo '01' > /etc/ssl/CA/serial"
trapmax@pasino:~$ sudo touch /etc/ssl/CA/index.txt
trapmax@pasino:~$ sudo openssl req -new -x509 -extensions v3_ca -keyout /etc/ssl
/private/cakey.pem -out /etc/ssl/CA/cacert.pem -days 3650
Generating a 1024 bit RSA private key
..............................++++++
.............++++++
writing new private key to '/etc/ssl/private/cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:FI
State or Province Name (full name) [Some-State]:Pirkanmaa
Locality Name (eg, city) []:Tampere
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Testi Oy
Organizational Unit Name (eg, section) []:Markkinointi
Common Name (eg, YOUR name) []:Taneli Mäenpää
Email Address []:taneli.maenpaa@cs.tamk.fi
trapmax@pasino:~$
```

KUVA 6: CA-sertifikaatin luominen.

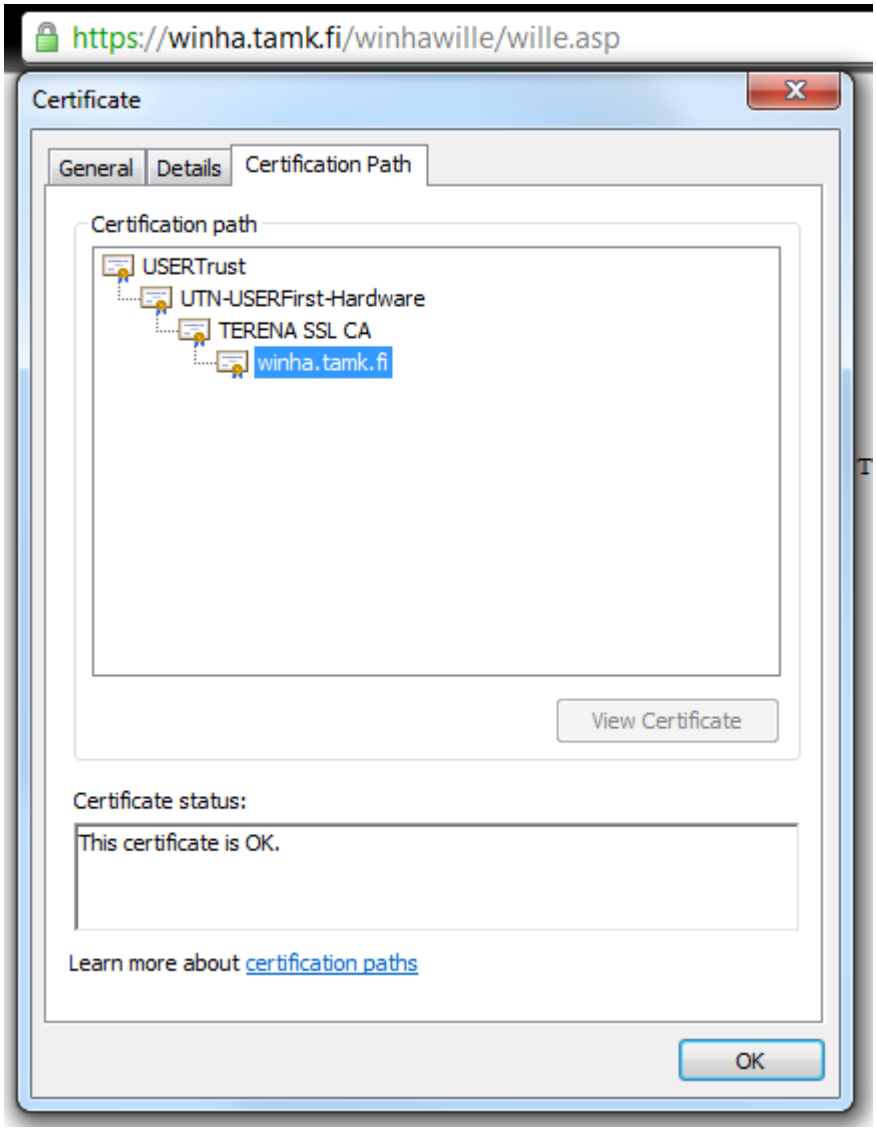
1) Aluksi luodaan hakemisto CA-tiedoille. Tämä hakemisto on loogista sijoittaa /etc/ssl/ -hakemiston alle.

2) Luodaan serial-niminen tiedosto, joka pitää sisällään rivin "01". Tämä tiedosto pitää sisällään juoksevan numeron kyseisen CA:n myöntämien sertifikaattien määrästä.

3) Luodaan index.txt -tiedosto, johon OpenSSL päivittää sertifikaattitietokantansa siihen tullessa muutoksia.

4) Luodaan OpenSSL-komennolla uusi CA-sertifikaatti, määritellään sertifikaatin ja avaimen ulostulopolut sekä sertifikaatin voimassaoloaika.

5) Tämän jälkeen täytetään itse sertifikaattiin tulevat kohdat kyselyjärjestyksessä. Tärkeä kohta on "Common Name", joka määrittelee varmennetta käyttävän tahon nimen. Palvelintapauksissa tähän kohtaan laitetaan yleensä palvelimen nimi, ja palvelun ollessa kyseessä nimenä käytetään sen palvelun nimeä. Tämä käytäntö mahdollistaa palvelun tunnistamisen sen nimen perusteella: näin on asian laita esimerkiksi winha.tamk.fi -sivuston käyttämässä sertifikaatissa (Kuva 7).



KUVA 7: Esimerkki palvelun nimestä verrattuna sertifikaatin nimeen

6.2.2 CSR ja allekirjoitus

Kun CA on laitettu toimintakuntoon, voidaan keskittyä palvelujen ja henkilöiden henkilökohtaisten sertifikaattien luomiseen. Tämä tapahtuu luomalla yksityinen avain ja tekemällä allekirjoituspyyntö (CSR) (Kuva 8).

```
trapmax@pasino:~$ openssl genrsa -des3 -out www.testi.fi.key 2048
Generating RSA private key, 2048 bit long modulus
.............+++
.....................................................................+++
e is 65537 (0x10001)
Enter pass phrase for www.testi.fi.key:
Verifying - Enter pass phrase for www.testi.fi.key:
trapmax@pasino:~$ openssl req -new -key www.testi.fi.key -out www.testi.fi.csr
Enter pass phrase for www.testi.fi.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:FI
State or Province Name (full name) [Some-State]:Pirkanmaa
Locality Name (eg, city) []:Tampere
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Testi Oy
Organizational Unit Name (eg, section) []:www-osasto
Common Name (eg, YOUR name) []:www.testi.fi
Email Address []:admin@testi.fi

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

KUVA 8: RSA-avaimen ja allekirjoituspyynnön luominen OpenSSL:n avulla.

Tämä ei kuitenkaan vielä riitä, sillä pyyntö pitää lähettää vielä allekirjoitettavaksi varmentajataholle, joka lähettää allekirjoitetun sertifikaatin sitä anovalle taholle takaisin. Tämä vaihe on se, jossa varmentajan on todettava hakijatahon oikeellisuus (Kuva 9). Ennen pyynnön allekirjoitusta on tiedettävä, kenen käyttöön kyseinen sertifikaatti on tulossa.

```
trapmax@pasino:~$ sudo openssl ca -in www.testi.fi.csr -out www.testi.fi.pem
Using configuration from /usr/lib/ssl/openssl.cnf
Enter pass phrase for /etc/ssl/private/cakey.pem:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 1 (0x1)
        Validity
            Not Before: May 17 11:55:26 2011 GMT
            Not After : May 16 11:55:26 2012 GMT
        Subject:
            countryName               = FI
            stateOrProvinceName       = Pirkanmaa
            organizationName          = Testi Oy
            organizationalUnitName    = www-osasto
            commonName                = www.testi.fi
            emailAddress              = admin@testi.fi
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                EC:1C:17:1A:F6:3E:E5:20:83:B9:21:8B:8B:8C:3F:AC:2C:E8:1E:BC
            X509v3 Authority Key Identifier:
                keyid:20:B9:0F:3A:DA:4C:03:C2:FB:35:A1:45:8F:D2:B3:D0:E7:0C:84:E
8

Certificate is to be certified until May 16 11:55:26 2012 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
trapmax@pasino:~$
```

KUVA 9: CSR:n allekirjoitus OpenSSL:n avulla.

Tilaajayrityksen palvelimella varmentajataho sijaitsee samassa järjestelmässä kuin varmenteita hakevat osapuoletkin, joten pyyntöjä tekee ylläpitäjä, joka myös allekirjoittaa ne. Tämä helpottaa muiden työntekijöiden toimintaa.

6.2.3 Sulkulista ja sen päivitys

Kun varmenteita myönnetään, täytyy olla myös keino estää vanhentuneen tai muulla tavalla haitalliselle taholle altistuneen sertifikaatin kumoamiseksi. Varmentajatahon hallinnan kannalta tilanne on ratkaistu kumoamisen (revocation) avulla.

Pelkkä kumoaminen ei vielä riitä kertomaan muille, mitkä sertifikaatit on kumottu, vaan varmenteista on pidettävä kirjaa. Sulkulista pitää sisällään tiedot sertifikaateista, joiden oikeudet on kumottu. Tässä kohtaa tulee mukaan OpenSSL:n helppokäyttöisyys: sulkulistoja voidaan luoda ja päivittää ja niitä on mahdollista myös tutkia (Kuva 10).

```
trapmax@pasino:~$ sudo openssl ca -gencrl -out /etc/ssl/crl.pem
Using configuration from /usr/lib/ssl/openssl.cnf
Enter pass phrase for /etc/ssl/private/cakey.pem:
trapmax@pasino:~$ sudo openssl crl -in /etc/ssl/crl.pem -text -noout
Certificate Revocation List (CRL):
        Version 2 (0x1)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: /C=FI/ST=Pirkanmaa/L=Tampere/O=Testi Oy/OU=Markkinointi/CN=Tanel
i M\xC3\xA4enp\xC3\xA4\xC3\xA4/emailAddress=taneli.maenpaa@cs.tamk.fi
        Last Update: May 17 12:39:01 2011 GMT
        Next Update: Jun 16 12:39:01 2011 GMT
        CRL extensions:
            X509v3 CRL Number:
                8
Revoked Certificates:
    Serial Number: 01
        Revocation Date: May 17 12:10:41 2011 GMT
    Serial Number: 02
        Revocation Date: May 17 12:36:07 2011 GMT
    Signature Algorithm: sha1WithRSAEncryption
        4a:40:77:14:6f:24:f1:72:bb:22:76:76:b6:ee:ca:e8:1e:5f:
        90:7d:67:0f:f1:3d:53:5f:0c:ac:a1:02:7f:7c:59:9b:ca:0a:
        03:f9:11:61:02:ec:1f:b9:da:c0:30:2f:e9:21:ca:0c:8e:d9:
        bf:12:44:a5:6d:de:b2:20:69:f7:3a:f5:3d:8a:9a:e5:1d:69:
        02:f9:55:4e:f7:27:93:b2:6f:5f:ca:34:5d:97:30:2e:c8:a8:
        91:95:46:d3:2d:70:7d:f2:1c:eb:9d:c4:16:0f:95:a0:66:a5:
        f5:68:ba:99:07:ed:11:31:ca:ed:40:4a:45:83:89:d0:7a:b0:
        fc:41
trapmax@pasino:~$
```

KUVA 10: Sulkulistan päivitys ja sen tutkiminen.

6.2.4 Asiakassertifikaatti selainta varten

Etäkäytön kannalta työntekijöille tärkein osuus on PKCS#12-muotoinen sertifikaatti, jota voidaan käyttää selaimeen asennettuna tunnistautumiseen. OpenSSL:n avulla jo tehdystä, PEM-muotoisesta sertifikaatista ja sitä suojaavasta RSA-avaimesta voidaan tehdä PKCS#12-muotoinen erittäin helposti (Kuva 12). Tämä sertifikaatti voidaan myöhemmin asentaa selaimeen tai Windows-ympäristössä varmennevarastoon. Linux-maailmassa tarjotaan mahdollisuus hallita avaimia ja varmenteita keytool-ohjelmalla (Ubuntu 2011).

```
trapmax@pasino:~$ sudo openssl pkcs12 -export -in taneli-user.pem -inkey www.tes
ti.fi.key -out taneli-user.p12
Enter pass phrase for www.testi.fi.key:
Enter Export Password:
Verifying - Enter Export Password:
trapmax@pasino:~$
```

KUVA 12: Esimerkki PKCS#12 -asiakassertifikaatin tekemisestä.

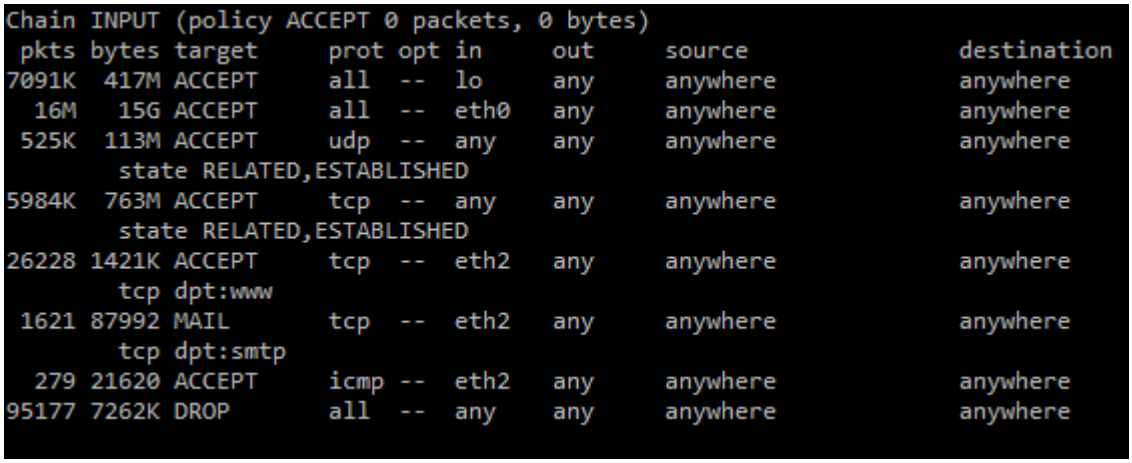
6.3 Portin avaaminen

Kuten jo aiemmin mainitsin, ei tilaajayrityksen palvelimella ole ulkoverkosta päin olevia avoimia portteja kuin kaksi: 80 salaamattomalle http-liikenteelle, ja 25 saapuvaa sähköpostia varten. Jotta https-yhteys olisi mahdollista ottaa, tarvitsee muokata avoimien porttien käytäntöä. On tärkeää muistaa sääntöjen järjestys - ensin mainittu vaikuttaa ensiksi. Porttiohjaus voidaan toteuttaa esimerkiksi Iptables-ohjelmalla, jonka avulla seuraavaksi demonstroin portin avaamisen.

Ensiksi aloitan ketjujen tutkimisen (Kuva 12). Huomattava on siis järjestys, jossa käsitellään tulevaa dataa. Kun tutkitaan listaa, voidaan huomata sisääntulevaan liikenteeseen (Chain INPUT) vaikuttavat säännöt:

1) Loopback-liittimeen tuleva liikenne sallitaan.
2) Eth0-liittimeen tuleva liikenne sallitaan

3) Mihin tahansa liittimeen tuleva, jo muodostettu UDP-yhteys sallitaan
4) Mihin tahansa liittimeen tuleva, jo muodostettu TCP-yhteys sallitaan
5) Eth2-liittimeen tuleva TCP-yhteys WWW-porttin (dpt:www) sallitaan
6) Eth2-liittimeen tuleva TCP-yhteys SMTP-porttiin (dpt:smtp) sallitaan
7) Eth2-liittimeen tuleva ICMP-liikenne sallitaan
8) Kaikki muu liikenne evätään.



```
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
7091K  417M ACCEPT     all  --  lo     any     anywhere             anywhere
  16M   15G ACCEPT     all  --  eth0   any     anywhere             anywhere
 525K  113M ACCEPT     udp  --  any    any     anywhere             anywhere
        state RELATED,ESTABLISHED
5984K  763M ACCEPT     tcp  --  any    any     anywhere             anywhere
        state RELATED,ESTABLISHED
26228 1421K ACCEPT     tcp  --  eth2   any     anywhere             anywhere
        tcp dpt:www
 1621 87992 MAIL       tcp  --  eth2   any     anywhere             anywhere
        tcp dpt:smtp
  279 21620 ACCEPT     icmp --  eth2   any     anywhere             anywhere
95177 7262K DROP       all  --  any    any     anywhere             anywhere
```

KUVA 12: Iptables sääntöjen listaus. Portti 443 evätty.

Jotta tulevaa https-liikennettä ei evättäisi, se täytyy lisätä listalle oikeaan kohtaan. Tässä tapauksessa järkevä paikka on WWW-yhteyden sallivan säännön jälkeen, eli listan sijalle kuusi, koska numerointi alkaa ensimmäisestä rivistä numerolla 1. Lisäsin ulkoverkosta tulevan liikenteen https-porttiin 443. Tämän jälkeen varmistin säännön olemassa olon listassa (Kuva 13) ja tallensin asetukset. Näiden toimien jälkeen ulkoverkosta oli mahdollista päästä käsiksi porttiin 443.

```
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
7096K  418M ACCEPT     all  --  lo     any     anywhere             anywhere
  16M   15G ACCEPT     all  --  eth0   any     anywhere             anywhere
 526K  114M ACCEPT     udp  --  any    any     anywhere             anywhere
        state RELATED,ESTABLISHED
5984K  763M ACCEPT     tcp  --  any    any     anywhere             anywhere
        state RELATED,ESTABLISHED
26237 1421K ACCEPT     tcp  --  eth2   any     anywhere             anywhere
        tcp dpt:www
15871  784K ACCEPT     tcp  --  any    any     anywhere             anywhere
        tcp dpt:https
 1622 88048 MAIL       tcp  --  eth2   any     anywhere             anywhere
        tcp dpt:smtp
  279 21620 ACCEPT     icmp --  eth2   any     anywhere             anywhere
95478 7327K DROP       all  --  any    any     anywhere             anywhere
```

KUVA 13: Iptables sääntöjen listaus. Portti 443 sallittu.

6.4 Apache

Kun portti 443 oli avattu, niin piti suorittaa salatun yhteyden testaaminen. Apache käyttää oletusarvoisesti sertifikaattia, joka asentuu ssl-cert -paketin yhteydessä, joka voidaan todeta default-ssl -oletuskonfiguraatiotiedostoa tutkittaessa (Kuva 14, kommentoidut rivit alkavat #-merkillä). Ssl-cert -paketti puolestaan asentuu automaattisesti OpenSSL-paketin asennuksen yhteydessä.

```
#   SSL Engine Switch:
#   Enable/Disable SSL for this virtual host.
SSLEngine on

#   If both key and certificate are stored in the same file, only the
#   SSLCertificateFile directive is needed.
SSLCertificateFile    /etc/ssl/certs/ssl-cert-snakeoil.pem
SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key

#   Server Certificate Chain:
#   Point SSLCertificateChainFile at a file containing the
#   concatenation of PEM encoded CA certificates which form the
#   certificate chain for the server certificate. Alternatively
#   the referenced file can be the same as SSLCertificateFile
#   when the CA certificates are directly appended to the server
#   certificate for convinience.
#SSLCertificateChainFile /etc/apache2/ssl.crt/server-ca.crt
```

KUVA 14: Ote default-ssl -tiedostosta

Seuraava vaihe oli aktivoida Apache siten, että default-ssl -tiedoston sisältämät konfiguroinnit tulevat käyttöön. Tämä tapahtui komennoilla:

" a2ensite default-ssl"
" /etc/init.d/apache2 reload"

Kun tämä oli tehty voitiin salattu yhteys testata selaamalla palvelimen WWW-sivuille https-alkuisena. Tässä vaiheessa ei turvattavia palveluja vielä ollut välitettynä, joten testaaminen oli turvallista.

Testaamisen jälkeen suoritin konfiguraatiomuutoksen, jonka avulla Apache-palvelimelle vaihdettiin yrityksen oma varmenne - tämä tapahtui vain tiedostopolkua muuttamalla. Muut sertifikaatteihin liittyvät, varmenteita käytettäessä tärkeät konfiguraatioparametrit Apachen asetustiedostossa ovat taulukossa 1.

Parametri	Selitys	Arvo	Arvon selitys
SSLEngine	Kertoo, onko SSL-moottori päällä	On	Moottori on päällä
SSLVerifyClient	Tarvitaanko asiakassertifikaatin tarkastusta	require	Asiakkaan täytyy esittää sertifikaatti
SSLCertificateFile	Missä sertifikaatti sijaitsee isäntäkoneella	/etc/ssl/cert.pem	Tiedostopolku käytettävää sertifikaattia varten
SSLCACertificateFile	Missä juurisertifikaatti sijaitsee isäntäkoneella	/etc/ssl/ca.pem	Tiedostopolku juurisertifikaatille
SSLCARevocation-File	Missä sulkulista sijaitsee isäntäkoneella	/etc/ssl/crl.pem	Tiedostopolku sulkulistalle

TAULUKKO 1: Tärkeät parametrit Apachen asetustiedostossa salatun liikenteen käytössä

Kun tarvittavat asetusmuutokset olivat voimassa, pääsi salatuille sivuille enää esittämällä omalla juurisertifikaatilla allekirjoitettu käyttäjäsertifikaatti. Tämä oli haluttu lopputulos.

Välityspalvelimena toimiminen onnistuu Apachen proxy-moduulin avulla. Palvelut eivät näy ulkoverkkoon ennen niiden välittämistä sinne. Confluencen tarjoamalle palvelimelle täytyy liikenne ohjata välityspalvelinta käyttäen.

Apache Tomcat -javasovelluspalvelinta käytettäessä voidaan käyttää AJP-protokollaa, joka välittää pyynnöt WWW-palvelimelta (Apache) Apache Tomcat -sovelluspalvelimelle. AJP-protokollaa varten on Apachelle oma moduuli: proxy_ajp, joka voidaan ottaa käyttöön komennolla:

" a2enmod proxy_ajp"

Moduulin käyttöönoton jälkeen täytyy vielä muokata asetustiedostoa siten, että uudelleenohjaus halutulla tavalla. Tilaajayrityksen tapauksessa haluamme palveluista Confluencen avulla toteutettuun wikin välittymään yrityksen juurisertifikaatilla myönnettyjen varmenteiden omistajille. Tämä tarkoittaa välityspalvelinasetusten tekemistä pelkästään salatun yhteyden taakse, joten muokattavat asetukset ovat **p**alvelimella sijaitsevassa default-ssl -tiedostossa.

On tärkeä muistaa sivustoa konfiguroidessa, että Apachen direktiivit: "Location" ja "Directory" vaikuttavat eri asioihin. "Location" vaikuttaa selaimessa näkyvään osoitteeseen, kun taas "Directory" vaikuttaa palvelinkoneella olevaan hakemistoon. Kummallekin voidaan asettaa samantapaisia parametrejä, kuten sallitut IP-osoitteet tai käyttäjät.

Välityspalvelinasetuksia tehdessä tein muutokset "Location" -direktiivin alle, jolloin selattaessa salatulla yhteydellä osoitteeseen saavutetaan haluttu uudelleenohjaus. Välityspalvelinasetukset selityksineen ovat taulukossa 2.

Parametri	Selitys
ProxyPass	Kääntää välitettävän koneen osoitteet välittävän koneen muotoon
ProxyPassReverse	Muokkaa välittävälle koneelle tulevat osoitteet välitettävän koneen muotoon
ProxyPassReverseCookieDomain	Muokkaa palvelimen lähettämiä tietoja evästeissä

TAULUKKO 2: Tärkeät välityspalvelinasetukset.

Näiden muutosten voimaantulon jälkeen tilaajayrityksen sisäistä yrityswikiä voidaan käyttää ulkoverkosta käsin, jos käyttäjällä on kyseisen yrityksen juurisertifikaatin varmistama käyttäjäsertifikaatti asennettuna.

7 TULEVAISUUS

Vaikka OpenSSL:n avulla toteutettu PKI-järjestelmä on helppo hallita, vaatii se jatkuvaa ylläpitoa. Työntekijöistä noin puolet käyttävät järjestelmän palveluita etänä, joka tarkoittaa runsaasti mahdollisuuksia jonkun varmenteen joutumisesta vääriin käsiin. Myös työntekijöiden puutteellinen suhtautuminen tietoturvaan yleensä ja virallisen ohjeistuksen puuttuminen tietoturvaa koskevissa asioissa saa aikaan sen, että sertifikaateille annetaan kolmen kuukauden voimassaoloaika. Kahdeksan henkilön sertifikaattien uudelleen luominen näin usein saattaa vaikuttaa turhalta, mutta ehkäisee arkaluontoisiin dokumentteihin pääsyn tehokkaasti.

Ylläpidon kannalta uudelleenluomisprosessi on turhauttava. Varsinkin tämän hetken tilanteessa, sillä ylläpitäjä on vastuussa myös allekirjoituspyyntöjen tuottamisesta sekä salasanojen luomisesta. Yrityksessä on sentään käytössä LDAP-järjestelmä, jonka kautta voitaisiin ylläpitää myös käyttäjien varmenteita.

Saatuani valmiiksi nykyisen varmennekäytännön tuli seuraava pohdinta näin ollen ajankohtaiseksi: Kuinka tehdä PKI-järjestelmästä vielä toimivampi? Tämä kysymys mielessä tutustuin järjestelmään nimeltä EJBCA, josta seuraavassa tarkemmin.

EJBCA on avoimen lähdekoodin alustariippumaton yritystason PKI-ratkaisu, joka on toteutettu JEE-teknologian avulla (EJBCA 2011). EJBCA tarvitsee toimiakseen JBoss-sovelluspalvelimen, Apache Ant -työkalun sekä JDK-alustan.

Oma työkoneeni toimi testialustana, jonka avulla pystyin kokeilemaan EJBCA:n asennusta, konfigurointia ja käyttöä. Kun EJBCA:ta asennetaan, on huomionarvoista oletusarvoisten asetusten muokkaaminen. Nämä tiedostot pitävät sisällään asennuksen jälkeen itse ohjelmassa näkyvät tiedot - onko esimerkiksi haluttua luoda DemoCA-niminen CA, vai halutaanko sille joku todellisuutta vastaava nimi?

Asennuksen jälkeen itse EBCA:n käynnistäminen onnistui ja pääsin käsiksi julkisen puolen sivustoon localhost-osoittessa: localhost/ejbca. Jotta ylläpitopuolelle pääsee, on asennettava superadmin.p12. Tämän sertifikaatin avulla voidaan selata salatulle puolelle, josta EJBCA:ta voidaan hallita.

EJBCA:n konfigurointimahdollisuudet ovat erittäin laajat. Toiminnot on jaoteltu eri osioihin isompien kokonaisuuksien mukaan ja nämä ovat selkeästi näkyvissä aloitussivulla (EJBCA 2011). CA-toimintoja varten on omat valikkonsa, kuten myös RA-toimintoja. Näiden kahden valikon alta löytyvätkin tärkeimmät ominaisuudet PKI-järjestelmän luontivaiheessa.

EJBCA mahdollistaa PKI-hierarkian luomisen profiilien avulla. Profiileja voi tehdä erikseen varmentajille sekä loppukäyttäjille (End user). Profiileihin voi yhdistää julkaisijan - toiminto, joka tilaajayrityksen tapauksessa olisi tähdellinen, sillä sertifikaatit voisi julkaista suoraan LDAP-järjestelmään. Testauksen aikana päädyinkin kokeilemaan LDAP-julkaisua, joka toimi oletetulla tavalla.

Suurin yksittäinen havaittu puute EJBCA:n testikäytössä oli salasanojen käsittely. Valmiin LDAP-ratkaisun olemassaoloa ei voinut yhdistää itse EJBCA-järjestelmän kirjautumiseen. Toisin sanoen EJBCA:ta ei saanut autentikoimaan käyttäjää LDAP:n kautta.

Muutaman testiviikon jälkeen olen varma, että tilaajayrityksessä on tulevaisuudessa käytössä EJBCA yrityksen PKI-järjestelmänä, sillä sen helppokäyttöinen web-liittymä mahdollistaa jokaisen työntekijän omatoimisen sertifikaattihallinnan niiltä osin, kuin se on tarpeellista. Myös sulkulistojen päivitys ja varmenteiden kumoaminen ovat vain napinpainalluksen päässä - näin myös ylläpitäjän tehtävät helpottuvat OpenSSL-järjestelyyn verrattuna.

8 TUTKIMUSMENETELMÄT

Tämän työn aikana käyttämäni tutkimusmetodit ovat suurelta osin tekstianalyysin perustuvia. On kuitenkin muutamia tilanteita, joissa toisenlainen lähestymistapa on ollut tarpeen. Pääpaino pysyy kuitenkin eri ohjelmistojen ja komentojen manuaalisivujen ja niiden verkkosivujen dokumentoinnin tutkimisessa.

Perustilanteessa minulle annetaan tehtävä, joka pitää suorittaa. Käytettävät ohjelmistot ja menetelmät ovat vapaat. Aikaisempien kokemuksien puuttuessa, on osattava käyttää jo olemassa olevia resursseja tehokkaasti hyödyksi. Tässä kohtaa Linux-maailmassa vallitseva tapa kirjoittaa ohjelmille manuaalisivuja tulee erittäin hyödyllisesti esille, sillä oudomman komennon tullessa vastaan, voi aina kirjoittaa "man komento". Suurin osa ajasta uuteen ohjelmaan, esimerkiksi OpenSSL, liittyen menee tutustuen sen manuaalisivuun. Useat ohjelmantekijät tarjoavat kyseisten sivujen lisäksi vielä lisäinformaatiota erinäisten UKK-sivujen (usein kysytyt kysymykset) merkeissä.

Toisenlainen tapa lähestyä ongelmaa on IRC. Ainakin Ubuntu-yhteisöllä on runsaasti kanavia, joihin kysymyksensä voi asettaa. Itse olen työn puitteissa viettänyt useita antoisia hetkiä kysyen ja vastaten #ubuntu-server -kanavalla irc.freenode.org -palvelimella. Suurin hyöty tämänkaltaisessa dynaamisessa interaktiossa muiden saman alan ihmisten kanssa on se, että joku on jo sinun lisäksi käynyt samat ongelmat läpi aikaisemmin.

Kokonaisuutena opinnäytetyöni aikana käyttämistäni tavoista suurimman merkityksen saa, ja vielä helposti, konstruktiivinen tapa tutkia asioita. Ongelman ilmenemisen jälkeen siihen aletaan etsiä ratkaisua. Jos ratkaisu löytyy, niin sitä testataan ennen käyttöönottoa. Jos testauskin vielä onnistuu, niin ratkaisu voidaan toteuttaa.

9 POHDINTA

Toimeksiannon tavoitteena oli saada tilaajayritykselle toimiva ratkaisu sisäverkossa sijaitsevien palveluiden, joista tärkeimpänä yrityksen wiki, tietoturvalliseen etäkäyttöön. Tämä siksi, että työntekijöillä olisi mahdollisuus päästä käsiksi tarvittaviin dokumentteihin silloin, kun he ovat työtehtävissä toimistotilojen ulkopuolella. Ratkaisuvaihtoehdoista tuli, ainakin näin alkuun, valituksi OpenSSL-järjestelmä yhdistettynä Apache http -palvelimen SSL-yhteysominaisuuksiin. Tavoitteessa onnistuttiin hienosti johtuen pääosin ratkaisun yksinkertaisuudesta.

Opinnäytetyön työosuuden tuloksena tilaajayritys on ottanut aimo askeleen kohti palveluiden yhdistämistä, sillä Confluence-yrityswiki on alustana erittäin monipuolinen. Sen avulla voidaan luoda raportteja asiakkaan tiloissa tehdyistä toimenpiteistä reaaliajassa, päivittää tietoja siten, että vanhat versiot pysyvät tallessa ja liittää dokumentteja suoraan artikkeleihin - tästä kaikesta voi vielä tarvittaessa saada ilmoituksen suoraan sähköpostiin, joka auttaa yrityksen eri työntekijöitä saamaan informaatiota kulkemaan nopeammin. Uuteen ohjelmistoon tutustuminen on aina oppimiskynnyksen koettelemista, mutta uskon, että työntekijät ovat kaikki kykeneviä toimimaan yhteisen hyvän puolesta.

Valittu menetelmä tarvitsee tarkkaavaisuutta, sillä tällä hetkellä mikään ei vielä ole automatisoitua - jokainen varmenne ja sulkulista pitää luoda, päivittää ja kumota ylläpitäjän toimesta. Tähän joku voisi väliaikaisena ratkaisuna ehdottaa shell-skriptejä, mutta toimintatapa on tällä hetkellä sellainen, että käyttäjän interaktiota tarvitaan.

Tilaajayrityksen käyttämä domain-palvelu pitää sisällään mahdollisuuden lisätä tietueita alidomaineille, joten wikiosoitteelle saatetaan hyvinkin toteuttaa tulevaisuudessa alidoimainnimi. Lähitulevaisuudessa on näkyvissä myös projekti, jonka myö-

tä PKI-järjestelyt muuttuvat entistä monipuolisemmiksi ja helpommin laajennettaviksi. EJBCA onkin tulevaisuuden ratkaisu tilaajayrityksen PKI-järjestelmänä.

Työ on tehty yhteisymmärryksessä yrityksen vetovastuussa olevan työntekijän kanssa, jonka taholta idea opinnäytetyöhön syntyi. Olen erittäin tyytyväinen antamaani panokseen muiden työntekijöiden päivittäisten toimenpiteiden helpottamisessa. Oma käsitykseni ja osaamiseni sertifikaattijärjestelmien, Apache http -palvelimen ja Apache Tomcat -sovelluspalvelimen osalta on laajentunut työn ohessa myös huomattavasti. Kaiken kaikkiaan opinnäytetyön teko on tuonut käytännön kokemusta tietoturvallisuuteen ja tiedon turvalliseen käyttämiseen liittyvissä asioissa.

10 LÄHTEET

Alvestrand, H. 2004. Network Working Group Request for Comments: 3935. Luettu 3.5.2011. http://www.ietf.org/rfc/rfc3935.txt .

The Apache Software Foundation. [www-sivu]. Luettu 14.3.2011. http://httpd.apache.org/docs/2.2/.

The Apache Software Foundation. [www-sivu]. Luettu 14.3.2011. http://httpd.apache.org/docs/2.2/mod/mod_ssl.html.

Atlassian. [www-sivu]. Luettu 24.4.2011. http://confluence.atlassian.com/display/DOC/Confluence+Documentation+Home.

Atlassian. [www-sivu]. Luettu 16.5.2011. https://jira.atlassian.com/secure/IssueNavigator.jspa?mode=show&createNew=true .

Canonical Ltd.. [www-sivu]. Luettu 5.5.2011 http://manpages.ubuntu.com/manpages/gutsy/man1/keytool.j2se14.1.html.

EJBCA team. [www-sivu]. Luettu 18.5.2011. http://www.ejbca.org/.

EJBCA team. [www-sivu]. Luettu 18.5.2011. http://www.ejbca.org/screenshots/ejbca1.png.

Logica. [www-sivu]. Luettu 17.5.2011. https://winha.tamk.fi/winhawille/wille.asp.

Microsoft. [www-sivu]. Luettu 8.5.2011. http://support.microsoft.com/kb/257591.

Mozilla Foundation. [www-sivu]. Luettu 26.3.2011. http://www.mozilla.org/projects/security/certs/included/.

Weise, J. 2001. Public Key Infrastructure Overview. Luettu 18.5.2011. http://www.sun.com/blueprints/0801/publickey.pdf.

The OpenSSL Project. [www-sivu]. Luettu 26.4.2011. http://www.openssl.org/.

Roundcube.net. [www-sivu]. Luettu 17.5.2011.
http://www.roundcube.net/about#features.

Singh, S. 1999. Koodikirja - Salakirjoituksen historia muinaisesta Egyptistä kvantti-kryptografiaan. Helsinki: Tammi.

Viestintävirasto. [www-sivu]. Luettu 13.4.2011.
http://www.ficora.fi/index/palvelut/palvelutaiheittain/tietoturva/pki/varmenne.html.