Karri Korhonen

SAML 2.0 -tuen lisäys IMS-ohjelmistoon

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tietotekniikka

Insinöörityö

28.4.2014

Metropolia

Tekijä(t) Otsikko Sivumäärä Aika	Karri Korhonen SAML 2.0 -tuen lisäys IMS-ohjelmistoon 39 sivua 28.4.2014
Tutkinto	insinööri (AMK)
Koulutusohjelma	Tietotekniikka
Suuntautumisvaihtoehto	Ohjelmistotekniikka
Ohjaaja(t)	Simo Silander, Lehtori, Tutkintovastaava Jarkko Lakso, IMS Business Solutions Oy

Insinöörityön tavoitteena oli lisätä Javalla toteutettuun web-pohjaiseen IMS-ohjelmistoon tuki eri toimialueiden väliselle kertakirjautumiselle. Päämääränä oli mahdollistaa kertakirjautuminen IMS:n itse hallinnoimilla palvelimilla. IMS-ohjelmisto sisältää jo tuen kertakirjautumiselle, mutta vain asiakkaan toimialueen sisällä ja Windows-ympäristöissä.

Toimialueiden välisen kertakirjautumisen tekniikaksi valittiin XML-pohjainen standardi SAML. SAML 2.0 tukee IMS-ohjelmiston aloittamaa kertakirjautumisprosessia, ja useimmat käyttäjähallintoohjelmistot tukevat SAML-standardia. Itse projekti toteutettiin käyttäen OpenSAML-kirjastoa. Työn keskeisimmät toiminnallisuudet olivat SAML-todennuspyynnön lähettäminen ja todennusvastauksen käsittely IMS-ohjelmistossa.

Lopputuloksena syntyi toimiva ja SAML 2.0 -standardeja noudattava eri toimialueiden välillä toimiva kertakirjautumisratkaisu IMS-ohjelmistoon. Toiminnallisuus jäi IMS Business Solutions Oy:n käyttöön, ja se odottaa integrointia Spring Security SAML-kirjastoon. SAML-selainkertakirjautuminen on ollut jo yli vuoden käytössä useammalla asiakkaalla.

Avainsanat	kertakirjautuminen, saml, java, käyttäjähallinto

Author(s) Title Number of Pages Date	Karri Korhonen Implementing SAML 2.0 in IMS-software 39 pages 28 April 2014
Degree	Bachelor of Engineering
Degree Programme	Information technology
Specialisation option	Software engineering
Instructor(s)	Simo Silander, Principal Lecturer Jarkko Lakso, IMS Business Solutions Oy

The purpose of this final year project is to implement cross-domain single sign-on in IMS-software. IMS-software is web based and done by Java programming language. The goal is to make single sign-on possible on servers managed by IMS. IMS-software already support some single sign-on techniques but they are not cross-domain and work best within Windows-enviroments.

XML-based standard SAML was qualified as the means of implementing cross-domain single sign-on. SAML 2.0 supports service provider initiated single sign-on and most major Identity Management software support it. The programmatic side was made with OpenSAML-libraries. The most important functions of this project was to succesfully construct and send SAML authentication requests and have IMS receive and inspect SAML authentication response messages.

The outcome of this project is a functional cross-domain SSO-solution for IMS-software that abides SAML 2.0 standard. The implementation is being used at IMS Business Solutions Oy and is waiting to be integrated into Sprign Security SAML-library. SAML SSO has been in use for over a year with multiple clients.

Keywords	single sign-on, saml, java, identity management

Sisällys

Lyhenteet

SAML	Security Assertion Markup Language. XML-pohjainen avoin standardi käyttäjien todentamiseen palveluntarjoajan ja henkilöllisyydentarjoajan välillä.
XML	Extensible Markup Language. Rakenteellinen kuvauskieli, joka auttaa jäsentämään laajoja tietomassoja ja standardi.
SP	Service Provider eli palveluntarjoaja. Ohjelmisto tai järjestelmä, joka tarjoaa käyttäjille palvelua.
IdP	Identity Provider eli henkilöllisyydentarjoaja. Luotettu taho, joka antaa käyttäjälle sähköisen henkilöllisyyden palveluntarjoajaa varten.
AD	Active Directory. Microsoftin Windows-toimialueen käyttäjätietokanta ja hakemistopalvelu.
AD FS	Active Directory Federation Services. Microsoftin ohjelmisto, joka tarjoaa käyttäjille kertakirjautumistoiminnallisuuksia.
SOAP.	Simple Object Access Protocol. Tietoliikenneprotokolla rakenteellisen tiedon vaihtamiseen palveluiden välillä.

1 Johdanto

Kertakirjautuminen on melko tavallinen ominaisuus, jota vaaditaan yhä enemmän yrityskäyttöön tarkoitetuilta ohjelmistoilta. Pienet ja keskisuuret yritykset kykenevät toimimaan ilman sitä. Kertakirjautuminen on kuitenkin looginen jatke, jos yrityksellä on keskitettyä käyttäjähallintoa. Suurin osa tarjolla olevista yrityskäyttöön tarkoitetuista ratkaisuista on kuitenkin tarkoitettu toimimaan ainoastaan yrityksen omalla toimialueella.

SAML tarjoaa mahdollisuuden yrityksille laajentaa kertakirjautumistaan yrityksen sisäverkon ulkopuolelle. Toisin sanoen sovelluksen, jota työntekijä käyttää, ei tarvitse sijaita yrityksen hallinnoimilla palvelimilla, jotta kertakirjautuminen olisi mahdollista. SAML ja muut samanlaiset ratkaisut ovat elintärkeitä organisaatioille, joiden työntekijät eivät tee töitä ainoastaan yrityksen omissa tiloissa. Lisäksi se siirtää vastuuta ja työtaakkaa palvelimien ylläpidosta pois asiakkaalta ja enemmän palveluntarjoajan suuntaan.

IMS Business Solutions Oy on yritysten ja organisaatioiden toiminnanhallintaa ja -johtamista tukeviin ratkaisuihin perustuva yritys. Se sijaitsee Helsingin Pitäjänmäessä ja työllistää tällä hetkellä noin 30 henkilöä. Kirjainlyhenne IMS muodostuu sanoista Integrated Management System (suom. toimintajärjestelmä). IMS Business Solutions Oy:n päätuote on IMS-ohjelmisto ja sitä tukevat palvelut.

IMS-ohjelmisto on selainpohjainen Javalla toteutettu laadunhallinta-, toiminta- ja johtamisjärjestelmä yrityksille. Ohjelmiston keskeiset toiminnallisuudet ovat prosessien kuvaaminen, dokumenttien hallinta, palautteiden ja arvioiden käsittely, tulosten mittaaminen sekä käsikirjojen kokoaminen.

Työssä toteutetaan IMS Business Solutions Oy:lle ratkaisu ulkoverkon kertakirjautumisongelmalle. IMS-ohjelmisto mahdollistaa jo ennen työtä kertakirjautumisen, mutta tekniikkaa tukevat asennukset ovat asiakkaiden hallinnoimilla Windows-palvelimilla. Olemassa olevia ratkaisuja ei ole tarkoitettu toimimaan ulkoverkoissa, ja ne voivat vaatia useita porttiavauksia asiakkaan palomuureissa.

Työssä esitetään, kuinka SAML-kertakirjautumisen tuki lisättiin IMS-toimintajärjestelmään. Työtä alustetaan kertomalla käyttäjähallinnosta ja kertakirjautu-

misesta yleisesti ennen siirtymistä SAML-kertakirjautumisprotokollaan ja itse käytännön työhön. Toisessa luvussa esitetään syitä kirjautumisen tarpeellisuudelle. Lisäksi luvussa kerrotaan IMS-ohjelmiston kertakirjautumisratkaisuista ennen työtä ja työn tavoitteista.

Luku 3 käsittelee käyttäjähallintoa. Käyttäjähallinnolla tarkoitetaan pääsynvalvonnan hallinnollista puolta eli käyttäjien sekä heidän oikeuksiensa ylläpitoa. Luvussa avataan yleisiä käyttäjähallintoon liittyviä termejä ja kerrotaan tietyistä käyttäjähallinnon malleista. Luvussa 4 tarkastellaan kertakirjautumista. Luku esittelee yleisimpiä kertakirjautumisen protokollia. Lisäksi siinä pyritään pohtimaan kertakirjautumisen hyviä ja huonoja puolia.

Luku 5 kertoo työn kannalta oleellisimmasta kertakirjautumisprotokollasta eli SAML:sta. Siinä puhutaan SAML:n tärkeimmistä osa-alueista ja ominaisuuksista. Lisäksi luvussa avataan SAML-protokollan taustoja ja syytä sen olemassaololle. Luvussa 6 selostetaan työn alkuvaiheista ja miksi OpenSAML-kirjasto oli lopullinen päätös toteuttaa SAML-kertakirjautuminen IMS-ohjelmistossa. Luvussa kerrataan myös kirjautumisprosessin eteneminen sekä esitellään SAML-viestien lähettämistä ja vastaanottamista.

2 Lähtökohdat ja tavoitteet

Kirjautuminen järjestelmään on kuin oven avaamista. Jotta oven saa auki, henkilö tarvitsee avaimen jokaiseen lukkoon. Vaikka avain olisi kirjaimia, numeroita tai perinteisempi fyysinen esine, sisäänpääsy on avaimen omistajan oikeus eikä ulkopuolisella ole asiaa sisälle ilman, että toinen avaimen omistaja antaa luvan eli käytännössä lainaa omaa avaintansa. Tämäkään ei aina riitä. Miksi vieraat sitten täytyy pitää ovien ulkopuolella? Miksei kaikki ole aina avointa ja helposti saatavilla?

2.1 Pääsynvalvonta

Päiväkirjassa on lukko, jotta muut kuin kirjoittaja eivät voi lukea sitä. Kassakaapissa on lukitsemismekanismi, ettei sen sisällä olevaa rahaa tai arvopapereita voisi varastaa. Ovessa on lukko, että talon omistaja voi estää asiattomien ihmisten sisäänpääsyn. Lu-

kolla on käytännössä kaksi tilaa: auki ja kiinni. Poissaollessaankin sen tila voidaan määrittää olevan auki.

Lukko suojelee yksityisyyttä ja turvaa omaisuutta. Jos kaikki olisi avointa ja vapaasti otettavissa, ”omaisuus” olisi sanana turha, mutta niin olisivat lukotkin. Täysin avoimessa ja vapaassa yhteiskunnassa ei voi omistaa mitään, sillä kaikki on toisen otettavissa. Vaikka ihmiskunta ei näin vapaalla pohjalla toimikaan, niitäkin löytyy, jotka kokevat omaisuuden rajat häilyviksi. Miksi omaisuuden hallussapitäminen silti tarvitsee ylimääräistä suojaa ja miksi lukoissa on nimenomaan kyse turvasta? Emmekö voisi vain sopia yksityisyyden rajoista ja omistussuhteista?

Lukitun oven ja kirjautumisikkunan tavoite on estää eritoten pahoissa aikeissa liikkuvien pääsy sisään aivan kuten keskiaikaisen linnan portti piti viholliset ulkona. Ilman lukkoa kuka tahansa voisi teoriassa kävellä ovesta sisään. Miksei se joku voisi myös olla rikollinen? Talosta rikollinen saa esimerkiksi arvotavaroita ja sovelluksesta kaikkea yrityssalaisuuksien ja sijaintitietojen väliltä toisin sanoen rikollista hyötyä.

Aina yhden tason suojaus ei riitä. Konkreettisena esimerkkinä kerrostalon asukas pääsee samalla avaimella rakennukseen sekä omaan asuntoonsa. Ulkopuolisia estetään pääsemästä rakennukseen ylipäänsä ja asukkaita vielä erikseen muiden asuntoihin. Sama pätee myös yrityksen sisäiseen lähiverkkoon eli intranetiin. Intranetiin ei pääse yrityksen ulkopuolisesta verkosta eikä näin myöskään intranetissa toimiviin sovelluksiin. Yrityksen työntekijä voi käyttää lähiverkossa niitä sovelluksia, joihin hänellä on avain eli salasana.

Turvallisuuden tunne ja tieto ovat tärkeitä niin kodissa kuin sovelluksessakin. Miksi sovellusta sitten tarvitsee käyttää verkon yli? Eikö internetistä eristyksissä oleva kone olisi kaikkein turvallisin ratkaisu? Korvessa asuva erakko kaipaa kuitenkin joskus seuraa, ja myös järjestelmät ovat usein sosiaalisia. Kun yksityishenkilöt jakavat tuntemuksiaan sosiaalisessa mediassa, suuryrityksissä tulevaisuutta katsotaan laadunhallintajärjestelmien tarjoamien talouslukemien pohjalta. Ilman tietojen jakamista yritys ei voi toimia ja sähköiset järjestelmät jakavat tietoa kaikkein tehokkaimmin. Sähköinen järjestelmä taas toimii parhaiten verkon välityksellä.

2.2 IMS-ohjelmiston kirjautumisen lähtötilanne

Kirjautuessaan käyttäjä antaa sovellukselle avaimensa. Avain voi avata kuitenkin samalla monia ovia. Tätä kutsutaan kertakirjautumiseksi (engl. SSO, Single Sign-on). Käyttäjältä ei pyydetä uudelleen todentamista vaan kertakirjautumisen piirissä toimivat sovellukset ja palvelut keskustelevat taustalla keskenään. Jos kaikki sujuu kuten pitäisi, käyttäjä ei huomaa osapuolten välistä kanssakäyntiä.

IMS-ohjelmisto sisältää JESPA- ja JCIFS-kertakirjautumisratkaisut perinteisemmän järjestelmäkohtaisen kirjautumisen lisäksi. Kumpikin pohjautuu NTLM-autentikointiprotokollaan, mutta vain kaupallinen JESPA tukee tuoreempaa ja turvallisempaa NTLMv2-versiota. NTLM on osa sisäänrakennettua Windows-autentikointia, josta kerrotaan lisää luvussa 4.3.4. IMS-ohjelmistossa on myös mahdollista käyttää LDAPv3-kirjautumista, jossa käyttäjä todentaa itsensä IMS-ohjelmistossa kirjautumalla suoraan AD-palvelimelle salatun yhteyden yli.

Jotta kertakirjautuminen voisi toimia luotettavasti, IMS-ohjelmistolla täytyy olla ajankohtaista tietoa asiakkaan käyttäjätiedoista. IMS synkronoi käyttäjänsä asiakkaan AD-palvelimelta. Käyttäjiä ei voi lisätä IMS-ohjelmistoon käsin, mikäli kertakirjautuminen on käytössä. Käyttäjiä ei myöskään voi synkronoida AD-palvelimelta ilman kertakirjautumista.

Sekä NTLM-protokolla että käyttäjäsynkronointi asettaa haasteita IMS:n itse hallinnoimilla palvelimilla. Asiakkaan täytyy avata palomuuristaan portteja todennuskutsuille ja käyttäjäsynkronoinnille, mikäli asennusta ei tehdä asiakkaan omalle toimialueelle. Lisäksi käyttäjätiedot altistuvat ulkopuolisille urkkijoille, mikäli tiedonsiirtoa ei suojata.

2.3 Tavoite

Työn tavoitteena on ottaa IMS-ohjelmistossa käyttöön eri toimialueiden väliseen kertakirjautumiseen tarkoitettu SAML-kertakirjautuminen, tarkemmin ottaen SAML 2.0 -version, joka mahdollistaa käyttäjän todentamiseen tähtäävän SAML-prosessin alkamisen IMS-ohjelmistosta. Tällöin IMS-asennus voisi sijaita IMS:n itse hallinnoimalla pal-

velimella. Aiemmat kertakirjautumismenetelmät sallivat vain asiakkaan omalla toimialueella eli omilla palvelimilla toimivien asennusten käyttämisen.

IMS-asennukset sijaitsevat normaalisti yrityksen itse hallinnoimilla palvelimilla. Palvelimilla käytetään Linux-pohjaista käyttöjärjestelmää. IMS-ohjelmiston tukemat NTLMv1- että NTLMv2-autentikointiprotokollat tukevat parhaiten kertakirjautumista Windows-ympäristöissä. Osittain nykyisten ratkaisujen Windows-painotteisuuden takia IMS-ohjelmisto tarvitsi uuden kertakirjautumisratkaisun.

IMS-ohjemistossa on toimiva toteutus asiakkaan käyttäjien synkronointiin AD-palvelimelta. Aiemmin synkronointia ei tarvinnut salata, sillä tieto liikkui vain asiakkaan omalla toimialueella. Valmiissa työssä käyttäjien tiedot kuitenkin siirretään verkon yli, joten synkronoinnissa liikkuvan tiedon täytyy olla salattua.

Lähteviltä ja saapuvilta SAML-viesteiltä vaaditaan, että niihin lisätään digitaalinen allekirjoitus. Toisin sanoen IMS-ohjelmiston tulee osata todentaa ja allekirjoittaa SAML-viestejä. Yhdessä salatun yhteyden yli tehtävällä käyttäjäsynkronoilla IMS-ohjelmistolla on tarve keskitetylle sertifikaattien tallennussijainnille.

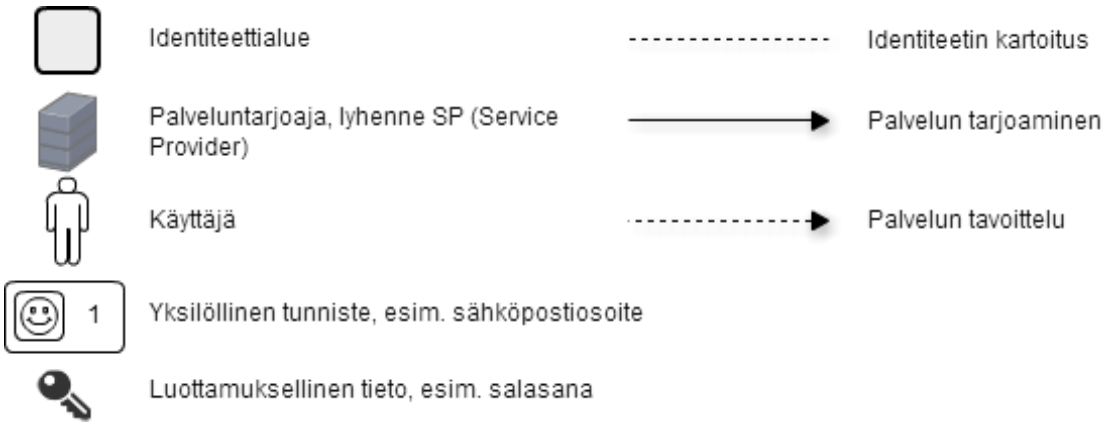
3 Käyttäjähallinto

Käyttäjähallinto (engl. Identity Management, IdM) on käyttäjätietojen sähköistä hallinnointia. Käyttäjätiedoilla viitataan tietoihin, jotka todentavat käyttäjän henkilöllisyyden ja jotka kertovat, mitä resursseja ja toimintoja käyttäjällä on valtuus hyödyntää. Se sisältää myös käyttäjän toimia kuvaavia tietoja. Hallinnoitavia olioita ovat käyttäjät, laitteisto, verkon resurssit ja ohjelmistot. [6.]

3.1 Käyttäjähallinnon peruskäsitteitä

Sähköisellä henkilöllisyydellä tarkoitetaan henkilön, ryhmän, asian tai käsitteen läsnäoloa verkossa. Se kattaa kaikki ne attribuutit, jotka yksilöivät kohteensa. Yhteystietojen lisäksi tällaisia attribuutteja voivat työelämässä olla esimerkiksi työntekijätunnus ja osasto, mutta myös listaus pääsyoikeuksista työpaikan sovelluksiin.

Henkilöllisyyden todentaminen eli autentikointi on prosessi, jossa toimija esittää tarvittavat perusteet omasta henkilöllisyydestään. Perusteena käytetään jotain, mitä vain toimija tietää, omistaa tai vain toimija voi olla. Salasana on jotain, mitä vain toimija tietää. Sormenjälki tai silmän verkkokalvo ovat tunnuksia, jotka ovat konkreettisesti osa ihmistä eli ovat toimija. Sertifikaatti tai luottokortti ovat asioita, jotka käyttäjällä ovat hallussa. Todennuksen jälkeen resursseja valvova taho valtuuttaa eli auktorisoi toimijan pääsemään käsiksi sisältöihin ja toiminnallisuuksiin. [5; 6.]

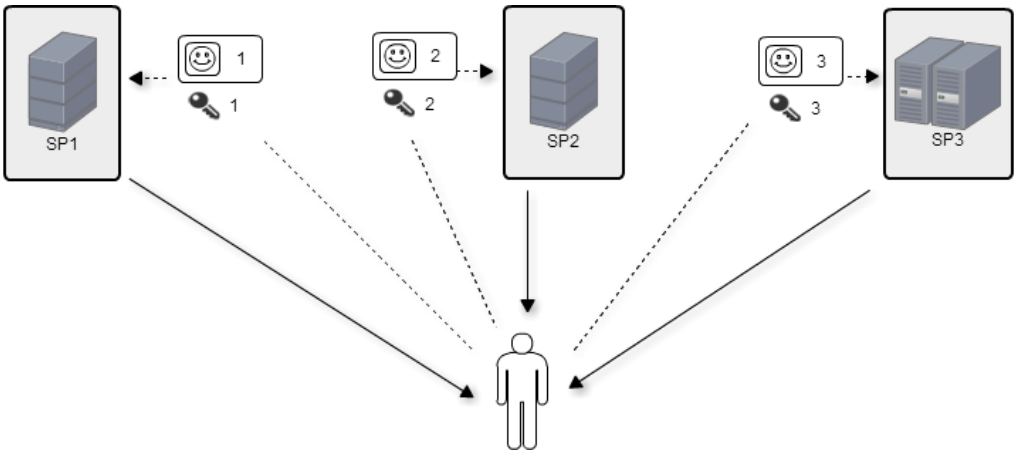


Kuva 1. Kuvissä käytettävien symbolien selitykset [4.]

Kappaleissa 3.2., 3.3. ja 3.4. on erilaisten käyttäjähallinnon mallien kuvia. Kuvissa käytettävät symbolit näkyvät kuvassa 1. Identiteettialue tai -toimialue rajaa paikan, jossa käyttäjällä on toiminnassaan yksi tietty henkilöllisyys. Palveluntarjoaja on järjestelmä, jonka sisältöjä käyttäjä haluaa nähdä.

3.2 Järjestelmäkohtainen käyttäjähallinto

Perinteisin käyttäjähallinnon malli on tilanne, jossa palvelu vastaa sekä käyttäjien tunnistamisesta että tunnistetietojen tarjoamisesta. Palveluntarjoaja hallitsee tietyn palvelun nimiavaruutta ja yhdistää yksilöllisen tunnisteen käyttäjään. Käyttäjällä on jokaisessa palvelussa oma yksilöllinen tunnisteensa ja sen lisäksi tähän liitettävä lisätieto kuten salasana. Jos käyttäjän tunnuksia täytyy muokata, tämä tehdään jokaisessa palvelussa tai henkilöllisyydentarjoajassa erikseen. Tällaista mallia kutsutaan eristetyksi tai järjestelmäkohtaiseksi käyttäjähallinnoksi. [4.]



Kuva 2. Järjestelmäkohtainen käyttäjähallinto [4.]

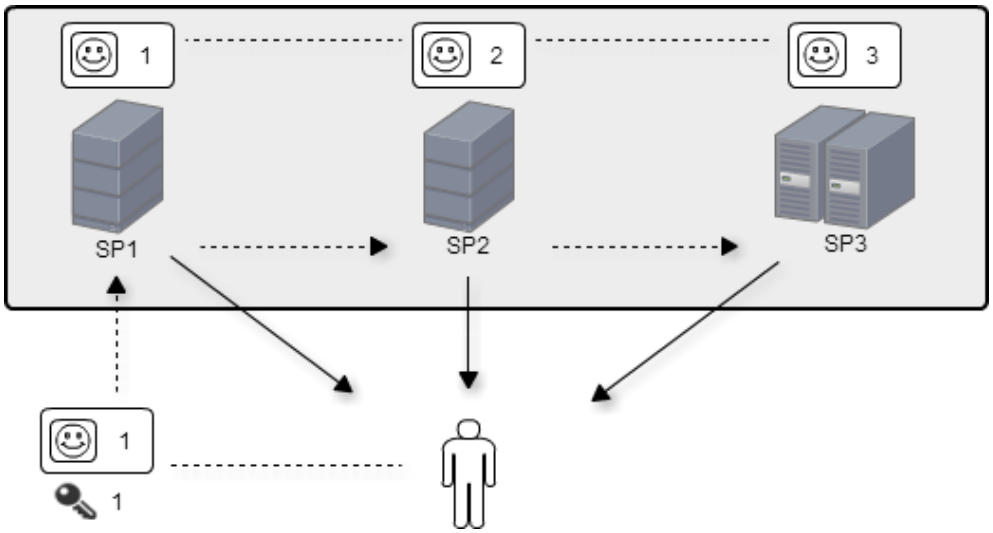
Eristetty käyttäjähallinto on palveluntarjoajalle helpoin toteuttaa. Ylläpidon näkökulmasta tämä järjestelmäkohtainen käyttäjähallinto muuttuu kuitenkin nopeasti vaivalloiseksi ja suuremmilla käyttäjämäärillä mahdottomaksi hallita tehokkaasti. Kun sovelluksiakin on useita, käyttäjien pääsynvalvonta on vähintäänkin haastavaa. [4.]

Tietoturvan näkökulmasta olisi suotavaa, että käyttäjällä on eri salasana-tunnuspari jokaisessa eri palvelussa. Nykyään lähes kaikki palvelut vaativat kirjautumista, joten turvallista linjaa noudattava henkilö joutuu kehittämään useita eri tunnuksia ja salasanoja useisiin palveluihin. [4.]

Useita eri tunnuksia käyttävä unohtaa pakostakin salasanojaan, joten tunnuksensa saattaa kirjoittaa muistiin. Tällöin altistuu salasanojen perinteisemmälle urkinnalle eli olan yli lukemiselle. Käyttäjä voi myös aktiivisesti pyydellä uusia salasanoja palveluntarjoajilta, mikä kuormittaa ylläpidon taakkaa. Lisäksi käyttäjä kuluttaa ylimääräistä aikaa jokaisen uuden salasanatilauksen kohdalla. [4.]

3.3 Yhteinen käyttäjähallinto

Federoidun eli yhteisen käyttäjähallinnon malli pyrkii purkamaan järjestelmäkohtaisen käyttäjähallinnon kasaamaa työkuormaa. Yhteisellä identiteetillä tarkoitetaan eri palveluntarjoajien välillä vallitsevaa luottamusverkkoa. Verkkoon kuuluvat toimitsijat sitoutuvat noudattamaan samoja pelisääntöjä käyttäjien tunnistamisessa sekä valtuuttamisessa yhteisellä toimialueella. [4.]



Kuva 3. Yhteinen käyttäjähallinto [4.]

Yhteisen identiteetin toimialueella palveluntarjoajat sopivat keskenään, miten käyttäjän palvelukohtainen henkilöllisyys tunnistetaan jokaisen palvelun omalla identiteettitoimialueella. Sopimuksessa määritellään toimintaperiaatteet ja käytettävät teknologiat. Käyttäjän palvelukohtaiset tunnisteet kartoitetaan, jolloin lopputulokseksi saadaan virtuaalinen identiteettitoimialue (engl. Virtual Identity Domain) yhdelle käyttäjälle. [4.]

Käyttäjä on käytännössä kirjautunut kaikkiin saman federoidun käyttäjähallinnon piirissä toimiviin palveluihin, kun hän on onnistuneesti kirjautunut niistä yhteen. Tämä tapahtuu palveluiden välillä toimivilla vakuutuksilla. Vakuutus sisältää enintään käyttäjän tunnuksen. Käyttäjän pääsy muihin palveluihin perustuu palveluntarjoajien väliseen luottamukseen. [4.]

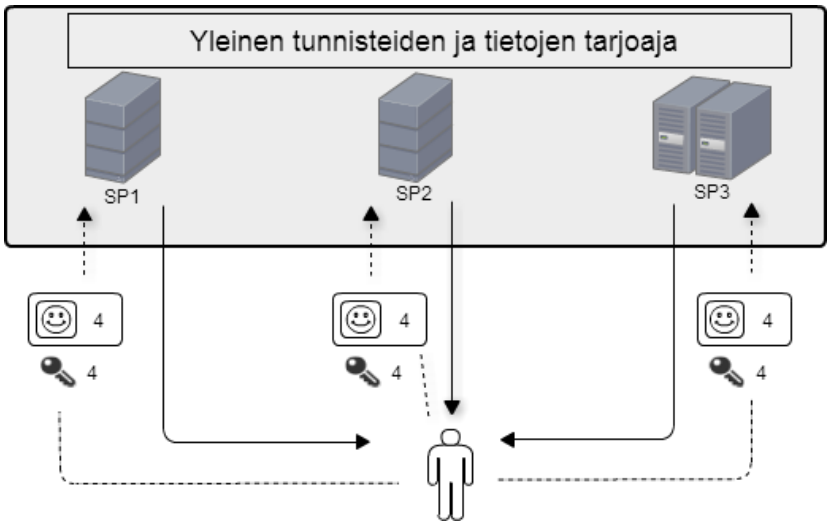
Federoidun identiteetin toimialue on käytännössä järjestelmäkohtaisten identiteettitoimialueiden liittouma. Käyttäjälle voi jäädä sellainen vaikutelma, että hänellä olisi käytössään vain yhdet tunnukset. Todellisuudessa käyttäjä voi silti päätyä hallitsemaan useita eri tunnuksia, vaikka käyttäisikin vain yhtä aktiivisesti. Federoitu identiteetti toimii parhaiten, kun käyttäjällä on yksi ja sama tunnus joka palveluun. [4.]

3.4 Keskitetty käyttäjähallinto

Keskitetyssä käyttäjähallintomallissa on yksi tunnuksien ja tunnisteiden tarjoaja. Kaikki palveluntarjoajat käyttävät tätä yhtä henkilöllisyydentarjoajaa joko yksinomaan tai yhdessä muiden samanlaisten tarjoajien kanssa. Keskitetystä käyttäjähallinnosta on olemassa useita eri vaihtoehtoja, joista tässä työssä selostan lyhyesti yhteisestä, käsitteellisestä sekä kertakirjautumisen käyttäjähallinnosta.

3.4.1 Yleinen käyttäjähallinto

Yleisessä käyttäjähallinnossa on ainoastaan yksi taho, joka hallinnoi käyttäjien tunnisteita sekä tietoja ja jota kaikki palveluntarjoajat käyttävät. Tällöin käyttäjällä on käytössään vain yhdet tunnukset, joilla hän pääsee kaikkien palveluntarjoajien sisältöihin käsiksi. Yleinen käyttäjähallinto toimii parhaiten yksittäisessä organisaatiossa, jossa tietyn käyttäjäryhmän tulee päästä käsiksi samoihin sovelluksiin. Suuremmassa mittakaavassa yhden henkilöllisyydentarjoajan ylläpito on haastavaa.

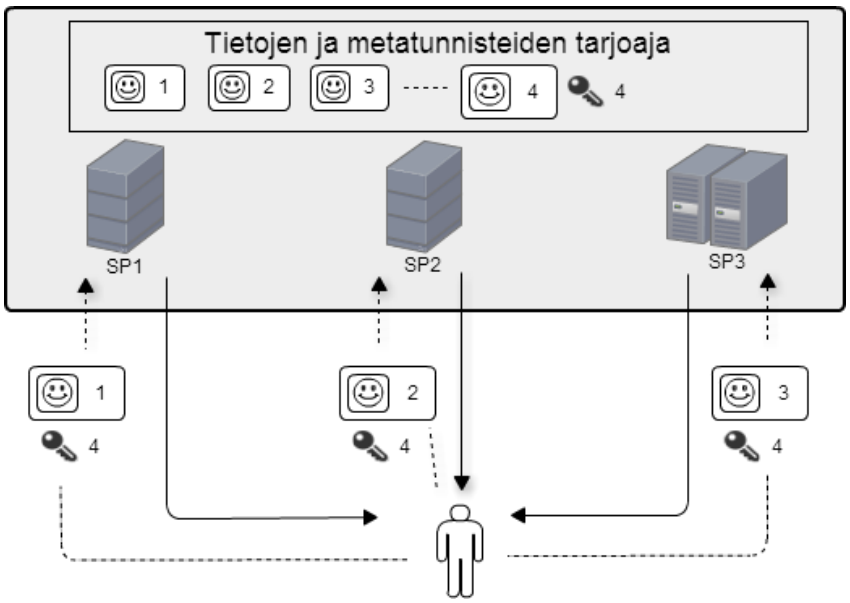


Kuva 4. Yleinen käyttäjähallinto [4.]

3.4.2 Käsitteellinen käyttäjähallinto

Käsitteellinen käyttäjähallinto on malli, jossa palveluntarjoajat jakavat keskenään käyttäjästä tiettyjä henkilöllisyyteen liittyviä tietoja. Näistä tiedoista muodostetaan yksilölli-

nen käsitteellinen tunniste eli metatunniste, johon liitetään myös salasana tai muu lisätieto. Metatunnisteen toteutus tapahtuu yleensä käyttämällä niin sanottua metahakemistoa kuten LDAP:ta (engl. Lightweight Directory Access Protocol), jolloin kaikki käsitteellisen käyttäjähallinnon toimialueella sijaitsevat palvelut toimivat yhden auktoriteetin alaisuudessa. [4.]

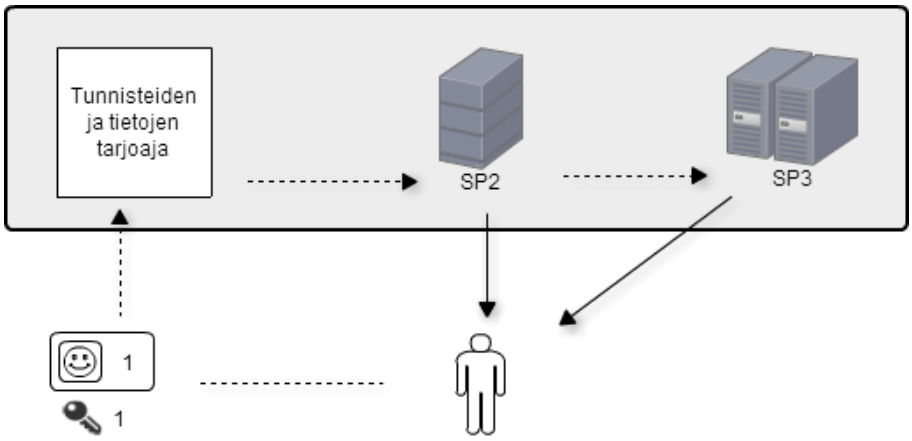


Kuva 5. Käsitteellinen käyttäjähallinto [4.]

Normaalisti metatunniste on käyttäjältä piilossa ja palveluntarjoajat käyttävät sitä vain sisäisesti. Käyttäjälle tämä näkyy salasanojen synkronoinnissa useiden palveluiden välillä. Toisin sanoen, kun käyttäjä vaihtaa salasanansa yhdessä palvelussa, se päivittyy myös kaikissa muissakin palveluissa. [4.]

3.4.3 Kertakirjautumisen identiteettitoimialue

Kertakirjautuminen on seuraava askel yleiselle ja käsitteelliselle käyttäjähallinnolle. Kun käyttäjä on todennettu yhdelle palveluntarjoajalle, hänet on todennettu muillekin palveluntarjoajille. Normaalisti yksi osapuoli vastaa käyttäjähallinnosta eli on henkilöllisyydentarjoaja. Kertakirjautuminen on hyvin samanlainen yhteisen käyttäjähallinnon kanssa. Oleellinen ero on, että käyttäjällä on yksi ainoa tunnus, jonka kukin palveluntarjoaja hyväksyy. Näin ollen erillistä tunnusten kartoittamista ei tarvita. [4.]



Kuva 6. Kertakirjautumisen identiteettitoimialue[4.]

4 Kertakirjautuminen

Kertakirjautumisella (engl. Single Sign-on, SSO) tarkoitetaan menetelmää, jossa käyttäjä pääsee yhdellä kirjautumisella käsiksi useaan eri palveluun. Sen parhaita puolia ovat työskentelyn nopeutuminen eri palveluiden välillä. Käyttäjien taakka vähenee, kun tunnuksia ei tarvitse syöttää tai muistaa yhtä enempää. Kertakirjautumisen vastapainona on myös uloskirjautuminen (engl. Single Sign-off), jolloin käyttäjä kirjataan kerralla ulos kaikista palveluista, joissa kirjautuminen on voimassa. [1.]

4.1 Etuja

Kertakirjautumisen suurimpana etuna on, ettei käyttäjän tarvitse hallinnoida useita eri tunnuksia ja salasanoja, kun hän käyttää eri Internet-palveluita työpaikalla tai vapaa-ajalla. Tietysti käyttäjä voisi tukeutua samoihin tunnuksiin ja salasanoihin jokaisessa palvelussa, mutta tällöin salasanojen vaihtaminen tasaisin väliajoin tarkoittaa ylimääräistä työtä puhumattakaan turvallisuudesta.

Tietoturvaa voi parantaa käyttämällä vahvoja salasanoja. Vahvalla salasanalla on monta määritelmää, mutta lyhyesti sanottuna sillä tarkoitetaan merkkijonoa, joka on yli kahdeksan merkkiä pitkä, sisältää sekä pieniä että isoja kirjaimia, numeroita ja erikoismerkkejä. Useiden vahvojen salasanojen muistaminen on normaalille ihmiselle mahdotonta. Salasanoja voi säilyttää esimerkiksi erillisessä tekstitiedostossa tai paperilla,

mutta tällä tavoin salasanat altistuvat haittaohjelmille tai olan yli lukijoille. Joka tapauksessa salasanoja pitäisi tasaisin väliajoin päivittää ja muistiinpanoja ylläpitää, työ ei missään vaiheessa vähene. Helppous ja hyvä tietoturva sulkevatkin usein toisensa pois.

Kertakirjautuminen keskittää käyttäjän tunnusten hallintaa. Yrityksillä ja organisaatioilla tämä tarkoittaa henkilökunnan sähköisten henkilöllisyyksien keskittämistä. Kun salasanojen jakamista ja käyttöoikeuksien hallinnointia hoidetaan yhdestä kohteesta, ylläpitäjien taakka pienenee huomattavasti. Ilman kertakirjautumista on ohjelmistopalveluja ostavan yrityksen IT-osasto saattanut hoitaa kaikki käyttäjien kirjautumisongelmat ja salasanojen uusimiset. Vähänkin suuremmassa yrityksessä kertakirjautuminen säästää huomattavasti ylläpitäjien aikaa.

Kertakirjautuminen ei vähennä ainoastaan asiakasyritysten työn määrää vaan myös palveluntarjoajan ylläpidon. Kun vastuu käyttäjähallinnosta siirtyy suurimmaksi osaksi asiakkaan ylläpidolle, palveluntarjoajan vastuulle jää lähinnä sekä palvelun että kirjautumisen toimiminen. Palveluntarjoajalla tulee kuitenkin olla ajan tasalla oleva tieto käyttäjistä, jotta esimerkiksi auditointi sekä muu sisällön luonnin jäljittäminen olisi mahdollista.

4.2 Haittoja

Kertakirjautumisen käyttöönotto yritysmaailmassa on usein kallista. Sen kustannukset ovat hyvin vahvasti alkupainotteisia, ja pitkällä tähtäimellä yhtenäinen kertakirjautumisratkaisu säästää enemmän kuin usean järjestelmäkohtaisen käyttäjähallinnon ylläpito rinnakkain. Alkuvaiheessa aikaa ja rahaa kuluu henkilöllisyydentarjoajan palvelun käyttöönotossa ja työntekijöiden kouluttamisessa.

Mikäli yritys ei ole noudattanut yhtenäistä linjaa tunnisteiden kanssa, on ylläpitäjillä edes puuduttavaa työtä. Jo käytössä olevien ohjelmistojen käyttäjätiedot täytyy kartoittaa vastaamaan uutta käyttäjärekisteriä, jolloin sekä palveluntarjoajien että asiakkaiden omat ylläpitäjät kuormittuvat huomattavasti. Tällainen työ harvemmin myöskään on ilmaista.

Kertakirjautuminen ei takaa tietoturvaa. Yksityinen henkilö saattaa käyttää esimerkiksi Facebook-yhteisöpalvelun tarjoamaa tunnistautumista useassa paikassa. Joku muu voi yhdellä salasanalla päästä kaikkiin muihinkin yhteisöpalvelun kautta käytettäviin palveluihin. Tätä silmällä pitäen monet yhteisöpalvelut tarjoavat kaksivaiheista kirjautumista. Ensimmäisessä vaiheessa käyttäjä tunnistautuu normaalilla tunnuksella ja salasanallaan. Toisessa vaiheessa, jos käyttäjän selain ei ole tunnistettavissa, käyttäjä saa esimerkiksi tekstiviestinä ilmoituksen tuntemattomasta kirjautumissijainnista.

4.3 Kertakirjautumisen protokollista

Kertakirjautumista varten on kehitetty useita eri tekniikoita. Tässä luvussa kerron lyhyesti Kerberos-todennusprotokollasta, OpenID:stä, Integrated Windows Authentication -todennusprotokollasta ja Oauth:sta. Lisäksi kuvailen lyhyesti, miten kirjautuminen kussakin protokollassa etenee. SAML-kertakirjautumistekniikasta on kerrottu tarkemmin luvussa 6.

4.3.1 Kerberos

Kerberos on verkon todennusprotokolla. Sen neljäs versio julkaistiin 1980-luvun loppupuolella, kun aiemmat versiot olivat vain Yhdysvaltojen MIT:n sisäisessä käytössä. Yksinkertaisen salasanatodennuksen sijaan Kerberos-todennuksessa käytetään symmetristä salausmenetelmää. Kolmas luotettu osapuoli toimii avaintenjakokeskuksena (engl. Key Distribution Center) ja auttaa käyttäjien todentamisessa eri palveluihin. Kun käyttäjä autentikoi itsensä avaintenjakokeskukselle, keskus lähettää tunnistekokoelman eli lipun, joka on sidoksissa käyttäjän tietokoneen sessioon. Kerberos-todennusta voi käyttää vain, jos avaintenjakokeskuksena toimiva palvelu on toiminnassa. [8.]

Kerberos-todennukseen pystyvät palvelut etsivät käyttäjän koneelta lippua sen sijaan, että käyttäjän tarvitsee käyttää salasanaansa. Tietokoneita, jotka ovat yhden tai useamman avaintenjakopalvelun vaikutuksen alaisuudessa, kutsutaan Kerberos-alueeksi. Kun Kerberos-alueella toimiva käyttäjä kirjautuu työasemalleen, hänen yksilöllinen henkilöllisyytensä (engl. Principal) lähetetään avaintenjakokeskukseen lippua varten. Keskus tarkastaa henkilöllisyyden tietokannastaan. Jos tarkistus onnistuu, keskus luo

lipunmyöntölipun (engl. Ticket-Granting Ticket), salaa sen käyttäjän avaimella ja lähettää lipun käyttäjälle. [8.]

Kerberoksen toiminnan käyttäjän työasemalla mahdollistava ohjelma purkaa lipun salauksen vain kyseisellä koneella olevalla käyttäjän avaimella. Avain lasketaan salasanasta eikä sitä lähetetä verkon yli. Lippu tallennetaan välimuistiin, johon Kerberos-todennusta käyttävät palvelut pääsevät käsiksi. Keytab-tiedosto on salaamaton listaus tunnetuista Kerberos-henkilöllisyyksistä ja henkilöllisyyksiin liitetyistä avaimista. Palvelimet tarkistavat käyttäjän tiedot tiedostosta ensimmäisen todennuksen jälkeen. [8.]

Lipunmyöntölippu vanhenee yleensä vuorokauden jälkeen. Vanhenemisajan myötä lipusta on hyötyä väärinkäyttäjälle vain lyhyeksi aikaa. Kun lippu on myönnetty, käyttäjä syöttää salasanansa seuraavan kerran lipun vanhenemisen jälkeen tai jos hän on kirjautunut ulos. Kerberos tarvitsee toimivan kellonaikojen synkronoinnin sekä DNS-palvelun, jotta se voi tehokkaasti myöntää sekä vanhentaa lippuja. [8.]

4.3.2 OpenID

OpenID on avoin, hajautettu ja ilmainen ohjelmistokehys sähköiselle henkilöllisyydelle, ja se julkaistiin 2005. Käyttäjä todennetaan käyttäen normaaleja HTTP- ja HTTPS-kutsuja, joten erillisiä ohjelmia ei tarvita selaimen lisäksi. OpenID-henkilöllisyydentarjoajaksi voi ryhtyä mikä tahansa käyttäjäprofiileja ylläpitävä sivusto. Esimerkiksi Google- ja Steam-käyttäjätunnukset ovat tarvittaessa OpenID-tunnuksia. [9.]

Kun käyttäjä pääsee käsiksi sivustoon, joka sallii OpenID-tunnuksen käytön, hän yleisimmin syöttää OpenID-tunnuksensa kirjautumisikkunaan tai valitsee OpenID-tarjoajan, jota hän haluaa käyttää todentamiseen. OpenID-tunnus olkoon esimerkeissä kuvitteellinen karri.metropolia.fi. Palveluntarjoaja kanonisoi saadun tunnuksen, jolloin se olisi http://karri.metropolia.fi, ja lähettää URI-osoitteeseen XRDS-dokumentin OpenID:tä ja Oauthia varten kehitettyä Yadis-protokollaa. Käytännössä palveluntarjoaja lähettää OpenID-tarjoajalle XML-viestin, jossa pyydetään OpenID-tarjoajan osoitetta käyttäjän todennusta varten. [9.]

OpenID-tarjoaja ja palveluntarjoaja muodostavat liitoksen (engl. Association) eli yhteisen salaisuuden Diffie-Hellmannin-avaimenvaihtoprotokollalla. Muodostamisen jälkeen OpenID-tarjoaja allekirjoittaa ja palveluntarjoaja varmentaa liitoksella kaikki keskinäiset viestit. Liitos ei ole välttämätön, mutta se poistaa ylimääräisen askeleen prosessin lopusta. [9.]

Palveluntarjoaja ohjaa käyttäjän uudelleen OpenID-tarjoajan sivuille ja lähettää ohessa todennuspyyntöviestin. OpenID-tarjoaja todentaa viestin ja käyttäjän joko voimassa olevaa sessiota vasten tai pyytää syöttämään tunnukset uudelleen. Onnistunut todennus palauttaa käyttäjän takaisin palveluntarjoajalle todennusvastauksen kanssa. Mikäli järjestelmien välistä liitosta ei luotu, palveluntarjoaja pyytää vielä OpenID-tarjoajalta vastausviestin allekirjoituksen varmennusta. [9.]

4.3.3 OAuth 2.0

Lokakuussa 2012 viisivuotias OAuth sai seuraajansa OAuth 2.0 eli OAuth2. OpenID:n tavoin se on ilmainen ja hajautettu todentamisen ohjelmistokehys. OAuth keskittyy enemmän sovellusten väliseen valtuuttamiseen, mutta se soveltuu myös käyttäjän todentamiseen. Nuoresta iästään huolimatta se on onnistunut saavuttamaan merkittävän jalansijan kertakirjautumisen maailmassa. Esimerkiksi Facebook ja Google suosittelevat käyttämään OAuth2:a tai OAuth:n ja OpenID:n välistä hybridiä.

Perinteisessä toimeksiantajan ja palvelun välisessä vuorovaikutuksessa toimeksiantaja eli käyttäjän puolesta toimiva sovellus (esimerkiksi selain) pyytää lupaa päästä suojattuun resurssiin käyttäjän tunnuksilla. Jotta kolmannen osapuolen sovellus pääsisi käsiksi suojattuun resurssiin, käyttäjän on jaettava tunnistetietonsa kolmannen osapuolen kanssa. Jakamisessa on omat ongelmansa:

- Kolmannen osapuolen on säilytettävä käyttäjän tunnistetietoja palvelussaan.
- Palvelimilta vaaditaan tukea salasana-kirjautumiselle, vaikka salasanat voivat tunnetusti heikentää tietoturvaa.
- Kolmannen osapuolen sovelluksilla voi olla liian laajat oikeudet käyttäjän suojattuihin resursseihin ilman, että käyttäjä voi itse rajata sisältöä tai sisällön käyttöaikaa.

- Käyttäjä ei voi perua yhden tietyn palvelun pääsyä suojattuun sisältöön ilman, että estää samalla kaikkien kolmansien osapuolten pääsyn.
- Jos kolmannen osapuolen tietoturva pettää, kaikki sen säilyttämät tunnistetiedot ovat vaarassa. [11.]

OAuth:ssa toimeksiantaja ei käytä käyttäjän tunnuksia päästäkseen suojattuun sisältöön. Sen sijaan toimeksiantaja hankkii sisäänpääsylipun, jonka avulla se saa oikeuden käsitellä suojeltuja resursseja. Lippu on merkkijono, joka sisältää erilaisia sisäänpääsytietoja kuten voimassaoloajan. Valtuutuspalvelin myöntää lippuja kolmannen osapuolen sovelluksille, jos käyttäjä puoltaa myöntämistä. [11.]

Käyttäjä voi antaa esimerkiksi Instagramin kaltaiselle kuvapalvelulle oikeudet päästä toisen sosiaalisen median palvelun (Facebook, Google+) kuviin käsiksi. Riittää, että Instagram-palvelulle antaa valtuutukset käyttäjän kuviin eikä tunnuksia tarvitse tallentaa muualle. Tunnusten tallentamisen sijaan käyttäjä todentaa itsensä suoraan esimerkiksi Facebook-palveluun, joka myöntää sisäänpääsylipun Instagramille aiemmin ilmoitetuin valtuutuksin. [11.]

4.3.4 Sisäänrakennettu Windows-autentikointi

Sisäänrakennettu Windows-autentikointi tai IWA (engl. Integrated Windows Authentication) käyttää Windows-käyttöjärjestelmän asiakasohjelmia ja palvelimia. Aiemmin se on tunnettu muun muassa nimellä NTLM-autentikointi. Sisäänrakennettu Windows-autentikointi käyttää todentamiseen erilaisia autentikointiprotokollia kuten NTLM:a ja Kerberosta. [15; 16.]

Sisäänrakennetun Windows-autentikoinnin hyviä puolia ovat mahdollisuus ohittaa käsin kirjautuminen ja salasanattoman kirjautumisen mahdollisuus. Salasanattomia autentikointitapoja ovat esimerkiksi biometrinen tunniste kuten sormenjälki tai fyysinen esine kuten kulkukortti. Yhtenä huonona puolena on, että asiakasohjelmien ja palveluiden tulee kuulua samaan Windows-toimialueiden verkkoon. [16.]

Sisäänrakennetussa Windows-autentikoinnissa toimeksiantaja pyytää Windows-järjestelmältä lippua eli merkkijonoa, joka kuvastaa toimeksiantajan tietokoneelle kirjautunutta käyttäjää. Pyyntö tehdään, kun kirjautumisen piirissä olevaan palveluun yritetään päästä käsiksi. Kun Windows-järjestelmä palauttaa merkkijonon, toimeksiantaja

lähettää sen tavoittelemaansa palveluun. Vain merkkijono lähetetään eikä palveluntarjoaja saa käyttäjän salasanaa. [16.]

Palveluntarjoaja palauttaa merkkijonon Windows-järjestelmälle tarkastusta varten. Windows kertoo palveluntarjoajalle, että merkkijono pätevä, jos tarkastus läpäistään. Kun palveluntarjoaja hyväksyy yhteydenoton, se saa tietoonsa käyttäjän tunnuksen. Tunnuksessa täytyy olla mukana tieto toimialueesta. [16.]

5 SAML

Security Assertion Markup Language on XML-pohjainen standardi, joka tarjoaa määrittelyn käyttäjien autentikointi- ja valtuutustietojen jakamiseen eri toimialueiden välillä. SAML on OASIS-yhtymän (Organization for the Advancement of Structured Infrormation Standards) koostaman SSTC-komitean (Security Services Technical Committee) vuonna 2002 luoma tuote. SAML 2.0 julkaistiin vuonna 2005. [12.]

SAML on luotu käsittelemään SSO:n ongelmia verkossa. Sisäverkon SSO-ratkaisuja on runsaasti, mutta laajentuminen ulkoverkkoon on osoittautunut hankalaksi. Luodut tekniikat ovat harvoin yhteensopivia, joten palveluntarjoaja voi joutua lisäämään sovellukseensa tuen usealle eri kertakirjautumistekniikalle. [12.]

5.1 Historia

OASIS perustettiin vuonna 1993, jolloin sen nimi oli SGML Open. Nimensä mukaan se pyrki edistämään XML:n ja HTML:n isän SGML:n (Standard Generalized Markup Language) käyttöönottoa. SGML Open koostuikin kielelle työkaluja myyvistä toimijoista. Kun XML-kielen suosio nousi, SGML Open -ryhmittymä kohdisti huomionsa XML:hen ja vaihtoi nimekseen OASIS Open. Lisäksi se siirsi voimavarojaan teknisiin määrittelyihin ja pois markkinoinnista. OASIS kehittää useita XML-pohjaisia standardeja kuten OpenDocument-formaattia, jota Open Office käyttää. [14.]

OASIS Openin alainen komitea SSTC kokoontui ensimmäistä kertaa vuonna 2001. Sen tavoitteena oli määritellä XML-viitekehys tunnistus- ja valtuutustietojen luomiseen ja vaihtamiseen. Vuoden 2002 marraskuussa julkaistiin OASIS-standardi SAML V1.0.

Syyskuussa 2003 SSTC julkaisi SAML V1.1 -standardin pienin parannuksin. Maaliskuussa 2005 SAML V2.0 tuli julki merkittävine uusine ominaisuuksineen ja parannuksineen. [14.]

5.2 Osapuolet ja roolit

SAML-viestinvaihtoa käydään vähimmillään vakuuttavan ja luottavan osapuolen välillä. Vakuuttava osapuoli on järjestelmä, joka luo SAML-vakuutuksia. Sitä kutsutaan myös SAML-auktoriteetiksi. Luottava osapuoli hyödyntää vakuutuksia. Monissa tapauksissa käyttäjä itse on vakuuttava osapuoli. Osapuolille määritellään eri toimialueiden välillä tapahtuvassa kertakirjautumisessa kolme roolia:

- toimeksiantaja (engl. Principal)
- henkilöllisyydentarjoaja (engl. Identity Provider)
- palveluntarjoaja (engl. Service Provider). [7.]

Toimeksiantaja on yleisesti käyttäjä, jonka tarvitsee todentaa itsensä. Palveluntarjoaja on esimerkiksi sovellus, johon käyttäjältä vaaditaan tunnistautumista. Henkilöllisyydentarjoaja on luotettu taho, joka antaa käyttäjälle sähköisen henkilöllisyyden palveluntarjoajaa varten. Henkilöllisyydentarjoajasta käytetään jatkossa myös lyhennettä IdP toiston estämiseksi. Palveluntarjoajasta käytetään lyhennettä SP. Lyhenteet ovat yhdenmukaisia SAML:n virallisen dokumentaation kanssa. [12.]

SAML käsittelee käyttötapauksesta, jossa käyttäjä pyytää pääsyä suojattuun sisältöön SP:lta. SP pyytää ja saa IdP:lta vakuutuksen käyttäjästä. Saadun tiedon perusteella palveluntarjoaja päättää, mihin sisältöön toimeksiantajalla on oikeudet. Henkilöllisyydentarjoaja voi vaatia toimeksiantajaa antamaan lisätietoja itsestään ennen kuin palveluntarjoaja saa varmennuksen käyttäjästä. [12.]

SAML tarjoaa määritykset SP:n ja IdP:n väliselle kanssakäynnille varsinkin toimeksiantajan henkilöllisyyteen liittyvissä asioissa. Useat palveluntarjoajat voivat käyttää samaa

henkilöllisyydentarjoajaa. Yksi palveluntarjoaja voi myös käyttää useita henkilöllisyydentarjoajia. [12.]

5.3 Assertio

Assertio eli vakuutus on SAML-osapuolen kokoama tietopaketti XML-muodossa. Assertio on tyypillisesti upotettu osaksi HTTP POST -kutsua tai XML-enkoodattua SOAP-viestiä. Vakuutus sisältää yhden tai useamman toteamuksen (statement), jotka viittaavat Subject -kenttään (koodiesimerkki 1.). Subject -kentästä käytetään jatkossa sanaa aihe. [13.]

SAML 2.0 määrittelee kolme erilaista tyyppiä assertiolle:

- Todennus-assertion luo osapuoli, joka on onnistuneesti todentanut käyttäjän. Vähimmillään se kertoo, milloin ja miten aihe todennettiin.
- Attribuutti-assertio sisältää henkilökohtaisia tietoja aiheesta kuten sähköpostiosoitteen.
- Valtuutuspäätös-asserio määrittelee, mihin resursseihin käyttäjällä on oikeus kuten, onko käyttäjällä oikeus muokata artikkelia. [7.]

Niin sanottu kantajavakuutus on tärkeä osa SAML-vakuutuksia ja selainkertakirjautumisen mahdollistamista [13.]. Koodiesimerkissä 1 esitetään henkilöllisyydentarjoajan luomaa kantajavakuutusta, joka lähetetään palveluntarjoajalle. Esimerkissä oleva vakuutus sisältää todennusassertion (AuthnStatement). Henkilöllisyydentarjoaja sijaitsee verkko-osoitteessa https://idp.client.com/ ja palveluntarjoajan osoitteessa https://sp.ims.fi/.

```
<saml:Assertion
   xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
   Version="2.0"
   IssueInstant="2014-04-23T09:12:05">
  <saml:Issuer>http://idp.client.com/adfs/services/trust</saml:Issuer>
  <ds:Signature
     xmlns:ds="http://www.w3.org/2000/09/xmldsig#">...</ds:Signature>
  <saml:Subject>
    <saml:NameID
       Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress">
       karri.korhonen@client.com
    </saml:NameID>
    <saml:SubjectConfirmation
       Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
      <saml:SubjectConfirmationData
         InResponseTo="aaf23196-1773-2113-474a-fe114412ab72"
         Recipient="https://sp.ims.fi/saml"
         NotOnOrAfter="2014-04-23T09:20:00Z"/>
    </saml:SubjectConfirmation>
  </saml:Subject>
  <saml:Conditions
     NotBefore="2014-04-23T09:10:00Z"
     NotOnOrAfter="2014-04-23T09:20:00Z">
    <saml:AudienceRestriction>
      <saml:Audience>https://sp.ims.fi/</saml:Audience>
    </saml:AudienceRestriction>
  </saml:Conditions>
  <saml:AuthnStatement
     AuthnInstant="2014-04-23T09:12:00"
     SessionIndex="b07b804c-7c29-ea16-7300-4f3d6f7928ac">
    <saml:AuthnContext>
      <AuthnContextClassRef>
         urn:federation:authentication:windows
      </AuthnContextClassRef>
    </saml:AuthnContext>
  </saml:AuthnStatement>
</saml:Assertion>
```

Koodiesimerkki 1. SAML-assertio [7.]

Assertioesimerkissä “saml:”-merkintä viittaa SAML 2.0 -nimiavaruuteen. Huomioitavia lapsielementtejä ovat:

- Issuer-elementti sisältää henkilöllisyydentarjoajan yksilöllisen tunnisteen.
- ds-nimiavaruuden Signature-elementissä on koko assertion eheyden todistava digitaalinen allekirjoitus(ei näkyvissä).
- Subject-kenttä osoittaa palveluntarjoajalle todennetun toimeksiantajan. Esimerkissä käyttäjän tunnuksena käytetään sähköpostiosoitetta.
- Conditions antaa rajaehdot vakuutuksen hyväksyttävyydelle.
- AuthnStatement-elementti sisältää lisätietoja henkilöllisyydentarjoajan käyttämistä todennuskeinoista.

Esimerkin mukaisessa tapauksessa vakuutus kertoo, että henkilöllisyydentarjoaja osoitteessa https://idp.client.com lähetti assertion 23. huhtikuuta 2014 kello 9:12:05 virallisen yleisajan mukaan. Palveluntarjoaja osoitteessa https://sp.ims.fi sai tietoa aiheesta, joka tunnistetaan sähköpostiosoitteella karri.korhonen@client.com. Autentikointiassertiossa mainitaan, että aihe on todennettu 23. huhtikuuta 2014 kello 9:12:00. Todennus tehtiin Windowsin sisäänrakennetulla autentikoinnilla.

5.4 Protokollat

SAML-protokolla kuvailee, miten tietyntyyppinen SAML-elementti kuten assertio pakataan pyyntö- tai vastausviestin sisälle. Samalla protokolla kertoo, miten vastaanotettava viesti puretaan. [12.] SAML määrittelee lukuisia pyyntö- ja vastausprotokollia:

- todennuksen pyyntöprotokolla (engl. Authentication Request Protocol)
- kertauloskirjautumisprotokolla (engl. Single Logout Protocol)
- vakuutuksen kysely- ja pyyntöprotokolla (engl. Assertion Query and Request Protocol)
- artefaktin ratkaisuprotokolla (engl Artifact Resolution Protocol)
- nimitunnisteen hallinnointiprotokolla (engl. Name Identifier Management Protocol)
- nimitunnisteen kartoittamisprotokolla (engl. Name Identifier Mapping Protocol). [7.]

Kertauloskirjautumisprotokolla määrittelee mekanismin, jolla toimeksiantaja tai muu ylläpitävä taho voi kirjata käyttäjän ulos kaikista henkilöllisyydentarjoaja piirissä toimivista palveluista. Vakuutuksen kysely- ja pyyntöprotokollassa rajaa kyselyjoukon, joilla SAML-vakuutuksia voi hankkia. Pyynnöllä kysytään assertioita lähettävältä osapuolelta aiemmin lähetettyä vakuutusta sen tunnisteen perusteella. Kyselyssä luottava osapuoli voi kysyä lisätietoja aiheesta. [7.]

Artifaktin ratkaisuprotokolla mahdollistaa tavan lähettää lyhyitä tietyn pituisia SAML-viestejä nimeltä artifaktit. Artifaktin vastaanottaja kysyy protokollan avulla lähettäjältä viestin purkua ja alkuperäisen viestin palautusta. Nimitunnisteen hallinnointiprotokolla tarjoaa tekniikan toimeksiantajaan viittaavan tunnisteen formaatin tai arvon muuttami-

seen. Protokollan avulla voi myös poistaa viittauksen. Sekä palvelun- että henkilöllisyydentarjoaja voi pyytää muutosta. [7.]

Nimitunnisteen kartoittamisprotokollan avulla voi linkittää SAML-nimitunnisteen toiseen tunnisteeseen. Se sallii esimerkiksi palveluntarjoajaa pyytämään henkilöllisyydentarjoajalta tunnistetta käyttäjälle, jotta tämä voi käyttää toista palvelua kahden eri palvelun integrointitilanteessa. [7.]

Todennuspyyntöprotokolla tarkentaa tapaa, jolla toimeksiantaja voi tilata autentikoivia tietoja itsestään henkilöllisyydentarjoajalta kirjautumista varten. Kuten suurin osa protokollista se on uusi ominaisuus SAML 2.0 -versiossa. Protokolla mahdollisti palveluntarjoajan aloittaman kirjautumisen (engl. SP-initiated login), jossa palveluntarjoaja lähettää ensimmäisen käyttäjän tunnistukseen päättyvän SAML-viestiketjun. Vanhemmat versiot tarjosivat vain henkilöllisyydentarjoajan yksisuuntaisen käyttäjäautentikoinnin (IdP-initiated login). [7; 16.]

5.5 Sidokset

SAML-sidokset selittävät yksityiskohtaisesti, miten eri SAML-viestit voi välittää käyttämällä olemassa olevia kuljetusprotokollia kuten HTTP POST -protokollaa. POST ja uudelleenohjaus ovat yleisimpiä sidoksia selainkertakirjautumisessa esimerkiksi, jos palveluntarjoaja lähettää pyyntönsä HTTP-uudelleenohjauksena ja henkilöllisyydentarjoaja voi vastata käyttämällä HTTP POST:a. SAML 2.0 tukee seuraavia sidoksia:

- HTTP Uudelleenohjaussidos
- HTTP POST -sidos
- SAML SOAP -sidos
- PAOS-sidos
- HTTP-artefaktisidos
- SAML URI -sidos. [7.]

HTTP POST -sidos tarkentaa, miten SAML-viestejä voi lähettää base64-kooditettuna HTML-lomakkeen sisällä. SAML SOAP -idos määrittelee viestit, jotka lähetetään SOAP

1.1 -viesteinä. Lisäksi käänteinen SOAP eli PAOS-sidos täsmentää monitasoisen SOAP- ja HTTP-viestien vaihdon, joka sallii selaimen toimia vastaajana SOAP-viesteille. Sidosta käytetään henkilöllisyydentarjoajan löytämisprofiilissa. [7.]

HTTP-artefaktisidos tarkentaa, miten artefaktit lähetetään selaimen välityksellä. Artefakti joko on osana HTML-lomaketta tai kyselymerkkijonona URL:ssa. SAML URI -sidoksessa määritellään, kuinka olemassa olevan SAML-vakuutuuksen voi hakea tarjotusta URI:sta. [7.]

SAML:n protokollaviestit tulevat usein URL:n parametreina HTTP GET -kutsussa. HTTP/1.1 virallisen määritelmän mukaan verkko-osoitteella ei ole enimmäispituutta [2.]. Esimerkiksi Internet Explorer 10 -selain tukee kuitenkin vain 2083 merkin pituista URL:a [3.]. Tämän takia uudelleenohjaus sopii parhaiten lyhyisiin viesteihin, joiden allekirjoittaminen ei ole välttämätöntä, kuten todennuksen pyyntöä varten.

Uudelleenohjaussidoksella lähetetyissä pyynnöissä ja vastauksissa on mukana SAMLRequest- tai SAMLResponse-kyselyparametri kertomassa viestin tyypistä (koodiesimerkki 2.). Ennen lähetystä viesti tiivistetään, base64-kooditetaan ja URL-kooditetaan tässä järjestyksessä. Viestin vastaanotossa tehdään samat asiat päinvastaisessa järjestyksessä. [13.]

```
https://idp.client.com/adfs/ls?SAMLRequest=fZBBT4NAEIXPNul%2FaLizCwiJTqAJ2
oNNaiQFPXgxKyx2EtjFncX4813ApBoTj%2FPmfW8mLyXRdwPkoz2po3wfJdnNenWRE0ljUatbr
WjspSml%2BcBaPh4PmXeydiDgnAaGPbEW%2BRTiTdzO8ajERJ592Ays7lAqy2rdc9G0xDviM7D
fZR42boUtSvMSepsnaWjGIxYsFqJR7hVZoaxTgzD2g9iPLqswgTiBKGHXV9HzbC2MtrrW3Q2qB
tVb5o1GgRaEBEr0ksDWUOb3B3DZ8LqYCO6qqvCLh7KaMz77ThHMtfzPD9%2FHvK2j0gmA%2BVW
z%2FVNQyn%2Bu16tl%2Fl27k78A
```

Koodiesimerkki 2. HTTP-uudelleenohjaussidoksella luotu todennuspyyntö

5.6 Profiilit

SAML-profiilit määrittelevät, miten SAML-assertioita, -protokollia ja -sidoksia tulee yhdistellä ja rajoittaa, jotta vaadittu yhteentoimivuus saavutettaisiin tietyntyyppisessä skenaariossa. Selainkertakirjautumisen profiili on selitetty laajemmin luvussa 5.6.1. SAML 2.0 tuntee seuraavat profiilit:

- Selainkertakirjautumisprofiili (engl. Web Browser SSO Profile)
- tehostettu asiakasohjelma- ja välityspalvelinprofiili (engl. Enhanced Client and Proxy Profile)
- henkilöllisyydentarjoajan löytämisprofiili (engl. IdP Discovery Profile)
- kertauloskirjautumisen profiili (engl. Single Logout Profile)
- vakuutusten kysely- ja pyyntöprofiili (engl. Assertion Query/Request Profile)
- artefaktin ratkaisuprofiili (engl. Artifact Resolution Profile)
- nimitunnisteen hallinnointiprofiili (engl. Name Identifier Management Profile)
- nimitunnisteet kartoitusprofiili (engl. Name Identifier Mapping Profile). [7.]

Selainkertakirjautumisen profiili määrittelee, miten SAML-osapuolten tulee käyttää todennuspyyntöprotokollan ja SAML-vastausten viestejä sekä vakuutuksia, kun halutaan saavuttaa kertakirjautuminen perinteisissä selaimissa. Profiili tarkentaa, miten uudelleenohjaus-, POST- ja artefaktiviestien tulisi toimia eri yhdistelmillä [7.]. Luvussa 6.4. esitetään selainkertakirjautumisen profiili koodiesimerkein IMS-ohjelmiston parissa.

Tehostettu asiakasohjelma- ja välityspalvelinprofiili on erikoistunut kertakirjautumisprofiili. Se koskee asiakasohjelmia, jotka eivät ole selaimia, mutta voivat käyttää PAOS- ja SOAP-sidoksia. Henkilöllisyydentarjoajan löytämisprofiili määrittelee yhden mahdollisen keinon, jolla palveluntarjoaja voi tutkia, keitä henkilölllisyydentarjoajia käyttäjä on aiemmin hyödyntänyt. Muut profiilit pitkälti kuvaavat saman nimisten sidosten toimintaa. [7.]

5.7 SAML-metadata

Kun henkilöllisyydentarjoaja vastaanottaa todennuspyynnön palveluntarjoajalta, käy se läpi omat SAML-metadatansa, joista näkyy, onko palveluntarjoaja luotettujen palveluiden joukossa. Henkilöllisyydentarjoaja myös katsoo vastausosoitteen palveluntarjoajan ilmoittamasta metadatojen osoitteesta. Osapuolet voivat lisäksi tarkastaa, että SAML-viestien allekirjoituksissa käytetyt sertifikaatit ovat samoja kuin lähettäjät ovat ilmoittaneet metadatoissaan. [13.]

SAML-metadata viittaa niihin konfigurointitietoihin, mitä henkilöllisyyden- ja palveluntarjoaja tarvitsevat toisiltaan, jotta ne voisivat onnistuneesti kommunikoida keskenään. Metadatojen ideana on myös nopea tietojen jakaminen. Kun henkilöllisyydentarjoajan tarvitsee lisätä uusi palveluntarjoaja luottamusverkkoonsa, voi tietojen syöttämiseksi riittää pelkkä palveluntarjoajan metadatojen verkko-osoite. Metadatojen onkin tarkoitus olla suoraan jaettavissa.

6 SAML-toiminnallisuuden lisääminen IMS-ohjelmistoon

SAML-kertakirjautumisen lisääminen IMS-ohjelmistoon vaatii yksinkertaisimmillaan SAML-viestien lähettämiseen ja vastaanottamiseen kykenevän ohjelmiston. Toteutustapoja kertakirjautumisen käyttöönottoon on lukemattomia, joista useimmat olivat liian heikosti dokumentoituja.

6.1 Sivuutettuja toteutuksia

Aloitin projektin tutkimalla, miten SAML-viestien lähettäminen IMS-ohjelmistosta toteutettaisiin. Asiakasorganisaation yhdyshenkilö ehdotti jo projektin alkuvaiheessa heille tutuksi tullutta Shibboleth-järjestelmää. Shibboleth on avoimen lähdekoodin projekti, joka voi olla sekä palveluntarjoaja että henkilöllisyydentarjoaja. Järjestelmä tukisi myös RelayState-ominaisuutta. Kävi kuitenkin nopeasti selville, että Shibboleth ei sovi IMS-ohjelmiston rinnalle.

Shibbolethin käyttöönotto vaatii Apache-palvelinohjelman asentamista, kun IMS-ohjelmisto toimii Tomcat-alustan kanssa. Erikoistapauksia varten kahden uuden järjestelmän asentaminen on turhan raskasta. Suurin osa IMS-asennuksista ei käytä kertakirjautumista eikä Shibbolethissa nähty olevan hyötyä muissa käytössä olevissa kertakirjautumisratkaisuissa.

Suurin syy uusien järjestelmien käyttöönottoa vastaan oli kuitenkin ylläpidon vaivalloisuus. Ongelmatilanteissa ylläpitäjän täytyy tuntea ja havaita kahden vain harvoin käytetyn järjestelmän kompastuskivet. Lisäksi Apache ja Shibboleth vievät palvelimella käytettäviä tehoja IMS-ohjelmistolta, mikä on turhan raskasta pelkältä kirjautumisvaihtoeh-

dolta. SAML-kertakirjautumisen tuen lisäys suoraan IMS-ohjelmistoon vaikutti parhaalta vaihtoehdolta.

Erillisen ohjelmiston jälkeen etsin mahdollisimman helposti käyttöönotettavaa lähdekoodia. Parhaalta vaihtoehdolta vaikutti OneLogin, joka on yhdysvaltalainen pilvipohjainen henkilöllisyyden- ja pääsynhallintapalvelu. Yrityksen kotisivuilta voi ladata esimerkiksi Javalla toteutetun SAML-viestien lähettämis-, vastaanottamis- ja todentamistyökalun. OneLogin-yrityksen tavoitteena on antaa palveluntarjoajille ilmaiseksi tuki SAML-kertakirjautumiselle ja myydä käyttäjähallintopalveluita palveluntarjoajan omille asiakkaille.

Suurin ongelma OneLoginin yksinkertaisessa työkalussa oli laajentamisen vaivalloisuus. Alkuperäinen työkalu ei tarjonnut suoraan viestien allekirjoittamista. Sinänsä allekirjoittamattomat viestit eivät olisi olleet suuri ongelma. Lähdekoodia voi laajentaa ja allekirjoittamisen lisätä myöhemmin. Asiakkaan henkilöllisyydentarjoaja salli kokeilemistarkoituksessa allekirjoittamattomat SAML-viestit, mutta kirjautuminen ei onnistunut.

Asiakkaan ylläpitäjä löysi lopulta virheilmoituksen AD FS -ohjelmistosta, jonka perusteella IMS:n lähettämässä SAML-viestin rakenteessa oli vikaa. Tutkimme asiaa yhdessä OneLoginin teknisen tuen kanssa, mutta emme löytäneet syyllistä. Rakenteellisten ongelmien lisäksi työkalusta puuttui yhä tuki viestin allekirjoittamiselle.

6.2 Spring ja OpenSAML

Spring-ohjelmistokehys sekä todentamiseen ja valtuuttamiseen keskittyvä Spring Security-ohjelmistokehys olivat tulossa mukaan IMS-ohjelmistoon SAML-projektin aikana. Spring-ohjelmistokehykselle on tarjolla useita laajennuksia, joista yksi on Spring Security SAML. Laaja käyttäjäkunta, vertaistuki sekä lukemattomat esimerkit vakuuttivat minut, että Spring-ohjelmistokehys olisi paras vaihtoehto.

Spring-ohjelmistokehyksen käyttöönotto oli suunniteltu valmistuvan huomattavasti myöhemmin kuin SAML-kertakirjautumisen. Koska Spring-projektia ei voinut vielä käyttää oman työni toteuttamisessa, päädyin tutkimaan OpenSAML-kirjastoa. Spring Security SAML -laajennus käyttää toteutuksessaan OpenSAML-kirjastoa.

OpenSAML 2 on toinen painos kokoelmasta avoimen lähdekoodin Java- ja C++ -kirjastoja. Se on Shibboleth-projektin oheistuote. OpenSAML ei tarjoa valmista pakettia SAML-palvelun- tai henkilöllisyydentarjoajaksi ryhtymiseen vaan tarkoitus on avustaa sovelluskehittäjiä SAML-viestien sekä profiilien turvallisessa käytössä. Kokoelma tukee SAML:n versioita 1.0, 1.1 ja 2.0. [10.]

6.3 SAML-kertakirjautumisen käyttöönotto

SAML-kertakirjautumisen käyttöönotto IMS-ohjelmistossa tapahtuu asennusnäkymän kautta. IMS:n asennusnäkymä on JSP-sivu, joka tulee näkyviin vain asennuksen tai päivityksen yhteydessä. Kun asennus on valmis, siihen liittyvät JSP-, HTML- ja Javascript-tiedostot poistetaan palvelimelta. Henkilöllisyydentarjoajapalvelun ja IMS:n täytyy jakaa keskenään seuraavat tiedot, että SAML-kertakirjautumisen käyttö on mahdollista:

- Issuer-elementin arvo eli palvelun käyttämä yksilöivä tunniste itsestään.
- Verkko-osoite, josta henkilöllisyydentarjoaja vastaanottaa SAML-autentikointikutsut.
- Verkko-osoite, josta IMS lähettää SAML-autentikointikutsun.
- Vakuutuksen kuluttajapalvelun osoite eli verkko-osoite, johon IMS vastaanottaa SAML-autentikointivastauksen.
- SAML-viestien allekirjoittamiseen tarkoitetut base64-kooditetut julkiset avaimet tekstimuodossa tai tiedostona.
- Toimivan käyttäjäsynkronoinnin

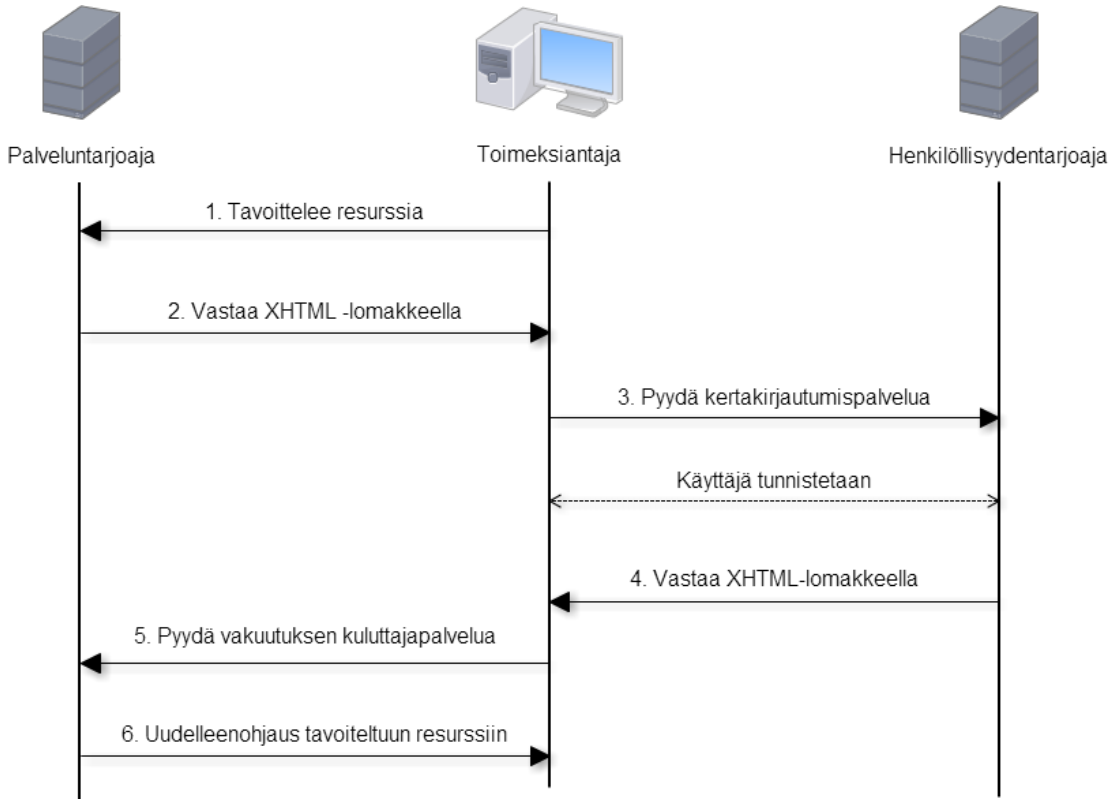
Issuer-elementin tehtävänä on yksilöidä palvelunsa vastapuolelle. Jos IMS tukee vain yhtä henkilöllisyydentarjoajaa, tunnisteella on vain nimellistä arvoa. Esimerkiksi osoite, josta henkilöllisyydentarjoaja lähettää SAML-viestinsä, on riittävän yksilöllistä. Palvelun on myös toimittava samassa kellonajassa, että viestien aikaleimoihin voi luottaa. IMS-palvelin synkronoi kellonaikansa NTP-palvelulla (engl. Network Time Protocol). Tämä pitää jatkuvasti huolen palvelimen ajasta.

Kertakirjautumiseen liittyy vahvasti käyttäjätietojen synkronointi asiakkaan AD-palvelimen sekä IMS:n välillä. Kertakirjautuminen ei voi toimia luotettavasti, jos IMS-ohjelmistolla ei ole ajankohtaista tietoa asiakkaan henkilöstön muutoksista. Käyttäjien

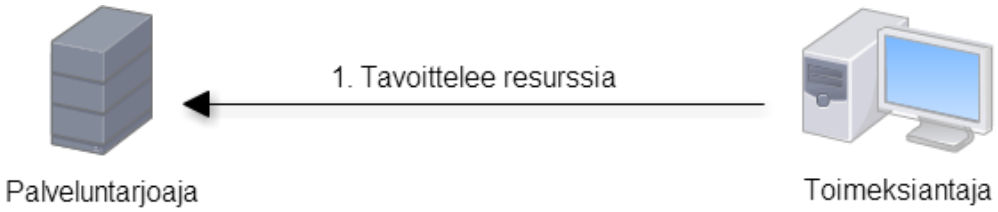
lisäys käsin voi aiheuttaa ristiriitoja tuotavien ja olemassaolevien käyttäjien tietojen kohdalla, joten sitä ei myöskään sallita. Vaikka käyttäjätiedot eivät olisikaan ajan tasalla, ei virhetilanteesta koidu tietoturvariskiä. Henkilöllisyydentarjoajalla on lopullinen päätäntävalta, kenelle valtuudet kirjautua IMS-ohjelmistoon annetaan.

6.4 Selainkertakirjautumisen profiili IMS-ohjelmistossa

SAML 2.0:n virallisten määritysten mukaan selainkertakirjautumisen profiiliin liittyy kolme osapuolta eli palveluntarjoaja, henkilöllisyydentarjoaja ja selainta käyttävä toimeksiantaja. Työssä IMS-ohjelmisto toimii palveluntarjoajana ja asiakkaan ADFS 2.0 henkilöllisyydentarjoajana. IMS tukee SAML-viestien lähetyksessä vain HTTP POST –sidosta. Kuvat 7 - 12 kuvaavat selainkertakirjautumisen profiilin askelia IMS- ja AD FS 2.0 -ohjelmistojen sekä toimeksiantajan eli selaimen välillä.

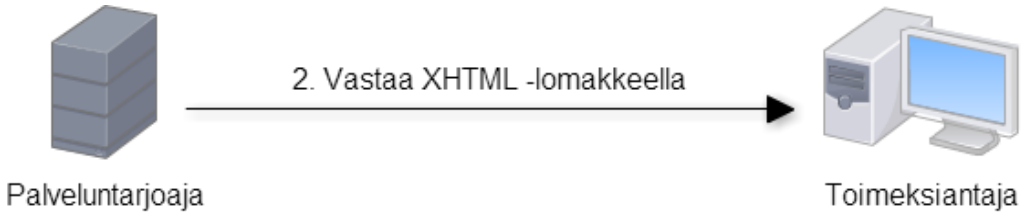


Kuva 7. Selainkertakirjautumisen profiili HTTP POST -sidoksilla



Kuva 8. Toimeksiantaja siirtyy palveluntarjoajan osoitteeseen

Kuvassa 8 käyttäjä tavoittelee palveluntarjoajan sisältöä siirtymällä IMS-ohjelmistoon selaimella. Mikäli käyttäjällä on IMS:in voimassa oleva sessio selaimessa, ei muita vaiheita tarvita, ja selain siirtyy tavoiteltuun resurssiin. Jos käyttäjää ei voi välittömästi todentaa, osoite asetetaan sessiomuuttujaksi ja siirrytään vaiheeseen kaksi. Erillistä sessiomuuttujaa ei tarvita, jos käyttäjä pyrkii IMS:n päänäkymään.



Kuva 9. Vastaa XHTML-lomakkeella toimeksiantajan pyyntöön

Käyttäjällä ei ole IMS:ssä voimassa olevaa sessiota, joten IMS kasaa todennuspyynnön OpenSAML-kirjaston avulla. Kappaleessa 6.6 kuvataan koodiesimerkein todennuspyynnön kokoaminen sekä tarjotaan esimerkki valmiista todennuspyynnöstä. IMS vastaa toimeksiantajalle XHTML-lomakkeella. Lomake sisältää todennuspyynnön sekä osoitetiedon käyttäjän tavoittelemasta sisällöstä (koodiesimerkit 3 ja 4).

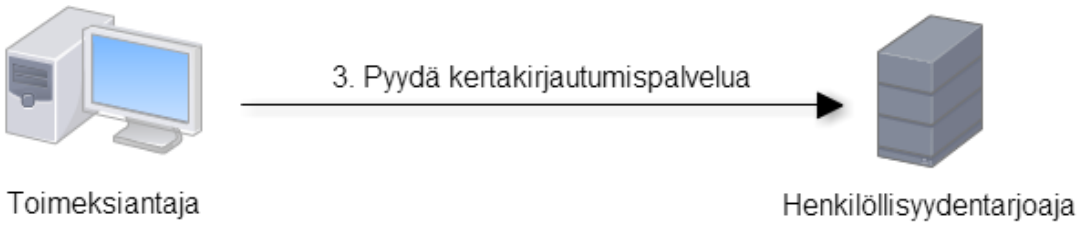
```
1  <form action="https://idp.client.com/adfs/ls/" method="post">
2      <input type="hidden" name="SAMLRequest" value="'authnRequest'">
3      <input type="hidden" name="RelayState" value="https://sp.ims.fi/resource/">
4      <input input type="submit" value="Continue">
5  </form>
```

Koodiesimerkki 3. XHTML-lomakkeen tiedot

```
<samlp:AuthnRequest
  AssertionConsumerServiceURL="https://sp.ims.fi/saml"
  Destination="https://idp.client.com/adfs/ls/"
  ID="identifier_1" Version="2.0"
  IssueInstant="2014-04-23T15:45:25.982Z"
  ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol">
  <saml:Issuer>https://sp.ims.fi</saml:Issuer>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    ...
  </ds:Signature>
</samlp:AuthnRequest>
```

Koodiesimerkki 4. SAML-todennuspyynnön XML ilman allekirjoituselementtiä

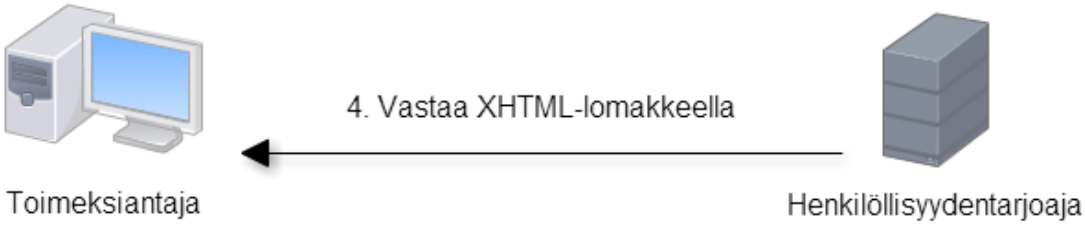
Henkilöllisyydentarjoajan palveluun vievä URL-osoite on tallennettu, kun IMS-ohjelmiston asennustoiminto edellisellä kerralla ajettiin. Tomcatin tai kontekstin käynnistyksen yhteydessä osoite asetetaan muistiin. Osoitteen vaihtaminen ei onnistu ilman Tomcatin tai IMS-kontekstin uudelleenkäynnistämistä.



Kuva 10. Toimeksiantaja välittää todennuspyynnön henkilöllisyydentarjoajalle

Toimeksiantaja lähettää todennuspyynnön POST-kutsuna henkilöllisyydentarjoajalle. Base64-kooditettu viesti on HTTP POST -kutsun lomaketiedoissa nimellä SamlRequest. Myös tavoiteltu osoite base64-kooditetaan ja lähetetään lomaketiedoissa nimellä RelayState. Jos käyttäjä on siirtynyt suoraan IMS:n etusivulle sen sijaan, että olisi tavoitellut jotain tiettyä sisältöä, RelayState-tietoa ei lisätä.

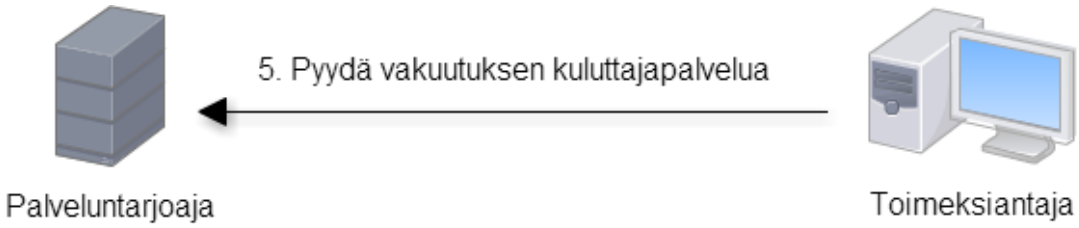
Henkilöllisyydentarjoajalle lähetetyssä viestissä on digitaalinen allekirjoitus, joka näkyy viestissä Signature-elementtinä. Insinöörityössä käytän allekirjoitukseen RSA-SHA-1- ja SHA-1-algoritmeja. Signature-elementti sisältää myös X.509-sertifikaatin, jota IMS käyttää lähtevien viestien allekirjoittamiseen. Sertifikaatti on tallennettuna asiakkaan henkilöllisyydentarjoajan palveluun. URL-osoitteet, joihin IMS lähettää ja joista IMS voi vastaanottaa SAML-viestejä, ovat asiakkaan palvelun tiedossa. Edellämainittujen tietojen avulla henkilöllisyydentarjoaja varmistaa, että viestit lähettävä palvelu on luotettava kumppani.



Kuva 11. Henkilöllisyydentarjoaja vastaa todennuspyyntöön XHTML-lomakkeella

Kun henkilöllisyydentarjoaja on varmistanut, että SAML-viestin lähettänyt taho kuuluu sen luottamusverkkoon, on käyttäjän vuoro esittäytyä palvelulle. Mikäli käyttäjällä ei vielä ole voimassa olevaa sessiota henkilöllisyydentarjoajan palvelun kanssa, selaimeen tulee näkyviin kirjautumisikkuna. Todennusvastaus lähetetään välittömästi, mikäli sessio on olemassa. Kuvassa 11 käyttäjä on todentanut itsensä henkilöllisyydentarjoajalle

Henkilöllisyydentarjoaja lähettää IMS:lle vastauksen todennuspyyntöön, kun käyttäjä on onnistuneesti todentanut itsensä. Todennusvastauksessa on tunniste, joka yksilöi käyttäjän sekä IMS:n puolella että henkilöllisyydentarjoajan palvelussa. Lisäksi vastauksessa on tieto todennuspyynnön hyväksynnästä. Todennusvastauksen lisäksi *RelayState*-parametri palautetaan takaisin IMS:lle, mikäli sitä oli prosessin alussa asetettu.

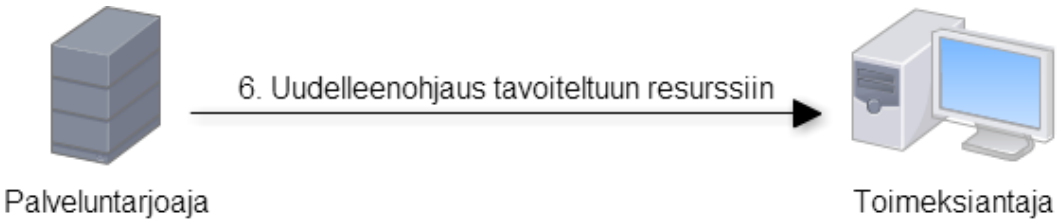


Kuva 12. Toimeksiantaja välittää henkilöllisyydentarjoajan vastauksen vakuutuksen kuluttajapalveluun

Kuvassa 12 näkyy, kuinka IMS vastaanottaa todennuspyynnön henkilöllisyydentarjoajalta verkko-osoitteeseen, jota kutsutaan vakuutuksen kuluttajapalveluksi (engl. Assertion Consumer Service). Mikäli viestiä ei lähetetä käyttäen HTTP POST -sidosta, IMS ei hyväksy vastausta. Kappaleessa 6.7 esitetään tiivistetysti, miten IMS-ohjelmisto todentaa todennusvastauksen eli *SAMLResponse*-viestin.

Todennusvastaus ja sen lähettäjä täytyy tarkastaa ennen kuin käyttäjää voi päästää kirjautumaan IMS-ohjelmistoon. Vastausviestistä katsotaan, että sen lähettäjä tunnetaan, aikaleimat eivät ole vanhentuneita ja että käyttäjän tunniste on olemassa. Lisäksi

autentikointivastauksen on oltava vastaus alkuperäiseen autentikointipyyntöön. Pyyntöä lähetettäessä IMS tallentaa viestin *ID*-tiedon sessiomuuttujaan. Tunnisteen tulee vastata henkilöllisyydentarjoajan viestissä ilmoitettua *InResponseTo*-tietoa. Kaikkein tärkein tieto on silti digitaalinen allekirjoitus, jonka perusteella voi varmistaa, ettei viestiä ole muokattu matkalla.



Kuva 13. Uudelleenohjaus tavoiteltuun resurssiin

Toimeksiantaja ohjataan lopulta IMS-ohjelmistoon tai RelayState-parametrin osoittamaan resurssiin. Viimeinen askel on esitetty kuvassa 13. Jatkossa käyttäjän todennus tapahtuu IMS-ohjelmistossa sisäisesti, kunnes käyttäjä kirjautuu ulos, sulkee selaimen tai käyttäjän sessio IMS:ssä vanhenee.

6.5 Todennuspyynnön kokoaminen

OpenSAML-kirjasto tarjoaa kattavat työkalut SAML-viestien kokoamista, lähettämistä ja purkamista varten. Velocity-kirjasto sisältää valmiin sapluunan XHTML-lomakkeesta, jonka IMS välittää henkilöllisyydentarjoajalle. Ennen lomakkeen lähettämistä luodaan viitekehys, johon on helppo lisätä yksitellen lähetykseen tarvittavat tiedot (koodiesimerkki 5).

```
1  BasicSAMLMessageContext msgContext = new BasicSAMLMessageContext();
2  msgContext.setOutboundMessageTransport(
3      new HttpServletResponseAdapter(response, request.isSecure())
4  );
5  msgContext.setOutboundSAMLMessageSigningCredential('IMS_KEYPAIR');
6  msgContext.setPeerEntityEndpoint(addEndpoint());
7  msgContext.setOutboundSAMLMessage(buildAuthnRequest());
8  msgContext.setRelayState('RELAYSTATE');
9
10 HTTPPostEncoder encoder =
11     new HTTPPostEncoder(SamlUtil.getVelocityEngine(), "/templates/saml2-post-binding.vm");
12 encoder.encode(msgContext);
```

Koodiesimerkki 5. SAML-autentikointipyynnön luominen ja lähettäminen OpenSAML-kirjastolla

SAML-viestin lähettäminen on hyvin suoraviivainen operaatio (koodiesimerkki 5). Lähtevän viestin digitaaliseen allekirjoitukseen tarvittavan avainparin luominen on jätetty pois esimerkistä. IMS kuitenkin allekirjoittaa lähtevät SAML-viestit. Samoin pois on jätetty *RelayState*-parametrin ottaminen sessiomuuttujasta. VelocityEngine-olion alustaminen tehdään erillisessä luokassa. Esimerkki vastaa pääpiirteissään lopullista lähdekoodia, mutta on silti hyvin karsittu versio.

Viidennessä koodiesimerkissä rivillä kuusi *addEndPoint*-metodi palauttaa verkko-osoitteen, johon XHTML-lomake lähetetään sekä osoitteen, johon henkilöllisyydentarjoajan tulisi vastata. Metodi *addIssuer* lisää IMS:n osoitteen *Issuer*-elementtiin. Esimerkissä esiintyvä *buildAuthnRequest*-metodi rakentaa nimensä mukaan itse todennuspyynnön.

Todennuspyyntö luodaan OpenSAML-kirjaston *AuthnRequestBuilder*-olion avulla. Pyynnön sisältö lisätään yksitellen kuten viestin viitekehyksen kanssa. Viestille ilmoitetaan muun muassa lähettämiseen käytettävä protokolla eli HTTP POST, myöntäjä eli *Issuer* ja lähtevän pyynnön *ID* eli tunniste, jota vasten todennuspyynnön vastaus tarkastetaan. Lähtevän viestin kellonaika asetetaan virallisen yleisajan eli UTC:n mukaan. Koodiesimerkki 6 esittää pyynnön rakentamisen kokonaisuudessaan.

```
1   AuthnRequestBuilder authnRequestBuilder = new AuthnRequestBuilder();
2   AuthnRequest authnRequest = authnRequestBuilder
3     .buildObject("urn:oasis:names:tc:SAML:2.0:protocol", "AuthnRequest", "samlp");
4
5   authnRequest.setForceAuthn(false);
6   authnRequest.setIsPassive(false);
7   authnRequest.setID('SAML_REQUEST_ID');
8   authnRequest.setIssueInstant( new DateTime(DateTimeZone.UTC) );
9   authnRequest.setDestination("https://idp.client.com/adfs/ls/");
10  authnRequest.setProtocolBinding(SAMLConstants.SAML2_POST_BINDING_URI);
11  authnRequest.setAssertionConsumerServiceURL("https://sp.ims.fi/saml");
12  authnRequest.setVersion(SAMLVersion.VERSION_20);
13  authnRequest.setIssuer( addIssuerElement() );
14  return authnRequest;
```

Koodiesimerkki 6. Todennuspyynnön kokoaminen

6.6 Todennusvastauksen vastaanotto ja purkaminen

IMS ottaa HTTP POST -sidoksella saapuneen SAML-todennusvastauksen vastaan vakuutuksen kuluttajapalvelun osoitteessa. Kun SAMLResponse-parametri on irrotettu XHTML-lomakkeesta, ensimmäinen asia, mitä tarkistetaan, on viestin status-elementti. Elementin arvona on tieto kirjautumisen onnistumisesta. Todennäköisin syy epäonnistumisesta kertovalle viestille on virheellinen SAML-todennuspyyntö. Epäonnistuneet kirjautumiset ja muut samanlaiset virhetilanteet eivät johda todennusvastauksen lähettämiseen, sillä henkilöllisyydentarjoaja pysäyttää käyttäjän etenemisen.

Tärkein yksittäinen asia, joka täytyy tarkastaa, on vastaanotetun viestin digitaalinen allekirjoitus. Jos allekirjoitus ei vastaa viestin sisältöä, todennäköistä on, että todennusvastausviestiä on muokattu matkalla ja IMS-ohjelmisto on mies välissä –hyökkäyksen kohde. Tarkastuksessa katsotaan ensin, että digitaalisen allekirjoituksen rakenne vastaa SAML-standardin hyväksymää rakennetta. Vasta tämän jälkeen tutkitaan, onko allekirjoitus pätevä. OpenSAML-tarjoaa työkalut tarkistuksiin ja allekirjoitusta verrataan asiakkaan ilmoittamaan sertifikaattiin.

Jos SAML-vastausviestin allekirjoitus on luotettava, voidaan hyvällä todennäköisyydellä sanoa, että viestiä ei ole käsitelty matkalla IMS-ohjelmistoon. Seuraava askel on tarkastaa viestin pääelementin aikaleima. On syytä epäillä, että luottamusverkon ulkopuolinen osapuoli yrittää toistaa kirjautumistapahtumaa lähettämällä uudelleen vanhan todennusvastauksen, jos viestin aikaleima on useita minuutteja vanhempi kuin palvelimen aika viestiä vastaanotettaessa.

IMS-toimintajärjestelmä asettaa todennuspyynnössä olevan yksilöllisen tunnisteen sessiomuuttujaksi viestin lähetyksen yhteydessä. Henkilöllisyydentarjoajan lähettämässä SAML-vastausviestissä täytyy olla mukana sama tunniste inResponseTo-kentän arvona. Jos arvoa ei ole asetettu tai se on väärä, on syytä epäillä, onko myöskään vastauksen aihe eli todennettava käyttäjä oikea. Syynä väärään tunnisteeseen voi esimerkiksi olla virhetilanne henkilöllisyydentarjoajan sovelluksessa.

Viimeinen tarkastus ennen assertio-elementin tutkimista on itse SAML-todennusvastausviestin lähettäneen osapuolen yksilöivä tunniste eli Issuer-elementti. IMS-ohjelmistoon asetetaan asennuksen yhteydessä henkilöllisyydentarjoajan kertoma

tunniste itsestään. Tunniste on myös henkilöllisyydentarjoajana toimivan palvelun SAML-metatiedoissa. IMS tukee vain yhtä henkilöllisyydentarjoajaa, jota Issuer-elementin arvon tulee vastata.

SAML-standardin mukaan luotettava assertion täytyy olla allekirjoitettu. Muita tarkastettavia tietoja ovat Issuer-elementti ja ehdot. Ehtoja ovat esimerkiksi aikaleimat, joiden välisenä aikana assertio on voimassa, sekä tieto assertion yleisöstä. Yleisöllä tarkoitetaan sitä toimialuetta eli verkko-osoitetta, jossa assertio on voimassa.

Jos assertion allekirjoitus on hyväksytty, seuraava askel on tarkastaa aihe eli Subject-elementti. Aiheen elementti sisältää myös aikaleimat, jolloin aihe on hyväksyttävissä sekä todennuspyynnön tunnisteen, joka todistaa, mihin pyyntöön henkilöllisyydentarjoaja vastaa ja kenelle vastaus on lähetetty. IMS-ohjelmiston selainkirjautumisen tapauksessa aiheena on käyttäjätunnus.

Kun henkilöllisyydentarjoajan lähettämästä viesti on varmistettu ja IMS-ohjelmisto voi todeta, että todennusvastauksen on lähettänyt luotettava taho, tehdään kirjautumisprosessin viimeinen tarkistus. IMS tarkistaa, löytyykö SAML-assertion aiheena olevaa käyttäjätunnusta tietokannasta. Jos käyttäjää ei löydy, on mahdollista, etteivät IMS:n käyttäjätiedot ole ajan tasalla. Vaihtoehtoisesti henkilöllisyydentarjoaja ei tee pääsynhallintaa omassa sovelluksessaan vaan antaa IMS:n ilmoittaa, ettei käyttäjällä ole olemassa olevaa tunnusta IMS:n toimialueella.

6.7 Jatkokehitys

Projektin aikana Spring-ohjelmistokehystä oltiin vasta ottamassa käyttöön. Jatkokehityksen kannalta olisi suotavaa, että SAML-toiminnallisuutta siirrettäisiin Spring Security SAML-laajennuksen alaisuuteen. Samalla voitaisiin toteuttaa SAML-metadatojen luominen. Asiakas voisi tällöin syöttää henkilöllisyydentarjoajana toimivalle palvelulleen suoraan XML-tiedoston, joka sisältää tarvittavat tiedot luettavassa muodossa kertakirjautumisen käyttöönotosta asiakkaan päädyssä.

Mikäli Spring-ongelmistokehystä ei tulla käyttämään, Velocity-kirjastosta olisi hyvä luopua. Velocity sisältää lukemattomia sapluunoita erilaisille viesteillä ja on muutenkin

kookas kirjasto. IMS käyttää Velocity-kirjastoa vain todennuspyyntöjen lähettämisessä viestin rungon määrittelemiseen. Rungon kasaamisesta voi yhtä hyvin tehdä oman pienen toteutuksensa sen sijaan, että säilytetään asennuksessa mukana suurta määrää turhaa koodia.

Työssä IMS-ohjelmistoon on lisätty tuki vain HTTP POST -sidoksen mukaisille viesteille. Lisäksi lähtevät viestit allekirjoitetaan aina, koska asiakkaan henkilöllisyydentarjoajana toimiva palvelin ei hyväksynyt allekirjoittamattomia viestejä IMS:n osalta. SAML todennuspyyntöjen-lähetysprotokolla ei välttämättä vaadi autentikointipyyntöviesteiltä digitaalista allekirjoitusta. Jos allekirjoitusta ei lisättäisi, todennuspyynnön voisi tällöin lähettää myös uudelleenohjaussidoksella.

Insinöörityön varsinaisen aiheen eli SAML-kertakirjautumisen lisäksi työssä toteutettiin käyttäjien synkronointi SSL-suojatun yhteyden yli käyttäen LDAPS-protokollaa. Varsinaisesti toteutus ei eronnut alkuperäisestä suojaamattomasta muuten, kuin että suojattu yhteys pitää kytkeä IMS-asennuksen yhteydessä ja LDAPS-yhteydelle pitää avata eri portti kuin suojaamattomalle versiolle. IMS todentaa yhteyden käyttäen asiakkaalta saatua sertifikaattia. Jatkokehityksen suhteen olisi suotavaa, että synkronoinnin voisi kohdistaa ennalta määrittelemättömään määrään AD-ryhmiä. Kirjoitushetkellä käyttäjät synkronoidaan aina kahdesta ryhmästä IMS-ohjelmistoon kerran vuorokaudessa.

SAML tarjoaa lukuisia mahdollisuuksia sekä kirjautumisessa että käyttäjähallinnossa. SAML-viestien avulla uudet käyttäjät voitaisiin lisätä IMS-ohjelmistoon SAML-viestin välityksellä. Jos aiheen osoittamaa käyttäjää ei ole vielä olemassa, mutta hän kuuluu ryhmään, jolla on oikeudet käyttää IMS-ohjelmistoa, voitaisiin käyttäjä lisätä lennosta IMS:in.

7 Yhteenveto

Insinöörityön tavoitteena oli lisätä IMS-ohjelmistoon tuki eri toimialueiden väliselle kertakirjautumiselle käyttäen SAML 2.0 -kertakirjautumisprotokollaa. IMS:n olemassa olleet kertakirjautumisratkaisut olivat tarkoitettu käytettäväksi asiakkaan omassa sisäverkossa ja omalla toimialueella. Tarkoituksena oli mahdollistaa kertakirjautuminen IMS:n itse hallinnoimilla palvelimilla.

Työssä selostetaan yleisesti käyttäjähallinnosta ja kertakirjautumisesta ennen siirtymistä SAML-standardiin ja sen ominaisuuksiin. SAML-selainkertakirjautuminen toteutettiin käyttäen HTTP POST -sidoksia ja OpenSAML-kirjastoja. OpenSAML on osa Spring Security SAML -kirjastoa, joka on tarkoitus ottaa käyttöön IMS-ohjelmistossa vielä määrittelemättömässä vaiheessa.

Työ saavutti sille asetetut tavoitteet, sillä SAML-kertakirjautumisprotokollaa on käytetty jo yli vuosi usean eri asiakkaan IMS-asennuksissa. Lopputuloksena IMS-ohjelmisto kykenee kokoamaan, lähettämään ja allekirjoittamaan SAML-standardien mukaisia autentikointipyyntöjä. Lisäksi IMS pystyy vastaanottamaan ja varmentamaan henkilöllisyydentarjoajan lähettämiä SAML-autentikointivastauksia.

Lähteet

1 Single Sign-on, Verkkodokumentti, Wikipedia, <http://en.wikipedia.org/wiki/Single_sign-on>, Päivitetty 23.4.2014.

2 Hypertext Transfer Protocol --- HTTP/1.1, Verkkodokumentti, R. Fielding, UC Ir-civine, J Gettys, ym, Network Working Group, <http://tools.ietf.org/html/rfc2616>, Päivitetty 1999.

3 Maximum URL length is 2,083 characters in Internet Explorer, Verkkodokumentti, Microsoft tuotetuki, <http://support.microsoft.com/kb/208427>, Päivitetty 27.10.2007.

4. User Centric Identity Management, 2005, Audun Josang & Simon Pope, University of Queensland Australia, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.60.1563&rep=rep1&type=pdf>.

5. Identity Management, Verkkodokumentti, Wikipedia, <http://en.wikipedia.org/wiki/Identity_management >, Päivitetty 26.4.2014.

6. Digital Identity, Verkkodokumentti, Wikipedia, <http://en.wikipedia.org/wiki/Digital_identity>, Päivitetty 7.10.2103.

7. Oasis-Open: Security Assertion Markup Language (SAML) V2.0 Technical Overview, 2008, Hal Lockhart, Brian Campbell, Nick Ragouzis, John Hughes, ym, Verkkodokumentti, OASIS Security Services TC, <http://www.oasis-open.org/committees/download.php/27819/sstc-saml-tech-overview-2.0-cd-02.pdf>.

8 Red Hat Enterprise Linux 6, Managing Single Sign-On and Smart Cards, 2010, Ella Deon Lackey, Verkkodokumentti, <https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html-single/Managing_Smart_Cards/index.html#About_Kerberos>, Luettu 12.3.2014.

9. OpenID Authentication 2.0 Final, 2007, Verkkodokumentti, <http://openid.net/specs/openid-authentication-2_0.html>, Luettu 20.4.2014.

10 OpenSAML Usein kysytyt kysymykset, Scott Cantor, 2006, Verkkodokumentti, Shibboleth, <https://wiki.shibboleth.net/confluence/display/OpenSAML/OSFaq>, Päivitetty 9.5.2012, Luettu 20.4.2014.

11. The OAuth 2.0 Authorization Framework, D. Hardt, Ed, 2012, Verkkodokumentti, Internet Engineering Taskfor(IETF), <http://tools.ietf.org/html/rfc6749> Luettu 24.4.2014.

12. Security Assertion Markup Language, Verkkodokumentti, Wikipedia, <http://en.wikipedia.org/wiki/Security_Assertion_Markup_Language>, Päivitetty 7.4.2014.

13. SAML 2.0, Verkkodokumentti, Wikipedia, <http://en.wikipedia.org/wiki/SAML_2.0>, Päivitetty 5.4.2014.

14. OASIS_(organization), Verkkodokumentti, Wikipedia, <http://en.wikipedia.org/wiki/OASIS_(organization)>, Päivitetty 2.4.2014.

15. Integrated Windows Authentication, Verkkodokumetti, Wikipedia, <http://en.wikipedia.org/wiki/Integrated_Windows_Authentication>, Päivitetty 19.2.2014.

16. SAS(R) 9.2 Intelligence Platform: Security Administration Guide: Integrated Windows Authentication, 2014, Verkkodokumentti, SAS, <http://support.sas.com/documentation/cdl/en/bisecag/61133/HTML/default/viewer.htm#a003140629.htm>.