Identiteetit ja roolit identiteetinhallintajärjestelmissä

Niila Mäkelä

Tampereen yliopisto
Tietojenkäsittelytieteiden laitos
Tietojenkäsittelyoppi
Pro gradu -tutkielma
Ohjaaja: Pirkko Nykänen
Toukokuu 2008

Tampereen yliopisto
Tietojenkäsittelytieteiden laitos
Tietojenkäsittelyoppi
Niila Mäkelä: Identiteetit ja roolit identiteetinhallintajärjestelmissä
Pro gradu -tutkielma, 53 sivua
Toukokuu 2008

Tässä pro gradu -tutkielmassa läpikäydään identiteetinhallinnan ja rooliperustaisen pääsynhallinnan perusteet sekä perehdytään tarkemmin identiteettien yksilöiviin tunnisteisiin ja roolienhallintaan. Tutkielman tutkimusongelmat ovat: ”Miten roolien määrittelyä voitaisiin kehittää ad hoc -tyyppisestä toiminnasta kohti harkittua suunnitteluprosessia?” ja ”Miten sähköiset identiteetit tulisi erotella toisistaan, eli millaisia yksilöiviä tunnistetietoja identiteetteihin tulisi liittää?”. Tutkimusongelmia lähestytään kirjoittajan kokemuksen ja kirjallisuusanalyysin avulla. Roolien määrittelyä varten esitetään erilaisia lähestymistapoja, joiden avulla määrittelyprosessia voidaan helpottaa ja selkeyttää. Tutkielmassa pohditaan erilaisten identiteettien yksilöivien tunnistetietojen vahvuuksia ja heikkouksia. Lisäksi esitetään tiivis ohjerunko, jota erilaiset organisaatiot voivat hyödyntää identiteettien ja roolien määrittelyssä. Tutkielma on suunnattu organisaatioille, jotka suunnittelevat identiteetinhallintajärjestelmän käyttöönottoa.

Avainsanat ja -sanonnat: identiteetinhallinta, roolit, käyttöoikeudet, IAM, IM, IdM, LDAP, RBAC.

Sisällys

1. Johdanto

1.1. Tutkielman tausta

Organisaatiot tallentavat usein henkilötietoja työntekijöistään ja asiakkaistaan erilaisiin tietojärjestelmiin. Mitä enemmän organisaatiolla on tietojärjestelmiä käytössään, sitä todennäköisemmin henkilötiedot ovat hajallaan eri järjestelmissä. Näitä tietoja voitaisiin hyödyntää erilaisten sähköisten palveluiden tuottamisessa, mutta tiedon pirstoutuminen eri järjestelmiin vaikeuttaa tietojen käyttämistä. Tätä ongelmaa helpottamaan on kehitetty identiteetinhallintajärjestelmiä. Niiden avulla henkilöille voidaan luoda yksilölliset sähköiset identiteetit. Kun jokainen henkilötieto on liitetty yhteen sähköiseen identiteettiin, tietojen koostaminen yhteen eri järjestelmistä helpottuu huomattavasti.

Eräs keskeisimmistä identiteetinhallinnan käyttötarkoituksista on käyttöoikeushallinnan helpottaminen. Tietojärjestelmien käyttöoikeuksia voidaan kytkeä identiteetinhallintajärjestelmässä käsiteltäviin rooleihin. Tämän jälkeen identiteetteihin voidaan kytkeä rooleja. Näin ollen yksittäiselle käyttäjälle tehtävien käyttöoikeusmääritysten määrä vähenee huomattavasti.

Identiteetinhallintaa koskevassa kirjallisuudessa viitataan usein siihen, kuinka vaikeaa tai työlästä roolien määrittely identiteetinhallintajärjestelmiä varten on. [Roeckle et al., 2000; Park et al., 2004] Kirjoittaja on osallistunut kahteen eri identiteetinhallintajärjestelmän käyttöönottoprojektiin, ja kokemus näistä projekteista on vahvistanut edellä mainitun ongelman todelliseksi. Organisaation koosta ja rakenteesta riippuen roolien määrittely on todennäköisesti yksi työläimmistä vaiheista identiteetinhallintajärjestelmän käyttöönotossa. Tämän työvaiheen onnistuminen tai epäonnistuminen vaikuttaa hyvin paljon koko käyttöönottoprojektin onnistumiseen.

Organisaatiot tavoittelevat identiteetinhallintajärjestelmillä muun muassa henkilötietojen synkronoinnin helpottumista eri tietojärjestelmien välillä. Tämän toteuttaminen edellyttää, että eri tietojärjestelmiin tallennetut henkilötiedot ovat yhdistettävissä. Tällöin voidaan hyödyntää sähköisten identiteettien yksilöiviä tietoja. Tällaisten tietojen käyttäminen ei kuitenkaan ole täysin mutkatonta ja siitä syystä tässä pro gradu -tutkielmassa perehdytään myös identiteettien yksilöiviin tietoihin.

1.2. Tutkielman tavoitteet ja rajaus

Organisaatioilla ei välttämättä ole vahvaa osaamista roolienmäärittelytyöstä ennen organisaation ensimmäistä identiteetinhallintajärjestelmän käyttöönottoprojektia. Tämä tutkielma on pyritty kirjoittamaan siten, että se voisi toimia ohjeena roolien määrittelyssä organisaatioille, jotka suunnittelevat identiteetin-

hallintajärjestelmän käyttöönottoa. Tutkielman luettavuutta on pyritty parantamaan keskeisten lukujen aluissa esiintyvillä johdanto-osuuksilla.

Tutkielmassa käydään läpi identiteetinhallinnan ja rooliperustaisen pääsynhallinnan perusteet sekä perehdytään tarkemmin identiteettien yksilöiviin tunnisteisiin ja roolienhallintaan. On siis huomattava, että vaikka tutkielma toimii ohjeena, se ei ole kattava opas siitä, kuinka identiteetinhallintajärjestelmä tulisi käyttöönottaa. Identiteetinhallintajärjestelmät sisältävät usein sellaisia toiminnallisuuksia, kuten kertakirjautumistoiminnallisuudet, joita tässä tutkielmassa ei käsitellä.

Tutkielmassa keskitytään yhden organisaation identiteetinhallintaan. Näin ollen useamman organisaation välistä identiteetinhallintaa sivutaan ainoastaan muutamassa kohdassa. Toimintaympäristön laajentaminen koskemaan useampaa organisaatiota herättäisi uusia kysymyksiä ja ongelmia ratkaistavaksi. Lisäksi identiteetinhallinnan kannalta oleellista käyttäjien autentikointia (eli todentamista) käsitellään ainoastaan vähän ja hyvin abstraktilla tasolla, koska aiheesta on jo olemassa paljon kirjallisuutta.

1.3. Tutkimusongelmat ja -menetelmät

Tähän tutkielmaan on valittu kaksi tutkimusongelmaa:

- Miten roolien määrittelyä voitaisiin kehittää ad hoc -tyyppisestä toiminnasta kohti harkittua suunnitteluprosessia?
- Miten sähköiset identiteetit tulisi erotella toisistaan, eli millaisia yksilöiviä tunnistetietoja identiteetteihin tulisi liittää?

Ensin mainittu tutkimusongelma on valittu tarkasteltavaksi, koska roolien määrittelyprosessi on osoittautunut haastavaksi käytännön työssä. Myös muut ovat havainneet saman ongelman jo aiemmin [Roeckle et al., 2000]. Tähän tutkimusongelmaan pyritään löytämään vastauksia esittämällä erilaisia lähestymistapoja, joita voidaan hyödyntää roolien määrittelyssä. Tähän ongelmaan perehdytään luvussa 6 Roolienhallinta. Johdantona aiheeseen toimii luku 5 Roolit ja rooliperustainen pääsynhallinta.

Jälkimmäisen tutkimusongelman ratkaisemisen tärkeyttä ei voida korostaa liikaa. Mikäli organisaatio käyttää jokaisessa tietojärjestelmässään erilaisia tunnistetietoja käyttäjistä, järjestelmäintegraatioiden toteuttaminen vaikeutuu huomattavasti. Tätä ongelmaa lähdetään purkamaan esittelemällä ensin identiteetit luvussa 2 Identiteetit ja identiteetinhallintajärjestelmät. Tämän jälkeen luvussa 3 Identiteetit ja yksilöinti esitellään erilaisia vaihtoehtoja identiteettien yksilöiviksi tiedoiksi ja pohditaan kunkin vaihtoehdon vahvuuksia sekä heikkouksia.

Kumpaakin tutkimusongelmaa lähestytään kirjoittajan oman kokemuksen kautta sekä analysoimalla aiheeseen liittyvää kirjallisuutta. Tämä tutkimusmetodi valittiin, koska se tuntui kirjoittajasta luonnollisimmalta tavalta lähestyä tutkimusongelmia. Tiedon kerääminen identiteetinhallintajärjestelmiä käyttäviltä organisaatioilta olisi todennäköisesti ollut vaikeaa, koska identiteetinhallinta liittyy olennaisesti organisaatioiden tietoturvaratkaisuihin. Vastaavasti järjestelmätoimittajien haastattelu olisi todennäköisesti tuottanut enemmän tietoa siitä, kuinka kunkin toimittajan järjestelmällä ongelmat ratkaistaan kuin siitä, miten ne olisi parasta tai hyvä ratkaista.

2. Identiteetit ja identiteetinhallintajärjestelmät

2.1. Identiteetit

Käsite sähköisestä identiteetistä (*identity, digital identity*) ei ole täysin vakiintunut. Osassa lähteistä käsite on rajattu viittaamaan ainoastaan tietojärjestelmiä käyttäviin henkilöihin. *"An Identity is a user of an information system, performing activities such as querying a patient's electronic medical record."* [Longstaff et al., 2003, 126] Osassa lähteistä käsite kattaa laajemman joukon tietojärjestelmissä kuvattavia objekteja. Esimerkiksi Microsoftin MS Identity Integration Server 2003 -tuotteella on mahdollista käsitellä käyttäjien, käyttäjäryhmien, palveluiden ja laitteiden sähköisiä identiteettejä [Microsoft, 2004]. ATK-sanakirja [2004] ehdottaa termin *identity* suomennokseksi sanaa henkilöllisyys tai identiteetti. Tässä pro gradu -opinnäytetyössä käytetään näistä jälkimmäistä.

Tässä tutkielmassa identiteetit ymmärretään siten, että mille tahansa tietojärjestelmissä käsiteltävälle objektille, joka on yksikäsitteisesti tunnistettavissa ja yksilöitävissä, voidaan muodostaa oma sähköinen identiteettinsä. Näitä objekteja voivat olla muun muassa käyttäjät, palvelut, erilaiset resurssit (esimerkiksi laitteet ja yksilöllisesti nimetyt tuotteet), käyttäjäryhmät ja organisaatiot. Tutkielmaan valittiin käytettäväksi edellä mainittu tulkinta termistä seuraavista syistä johtuen:

- Hyvin toimivan pääsynhallintaratkaisun kehittäminen vaatii tarkastelun ulottamista muihinkin toimiviin objekteihin kuin käyttäjiin. Esimerkiksi itsenäisesti tekoälynsä varassa toimiva sovellus tarvitsee myös oikeat käyttöoikeudet selviytyäkseen tehtävistään ja noudattaakseen organisaation asettamia sääntöjä. Voi olla myös hyvin oleellista saako tietojärjestelmän palvelu A kutsua palvelua B. Tämän kaltaisen käyttöoikeusvalvonnan tarve tulee entisestään kasvamaan palveluperustaiseen arkkitehtuuriin (*Service-Oriented Architecture*, SOA) pohjautuvien järjestelmien yleistyessä.
- Sähköisiä tunnisteita käytetään nykyään hyvin monenlaisissa yhteyksissä. Esimerkiksi RFID-tunnisteita voidaan käyttää muun muassa tunnistamaan karjaeläimiä. Mikäli eläimen RFID-tunnistetta käytetään ruokinta-automaatin ohjaamiseen, eläin toimii tällöin tietojärjestelmän käyttäjänä, eikä käyttäjä olekaan näin ollen *henkilö*, vaikka käyttäjällä onkin oma sähköinen identiteettinsä.
- Termin laajempi tulkinta voi luoda uusia ideoita ja auttaa tekniikkaa leviämään uusille sovellusalueille.

Tässä tutkielmassa käytetään siis sähköisestä identiteetistä seuraavaa määritelmää [Cameron, 2005; Slone, 2004; Windley 2005]:

- Sähköinen identiteetti voidaan luoda mille tahansa tietojärjestelmässä olevalle yksikäsitteisesti yksilöitävissä olevalle objektille.

- Sähköiseen identiteettiin on kuuluttava edellisen mukaan vähintäänkin yksi sellainen tieto tai tietojoukko, jonka mukaan objekti voidaan tunnistaa. Objektille annetaan esimerkiksi sellainen nimi, jota ei ole millään toisella objektilla. Lisäksi aina, kun tällä nimellä viitataan johonkin objektiin, niin se viittaa täsmälleen yhteen ja samaan objektiin. Huom. tämä ei kuitenkaan sulje pois sitä mahdollisuutta, että identiteetin tunnistetieto voi muuttua.
- Identiteettiin voi liittyä yksilöivän tiedon (”nimen”) lisäksi myös muita kuvaavia ominaisuuksia. Esimerkiksi yrityksen työntekijän sähköiseen identiteettiin voi kuulua hänen yhteystietonsa.

Määritelmä ei siis rajaa identiteettiä tarkoittamaan pelkästään järjestelmää käyttävää henkilöä eli käyttäjää. Näin siitä huolimatta, että tässä tutkielmassa keskitytään pääasiassa käyttäjähallintaan tietojärjestelmissä.

2.2. Identiteetinhallintajärjestelmät

Identiteetinhallinnalla tarkoitetaan tässä tutkielmassa sähköisten identiteettien hallintaa sekä pääsynhallintaa. Englanninkielisessä tietojärjestelmätieteen kirjallisuudessa käytetään useita erilaisia termejä ja lyhenteitä kuvaamaan tätä toimintaa. Näistä termeistä yleisimpiä ovat *Identity Management* (lyhennettynä IM tai IdM) sekä *Identity and Access Management* (IAM). Etenkään termin Identity Management käyttö ei ole täysin vakiintunut [Shaw, 2007]. Pfitzmann ja Hansen [2008] määrittelevät termin henkilöiden identiteettien hallinnaksi. Osa lähteistä taas määrittelee termin tarkoittavan pääsynhallintaa. ”*Identity management (IDM) - Systems and processes that manage and control who has access to resources, and what each user is entitled to do with those resources, in compliance with the organisations' policies.*” [Leenes et al., 2007, 2] IAM on termeistä siinä mielessä selkeämpi, että se tuo lukijalle mielikuvan siitä, että asia koskee identiteetin*- ja* pääsynhallintaa. Tässä tutkielmassa käytetään Identity and Access Management -termistä käännöstä identiteetinhallinta.

Laajan tulkinnan mukaan identiteetinhallintajärjestelmät voivat koostua hyvin monenlaisista toiminnallisuuksista. Kahdella organisaatiolla saattaa olla käytössään hyvinkin erityyppiset järjestelmät silti kummankaan organisaation ei tarvitse olla väärässä kutsuessaan tietojärjestelmäänsä identiteetinhallintajärjestelmäksi. Laitetoimittajat tarjoavat erilaisia tuoteperheitä identiteetinhallintaan. Identiteetinhallintajärjestelmää käyttöönotettaessa valitaan usein toimittajan tarjoamista tuotteista joitakin ja sovitetaan ne jo aiemmin hankittujen tietojärjestelmien rinnalle organisaation tarpeiden mukaiseksi omaksi ratkaisukokonaisuudeksi. Slone [2004, 5] kuvaa hyvin syitä, miksi eri organisaatioiden identiteetinhallintajärjestelmät ovat erilaisia: ”*Identity Management (IdM) is a convergence of technologies and business processes. There is no single approach to identity*

management because the strategy must reflect specific requirements within the business and technology context of each organization."

Identiteetinhallintajärjestelmät voivat koostua mm. seuraavista toiminnallisuuksista tai komponenteista [Slone, 2004]:

- Käyttäjätunnusten hallinta, joka voi sisältää automatisoituja osioita, kuten tunnusten luomisen HR-järjestelmän (*Human Resources Management System*) tietojen perusteella.
- Käyttäjätunnushakemistot, jotka usein perustuvat LDAP-tekniikkaan (*Lightweight Directory Access Protocol*, [RFC 1777, 1995]). Yleisimmin käytetty tällainen tuote lienee MS Active Directory.
- Autentikointi eli käyttäjien todentaminen/tunnistaminen.
- Salasanojen hallinta.
- *Single Sign On* (SSO) eli kertakirjautuminen. Tätä ominaisuutta hyödyntävien tietojärjestelmä- tai palvelukokonaisuuksien käyttäjien ei tarvitse kirjautua kuin kerran istunnon aluksi yhteen järjestelmään/palveluun, jonka jälkeen hänen kirjautumisensa säilyy järjestelmästä/palvelusta toiseen siirryttäessä.
- Roolien ja niihin liittyvien käyttöoikeuksien hallinta. Tähän asiaan syvennytään luvussa 5 Roolit ja rooliperustainen pääsynhallinta.
- Auditointi ja raportointi. Auditointiominaisuuksia voidaan hyödyntää muun muassa silloin, kun pyritään selvittämään mahdollisia järjestelmän väärinkäyttötapauksia. Raportointi toimii apuna, kun halutaan tietoja esimerkiksi siitä, kuinka monella käyttäjällä on käyttöoikeudet tiettyyn järjestelmään.
- Itsepalvelutoiminnallisuudet, kuten käyttöoikeuksien anominen työnkulun (*workflow*) avulla ja käyttäjän itse suorittama salasanan uudelleenasettaminen. Työnkulkuja käsitellään luvussa 6 Roolienhallinta.

3. Identiteetit ja yksilöinti

Eräs identiteetinhallinnan keskeisistä perusteista on se, että käyttäjät pystytään yksilöimään yksikäsitteisesti [Slone, 2004]. Tätä tarkoitusta varten jokaiseen identiteettiin on kuuluttava tieto tai joukko tietoja, jotka yksilöivät identiteetin. Toisin sanoen tämä yksilöivä tieto viittaa aina pelkästään yhteen tiettyyn kohteeseen. Kun kyse on ihmisistä, helppo ja luonnollinen tapa on käyttää ihmisten erisnimiä tähän tarkoitukseen. Joukossa Anna Aalto, Kaisa Kallio ja Matti Majava pelkkä etunimi riittää erottelemaan henkilöt. Ongelma kuitenkin syntyy, kun joukkoon lisätään Matti Mainio. Tämän laajennetun joukon henkilöt pystytään erottelemaan, kun yksilöinnissä käytetään apuna joukkoa tietoja eli henkilöiden etu- ja sukunimeä yhdessä. Kun esimerkkijoukkoa kasvatetaan, huomataan varsin nopeasti, ettei etu- ja sukunimen yhdistelmäkään riitä erottelemaan henkilöitä toisistaan. Tähän ongelmaan palataan tarkemmin kohdassa 3.1 Henkilöiden identiteettien yksilöinti.

Identiteetti ei aina liity henkilöön, vaan se voi kuvata esimerkiksi tuotetta, organisaatiota tai tietojoukkoa. Fyysisten tuotteiden valmistuksen yhteydessä tuotteeseen liitetään usein sarjanumero, joka yksilöi tuotteen. [Slone, 2004] Sarjanumeroa voidaan hyödyntää esimerkiksi hajonneen taulutelevision korjaamisessa. Sarjanumerosta voidaan päätellä laitteen valmistushetki, joka voi auttaa vian määrittämisessä. Henkilöauton kriittisimmät osat voivat olla sarjanumeroituja sen lisäksi, että autolle annetaan oma yksilöivä rekisterinumeronsa. Näiden tietojen avulla olisi mahdollista selvittää, onko käytetyssä autossa edelleen auton alkuperäinen moottori. Maitotölkin kylkeen painetaan viivakoodi ja tölkin ylälaitaan tuote-erän numero. Tällöin yksittäistä maitotölkkiä ei erota toisesta samaan tuote-erään kuuluvasta tölkistä, mutta valmistusvirheellisen erän vetäminen pois kauppojen hyllyiltä on mahdollista.

Organisaatiot voidaan erottaa toisistaan useimmiten jo pelkän nimen perusteella. Tämän lisäksi on olemassa erilaisia rekistereitä erityyppisistä organisaatioista. Suomalaiset yritykset voidaan tunnistaa yksikäsitteisesti Y-tunnuksen avulla [Asetus 288, 2001]. Varsinkin suuremmat organisaatiot on usein jaettu useampaan sisäiseen alaorganisaatioon. Organisaatiot voivat nimetä alaorganisaationsa melko vapaasti. Yrityksissä A ja B voi kummassakin olla yksikkö, joka on nimetty markkinointiosastoksi. Tällöin näiden erottamiseen tarvitaan alaorganisaation nimen lisäksi myös jotakin tarkentavaa tietoa kuten pääorganisaatioiden Y-tunnuksia.

Tietojoukon yksilöimiseen on useita tapoja ja niitä voidaan soveltaa erilaisissa käyttötarkoituksissa. Yrityksellä voi olla käytössään suurikin joukko lomakkeita, joita työntekijät täyttävät esimerkiksi lomia tai matkakorvauksia hakiessaan. Lomakkeita voidaan nimen lisäksi merkitä koodilla tai lyhenteellä, jolla pyri-

tään helpottamaan ihmisten välistä kommunikaatiota ja varmistamaan, että kussakin tilanteessa käytetään juuri oikeata lomaketta. Tällainen lyhenne ei tietenkään yksilöi yksinään kahta täytettyä lomaketta, jotka on tehty samanlaisille lomakepohjille. Kukin täytetty lomake voidaan kuitenkin yksilöidä hyväksikäyttämällä tietoja siitä, kuka lomakkeen on täyttänyt ja allekirjoittanut, milloin lomake on palautettu ja kuka lomakkeen on käsitellyt.

Erilaiset tietojoukot esitetään nykyään yhä enenevissä määrin sähköisessä muodossa tiedostoina tai tietokantoina. Tämä luo uusia mahdollisuuksia ja haasteita tietojen yksilöintiin. Yksittäisestä tiedostosta tai merkkijonosta on mahdollista laskea erilaisten algoritmien avulla tiivisteitä, jotka yksilöivät tiedoston. Mikäli tiedostoa muokataan, siitä laskettava tiivistekin muuttuu. Yksittäisestä tiedostosta voidaan kuitenkin ottaa identtinen kopio. Tällöin tiedoston yksilöivänä tietona voidaan pitää muun muassa tiedoston tallennuspaikkaa.

3.1. Henkilöiden identiteettien yksilöinti

Tämän luvun alussa todettiin, etteivät henkilöiden erisnimet riitä erottamaan näiden sähköisiä identiteettejä yksikäsitteisesti toisistaan. Seuraavissa alakohdissa lähdetään purkamaan tätä ongelmaa eri ratkaisuilla ja pohditaan, mitä etuja ja haittoja näihin ratkaisuihin liittyy.

3.1.1. Organisaation sisäiset tunnistenumerot

Eräs tapa lähteä ratkaisemaan henkilöiden yksilöintiongelmaa on antaa kullekin henkilöille organisaation sisäinen oma tunnistenumeronsa (vrt. tuotteiden sarjanumerot). Yrityksissä tällainen käytäntö onkin yleinen ja työntekijöille annetaan usein työntekijänumero. Kun uusi henkilö palkataan yritykseen, hänelle luodaan oma työntekijänumeronsa. Tämä numero voi olla yrityksen HR-järjestelmään tallennettu juokseva numero. Vastaavasti myös organisaatioiden asiakkaille jaetaan usein asiakasnumeroita. Seuraavaksi käydään läpi tähän organisaatioiden sisäisiin tunnistenumeroihin liittyviä vahvuuksia ja heikkouksia.

Ratkaisumallin vahvuudet:

- Mikäli tunnistenumerot annetaan uusille asiakkaille ja työntekijöille sillä periaatteella, että tunnistenumero on aina suurempi kuin edelliset, voidaan olla varmoja, ettei kahdella asiakkaalla tai työntekijällä ole koskaan samaa numeroa käytössään.
- Ratkaisu on usein nopeasti käyttöönotettavissa, etenkin siinä vaiheessa, kun organisaatio aloittaa toimintansa.

Ratkaisumallin heikkoudet:

- Valmiina pakettiratkaisuina hankittavat tietojärjestelmät ja ohjelmistot sisältävät usein omat tunnistenumeronsa henkilöille. Nämä numerot voivat näkyä suoraan järjestelmän käyttöliittymästä tai ne voivat olla piilossa käyttäjiltä tietokannan tunnistenumeroina. Organisaatiossa päätetyn tunnistenumeron vieminen näihin järjestelmiin voi olla vaikeaa tai jopa mahdotonta. Jos eri järjestelmät käyttävät omia tunnistenumeroitaan, jossakin on ylläpidettävä tietoa siitä, mitkä tunnistenumerot eri järjestelmissä vastaavat toisiaan. Mikäli tätä työtä ei ole aina aloitettu tekemään heti uuden järjestelmän hankinnan tai rekisterin perustamisen yhteydessä, tällaisen vastaavuustiedon kerääminen voi olla hyvin työlästä ja pahimmassa tapauksessa jopa mahdotonta.
- Vahvuudeksi mainittiin, ettei kahdella eri henkilöllä ole koskaan samaa tunnistenumeroa. Tämä ei kuitenkaan riitä yksinään takaamaan yksikäsitteistä identiteettien erottelua. *Lisäksi vaaditaan, ettei yhdelläkään henkilöllä ole kahta eri tunnistenumeroa samasta tunnistenumerosarjasta.* Tämä voi muodostua ongelmaksi, jos asiakkaille tai työntekijöille myönnetään useampia tunnistenumeroita. Työntekijöille ei tule myöntää uutta tunnistenumeroa jokaisen työsuhteen perusteella, vaan ainoastaan silloin, kun työntekijän ensimmäistä työsuhdetta perustetaan. Määräaikaisessa työsuhteessa olevien henkilöiden tunnistenumeroiden tulee säilyä alkuperäisinä, vaikka kahden työsuhteen välissä olisikin taukoa. Sama koskee myös asiakasnumeroita. Asiakkaan asiakasnumeron on säilyttävä samana asiakassuhteen mahdollisista katkoista huolimatta. Vaikka asiakas käyttäisi yrityksen eri yksiköiden palveluita, asiakas tulisi kirjata järjestelmiin aina samalla asiakasnumerolla. On myös varauduttava siihen, että asiakas unohtaa asiakasnumeronsa tai kadottaa kanta-asiakaskorttinsa, johon asiakasnumero on painettu.
- Koska tunnistenumero on organisaation sisäinen, sen hyödyntäminen organisaatioiden välisessä viestinnässä on haasteellista.
- Henkilö saattaa toimia yhdessä tilanteessa yrityksen työntekijänä ja toisessa asiakkaana. Yrityksen olisi hyvä varautua tällaisen tilanteen ratkaisemiseen jo etukäteen.
- Asiakkaiden yksilöinti juoksevalla numerolla voi paljastaa ulkopuolisille tahoille salaiseksi määriteltyjä tietoja organisaation asiakasmääristä.

3.1.2. Biometriset tunnisteet

Henkilöiden yksilöivänä tunnisteena voidaan käyttää myös henkilön omia fyysisiä ominaisuuksia. Biometrisistä tunnisteista yleisimpiä lienevät sormenjälkitunnisteet. Muina biometrisina tunnisteina voidaan käyttää muun muassa silmän iiristä ja verkkokalvoa sekä käden verisuonia. Sormenjälkitunnisteita käytettäessä käyttäjän sormenjälki ensin kuvataan ja kuvan pohjalta muodostetaan

sitten malli, joka tallennetaan tietojärjestelmään. Käyttäjän kirjautuessa tietojärjestelmään, sormi kuvataan, muodostetaan uusi malli kuvan perusteella ja kuvaa verrataan aiemmin tallennettuun malliin. Mikäli mallit muistuttavat riittävästi toisiaan, niiden katsotaan kuuluvan samalle henkilölle. [Ailisto et al., 2005] Tällaisissakin järjestelmissä identiteetillä on usein oma erillinen tunnistenumeronsa ja sormenjäljistä muodostetut mallit toimivat tämän identiteetin kuvaavina ominaisuuksina. Tämä ei kuitenkaan tarkoita sitä, ettei biometrinen tunniste voisi toimia identiteetin yksilöivänä tunnisteena tai muiden tunnistetietojen rinnalla osana tunnistetta.

Ratkaisumallin vahvuudet:

- Tunniste on henkilötunnisteena luonnollisemman oloinen kuin keksitty numerosarja.
- Käyttäjän ei tarvitse osata lukea eikä kirjoittaa voidakseen tunnistautua.

Ratkaisumallin heikkoudet:

- Laiteriippuvuus. Eri laitevalmistajien lukijat eivät välttämättä ole yhteensopivia keskenään.
- Henkilöt kasvavat, lihovat, laihtuvat ja vanhenevat. Toisin sanoen käyttäjien fyysiset ominaisuudet muuttuvat ajan kuluessa. Tämän vuoksi käyttäjien biometrisia tunnisteita voidaan joutua uusimaan.
- Biometristen tunnisteiden käyttäminen aiheuttaa usein kasvavia kuluja muun muassa laitehankintojen kautta.
- Biometristen tunnistetietojen tallentaminen eri tietojärjestelmiin voi olla haasteellista. Tästä syystä käyttäjille joudutaan todennäköisesti antamaan jokin numeerinen tunniste.

3.1.3. Kansalliset tunnistenumerot

Eri maissa kansalaiset pyritään rekisteröimään muun muassa veronmaksun vuoksi. Samassa yhteydessä kansalaiselle annetaan usein oma tunnistenumeronsa. Myös näitä tunnistenumeroita voidaan käyttää identiteettien yksilöivinä tunnisteina tietyin rajoittein. Myönnettyjen tunnistenumeroiden muodot sekä tunnistenumeroiden myöntämiseen liittyvät käytänteet vaihtelevat maittain. Esimerkiksi Isossa-Britanniassa on käytössä National Insurance Number [DWP, 2006].

Väestörekisterikeskuksen myöntämä suomalainen tunnistenumero tunnetaan nimellä henkilötunnus (hetu). Tunnus myönnetään jokaiselle Suomen kansalaiselle. Tässä kohdassa käydään vielä lävitse yleisellä tasolla kansallisten tunnistenumeroiden käyttämiseen identiteetinhallinnassa liittyvät edut ja haitat. Seuraavassa kohdassa 3.1.4 Henkilötunnus perehdytään tarkemmin suomalaiseen henkilötunnukseen.

Ratkaisumallin vahvuudet:

- Kansalliset tunnistenumerot ovat jo valmiiksi laajassa käytössä kyseessä olevassa maassa.
- Oman organisaation ulkopuolinen taho pitää huolen siitä, ettei kahdella henkilöllä ole samaa tunnistenumeroa ja että kullakin henkilöllä on ainoastaan yksi henkilötunnus.
- Kukin henkilö tuntee jo valmiiksi oman tunnistenumeronsa.

Ratkaisumallin heikkoudet:

- Tunniste on kansallinen, ei kansainvälinen. Maassa vierailevilla henkilöillä ei ole tätä tunnistetta. Myöskään tuoreilla maahanmuuttajilla ei välttämättä ole tunnistenumeroa tai heille on saatettu antaa väliaikainen tunnistenumero, joka muuttuu vielä myöhemmin toiseksi, pysyväksi tunnistenumeroksi.
- Paikallinen lainsäädäntö saattaa rajoittaa tunnisteen käyttöä.

3.1.4. Henkilötunnus

Suomessa on käytössä Väestörekisterikeskuksen Suomen kansalaisille myöntämä henkilötunnus. Tunnusta käytetään laajalti yritysten ja julkisen sektorin järjestämien palveluiden yhteydessä. Henkilötunnus perustuu tietoihin henkilön syntymäpäivästä ja sukupuolesta. Henkilötunnuksella on Suomessa jo melko pitkä historia. Se on ollut käytössä vuodesta 1970 asti [Asetus 198, 1970]. Tätä ennen oli jo kuitenkin käytössä vuodesta 1963 alkaen sosiaaliturvatunnus, joka sittemmin korvattiin henkilötunnuksella [Asetus 473, 1963].

Henkilötunnus on muotoa DDMMYYerotinmerkkiXXXY, jossa:

- DD on kuukaudenpäivä.
- MM on kuukausi.
- YY on vuosi.
- Erotinmerkki on plus- (1800), miinus- (1900), tai A-merkki (2000) henkilön syntymävuosisadan mukaan.
- XXX on kolminumeroinen sarja, joka erottelee samana päivänä syntyneet. Parillinen luku tarkoittaa naista ja pariton luku miestä.
- Y on tarkistusmerkki. Tarkistusmerkki saadaan jakamalla syntymäajan ja yksilönumeron muodostama yhdeksännumeroinen luku 31:llä, jolloin tarkistusmerkki määräytyy jakojäännöksen mukaan. Tarkistusmerkit löytyvät taulukosta 1. [Väestörekisterikeskus]

Jakojäännös	Tarkistusmerkki	Jakojäännös	Tarkistusmerkki
0	0	16	H
1	1	17	J
2	2	18	K
3	3	19	L
4	4	20	M
5	5	21	N
6	6	22	P
7	7	23	R
8	8	24	S
9	9	25	T
10	A	26	U
11	B	27	V
12	C	28	W
13	D	29	X
14	E	30	Y
15	F		

Taulukko 1: Henkilötunnuksen tarkistusmerkit

Ratkaisumallin vahvuudet:

- Henkilötunnus on miltei jokaisella suomalaisella.
- Henkilötunnus on yksikäsitteinen, koska samana päivänä syntyneet erotetaan toisistaan tunnuksen loppuosalla.
- Useimmat suomalaiset tietävät oman henkilötunnuksensa.
- Henkilötunnuksesta käy ilmi henkilön ikä ja sukupuoli.

Ratkaisumallin heikkoudet:

- Henkilötunnus ei ole julkista tietoa. Se ei saa näkyä muille järjestelmän käyttäjille.
- Henkilötunnuksen käyttöä säädellään laeilla. Organisaation on tunnettava lainsäädännön vaatimukset. Katso henkilötietolaki [L523, 1999].
- Tunnusta ei ole ulkomaisilla henkilöillä.
- Maahanmuuttajat eivät saa henkilötunnusta välittömästi. Tästä syystä heidät voidaan joutua tallentamaan tietojärjestelmiin väliaikaisilla henkilötunnuksilla. Mikäli tietojärjestelmää ei ole suunniteltu siten, että yksilöivä tunniste voi muuttua, väliaikaisten henkilötunnusten korvaaminen oikeilla voi aiheuttaa ongelmia.
- Tunnus on ainoastaan henkilöillä. Mikäli organisaatioille halutaan luoda omat tunnisteensa, ne eivät voi noudattaa samaa muotoa kuin henkilötunnukset.

3.1.5. ISO OID-yksilöintitunnukset

Y-tunnuksen lisäksi organisaatioiden on mahdollista hakea ISO OID-yksilöintitunnusta (*Object Identifier*). Suomessa näiden tunnusten myöntämisestä vastaa Suomen Standardisoimisliitto SFS ry. OID-yksilöintitunnuksia on mahdollista määritellä myös muun muassa dokumenteille ja henkilöille. [JHS 159, 2006] Tulee huomata, että organisaatio voi itsenäisesti ottaa käyttöön juurensa alaisia OID-tunnuksia.

OID-yksilöintitunnukset muodostuvat numeroista ja piste-merkeistä. Esimerkiksi 1.16.840.1.113883.6 on muodoltaan OID-tunnus. Suomeen rekisteröidyt tunnukset alkavat Suomen juuresta, joka on 1.2.246. Tämän jälkeiset luvut voivat saada arvoja yhden ja 16777215 väliltä. [JHS 159, 2006] OID-tunnuksia hyödynnetään muun muassa terveydenhuollon kansallisessa tietojärjestelmäarkkitehtuurissa [Alkula, 2007].

Ratkaisumallin vahvuudet:

- OID-yksilöintitunnus perustuu kansainväliseen ISO-standardiin.
- Samanmuotoisia tunnuksia voidaan antaa monen tyyppisille objekteille. OID-yksilöintitunnus voidaan myöntää esimerkiksi organisaatiolle ja organisaatiossa työskenteleville henkilölle.
- Ainakin Suomessa valvotaan, ettei kertaalleen myönnettyä OID-tunnusta anneta myöhemmin toiselle kohteelle [JHS 159, 2006].
- OID-tunnusavaruus on laaja, joten tunnusten ei pitäisi loppua heti kesken. (Vertaa Internet-protokollat: IPv4:ssä on 2^{32} ja IPv6:ssa on 2^{128} mahdollista IP-osoitetta [RFC 791, 1981; RFC 2460, 1998].)

Ratkaisumallin heikkoudet:

- Standardi ei aseta tiukkoja ehtoja siitä, kuinka muun muassa henkilötiedot kuuluu esittää.
- Suomessa julkishallintoa ohjeistetaan yhdenmukaiseen toimintamalliin [JHS 159, 2006]. Tämäkään ei kuitenkaan takaa sitä, että tiedot tallennettaisiin saman mallin mukaan eri organisaatioissa.
- Alempien tasojen objektit ovat sidoksissa ylempiin tasoihin. Tämä on heikkous esimerkiksi sellaisessa tilanteessa, jossa henkilö siirtyy organisaatiosta tai organisaatioyksiköstä toiseen.
- Organisaation tarvitsee itse huolehtia, ettei mitään organisaatiojuuren alaista OID-tunnusta käytetä viittaamaan kahteen eri objektiin.

3.2. Muut identiteetteihin liittyvät tiedot

Kohdassa 3.1 ja sen alakohdissa käsiteltiin identiteettien yksilöiviä tietoja. Näiden lisäksi sähköisiin identiteetteihin liittyy usein suuri joukko muita tietoja. Nämä tiedot voivat kuvata identiteetin ominaisuuksia tarkemmin. Tällaisia

tietoja voivat olla muun muassa nimi, käyttäjätunnus, yhteystiedot, valokuva ja roolit. Seuraavissa alakohdissa käsitellään tärkeimpiä näistä tiedoista. Roolien keskeistä osasta johtuen niihin perehdytään erikseen luvussa 5 Roolit ja rooliperustainen pääsynhallinta.

3.2.1. Nimitiedot

Identiteetin yksilöivän tiedon lisäksi identiteettiin liittyy usein myös muita nimitietoja. Nämä tiedot eivät välttämättä yksilöi identiteettiä, mutta niistä voi olla apua monessa eri käyttötarkoituksessa.

Tietojärjestelmissä hyödynnetään ja esitetään käyttäjien nimiä useissa eri tilanteissa. Se, miten nimi esitetään, vaihtelee usein käyttötarpeen mukaan. Käyttäjän koko nimeä käytetään usein tilanteissa, joissa henkilöiden sekaantumista halutaan erityisesti välttää ja ehkä korostaa asian virallisuutta. Tällainen tilanne voi esimerkiksi olla se, kun työntekijälle lähetetään palkkakuitti. Joskus käyttäjien nimet halutaan esittää aakkosjärjestyksessä. Tällöin henkilöiden nimet on järkevää esittää siten, että sukunimi on mainittuna ennen muita nimiä.

Seuraavaksi käydään läpi ehdotuksia nimiattribuuteista, jotka kannattaa liittää osaksi henkilöiden sähköisiä identiteettejä. Attribuuttien perässä on selitykset ja yksinkertaiset esimerkit.

- *Koko nimi*: Henkilön virallinen nimi kokonaisuudessaan (Juhani Antero Kolehmainen).
- *Kutsumanimi*: Henkilöä puhuteltaessa käytetty etunimi tai muu kutsumanimi (Jussi).
- *Sukunimi*: Henkilön sukunimi kokonaisuudessaan (Kolehmainen).

Identiteetinhallintajärjestelmän toteutuksesta riippuen voi olla hyvä tallentaa myös seuraavat attribuutit, vaikka ne voitaisiinkin koostaa edellisten perusteella.

- *Kutsumanimi ja sukunimi*: Edelliset kaksi yhdessä (Jussi Kolehmainen).
- *Sukunimi ja kutsumanimi:* Huomioi esitysjärjestys. (Kolehmainen Jussi).

Organisaation tarpeista riippuen voi olla hyödyllistä tallentaa myös muita yhdistelmiä, kuten:

- *Etunimi/kutsumanimi, Toisen etunimen ensimmäinen kirjain ja sukunimi* (Juhani A. Kolehmainen tai Jussi A. Kolehmainen).

3.2.2. Käyttäjätunnus

Sähköiseen identiteettiin on usein sidottu käyttäjätunnus tai useampia. Vaikka identiteetinhallintaratkaisu ei tarjoaisi todellista kertakirjautumispalvelua, voidaan IAM-järjestelmiä usein hyödyntää muutoin helpottamaan kirjautumista eri järjestelmiin. Nykyään useat ohjelmistot ja tietojärjestelmät tarjoavat käyttäjän tunnistukseen LDAP-tukea. Tämä tarkoittaa sitä, että käyttäjät voidaan tunnistaa jostakin ulkoisesta LDAP-hakemistosta. Mikäli IAM-ratkaisuun kuuluu LDAP-hakemisto, sitä voidaan hyödyntää tähän tarkoitukseen. Tämä mahdollistaa käyttäjien kirjautumisen useampaan eri järjestelmään samalla käyttäjätunnuksella ja salasanalla.

Seuraavaksi joitakin ehdotuksia siitä, millainen käyttäjätunnuksen ja siihen liittyvän salasanan olisi hyvä olla.

- Käyttäjätunnuksen tulee olla henkilökohtainen, jotta käyttäjä saa juuri oikeat käyttöoikeudet järjestelmään. Samoin käyttäjien suorittamien toimien auditointi vaikeutuu tai on mahdotonta, mikäli samaa tunnusta käyttää useampi henkilö. Tämä vaatimus tuntuu itsestään selvältä. Tästä huolimatta tätä sääntöä rikotaan useissa organisaatioissa erilaisissa tilanteissa.
- Käyttäjätunnuksen tulisi pysyä muuttumattomana, vaikka sen muuttaminen onkin mahdollista. Näin etenkin ympäristöissä, joissa useampi tietojärjestelmä käyttää samoja käyttäjätunnuksia. Tällaisessa ympäristössä toimittaessa tunnuksen muuttaminen aiheuttaa usein ylimääräistä ylläpitotyötä.
- Käyttäjätunnuksen olisi hyvä olla lyhyt. Useassa vanhassa ja joissakin uusissakin tietojärjestelmässä on asetettu käyttäjätunnuksen maksimipituudeksi kahdeksan merkkiä (merkit a-z ja 0-9). Siksi tunnuksen olisi hyvä olla tämän mittainen, etenkin mikäli organisaatiolla on käytössään vanhoja tietojärjestelmiä, jotka halutaan saada toimimaan samalla käyttäjätunnuksella kuin muutkin palvelut.
- Käyttäjätunnuksen sekä salasanan tulisi olla helposti muistettavia, jotta käyttäjät eivät unohtaisi niitä.
- Salasana ei saa kuitenkaan olla helposti arvattavissa oleva. Sen tulee olla riittävän pitkä ja sisältää sekä kirjainmerkkejä että numeroita. Salasanan ei tulisi myöskään sisältää kokonaisia sanoja.
- Järjestelmän tulee pakottaa käyttäjät vaihtamaan salasanaansa tietyin aikavälein. Sopiva vaihtotiheys riippuu siitä, kuinka usein käyttäjät kirjautuvat järjestelmään. Esimerkiksi yksi tai kaksi pakotettua salasanan vaihtokertaa vuodessa voi olla sopiva määrä.
- Tunnuksen tulee lukittua, jos väärällä salasanalla yritetään kirjautua useasti peräkkäin. Tällä pyritään estämään tunnuksen väärinkäyttö tilanteissa, joissa murtautuja yrittää joko arvata salasanan tai kokeilee oh-

jelmallisesti suurta joukkoa mahdollisia salasanoja (ns. *brute force* -hyökkäys).
- Käyttäjätunnuksen ei tulisi muodostua käyttäjän sukunimestä eikä organisaatioyksiköstä. Näillä tiedot voivat toisinaan muuttua ja tämä voisi aiheuttaa paineita myös tunnuksen muuttamiseen.

3.2.3. Yhteystiedot

Sähköisiin identiteetteihin on hyvä liittää käyttäjien yhteystiedot, koska niitä voidaan hyödyntää monessa erilaisessa sähköisessä palvelussa. Käyttäjähakemiston varaan on helppoa toteuttaa erilaisia yhteystietohakemistoja. Useassa kaupallisessa identiteetinhallintasovelluksessa onkin mukana tähän käyttöön sopiva sovellus. Tarvittavat yhteystiedot määräytyvät organisaation tarpeiden mukaan. Yhteystiedot voivat koostua muun muassa seuraavista tiedoista:

- Organisaation myöntämä sähköpostiosoite. Tähän osoitteeseen voidaan ohjata muun muassa automaattisesti lähetettävät viestit eri palveluista.
- Jokaiselle puhelinnumerolle (gsm, lankapuhelin, organisaation sisäinen lyhytvalintanumero, …) on hyvä varata oma tietokenttänsä. On hyvä varautua myös siihen, että yhdellä henkilöllä voi olla esimerkiksi useita gsm-numeroita.
- Osoitetiedot kannattaa tallentaa rakenteisina. Näin ollen tiedoista on helppoa näyttää ohjelmallisesti aina tilanteeseen soveltuvat.
- Mahdollisten pikaviestimien ja videoneuvottelulaitteistojen yhteystiedot.

3.3. Henkilötietojen käsittelyyn liittyvät velvollisuudet

Aiemmin todettiin, että henkilötunnuksen käsittelystä säädetään henkilötietolaissa. Sama laki ottaa kantaa myös muidenkin henkilötietojen käsittelyyn. Identiteetinhallinnassa on erityisesti huomioitava myös se, että identiteetinhallintajärjestelmä muodostaa usein yhden tai useamman henkilörekisterin, josta säädetään henkilötietolaissa. Tällöin rekisterinpitäjän tulee laatia rekisteriseloste, joka on pidettävä jokaisen saatavilla [L523, 1999]. Rekisteriselosteen pakolliset tiedot käyvät ilmi henkilötietolaista. Tämän lisäksi tietosuojavaltuutetun toimisto on laatinut rekisteriselostetta varten henkilötietolain mukaisia mallipohjia, joita on hyvä käyttää rekisteriselosteen mallina [Tietosuojavaltuutetun toimisto].

4. Autentikointi ja identiteetit

Useiden sähköisten palveluiden käytön kannalta on hyvin oleellista tunnistaa käyttäjä siksi toimijaksi, joka hän väittää olevansa. Henkilön varatessa kirjoja kirjaston verkkopalvelusta, hänen tulee pystyä todistamaan oma henkilöllisyytensä. Näin kirjaston järjestelmä tietää, kenelle kirja pitää varata ja ilkivallan tekeminen vaikeutuu, kun kirjoja ei voi varata muiden ihmisten nimissä. Tästä tunnistautumisprosessista, jossa käyttäjä todentaa henkilöllisyytensä, käytetään englanniksi nimitystä *(user) authentication*. Bishop [2003, 309] käyttää autentikoinnista usein siteerattua määritelmää: "*Authentication is the binding of an identity to a subject*."

Käyttäjän autentikoituessa tietojärjestelmään, hänet voidaan liittää oikeaan sähköiseen identiteettiin. Tässä tunnistautumisprosessissa voidaan hyödyntää yhtä tai useampaa seuraavista keinoista [Järvinen, 2006; Windley, 2005]:

- Käyttäjä esittää jotakin, mitä hän ainoastaan tietää (esim. salasanan).
- Käyttäjä esittää jotakin, mikä on ainoastaan hänen hallinnassaan (esim. avaimen tai kulkukortin).
- Käyttäjä esittää jotakin, mitä hän ainoastaan on (biometriset tunnisteet, esim. sormenjäljet)
- Hyödyntämällä useampaa kuin yhtä edellisistä keinoista, tunnistamisen varmuus kasvaa. Tällaisista ratkaisuista käytetään usein nimitystä *vahva tunnistautuminen*.

Perinteisesti tietojärjestelmiin on päässyt kirjautumaan tunnus–salasana-parilla. Organisaation on kuitenkin hyvä pohtia vahvan tunnistautumisen käyttöönottoa ainakin niissä järjestelmissä, joissa käsitellään kaikkein tärkeimpiä ja salaisimpia tietoja. Tässä tutkielmassa ei perehdytä tarkemmin erilaisiin tunnistautumistekniikoihin, koska aiheesta löytyy varsin runsaasti erilaista kirjallisuutta.

5. Roolit ja rooliperustainen pääsynhallinta

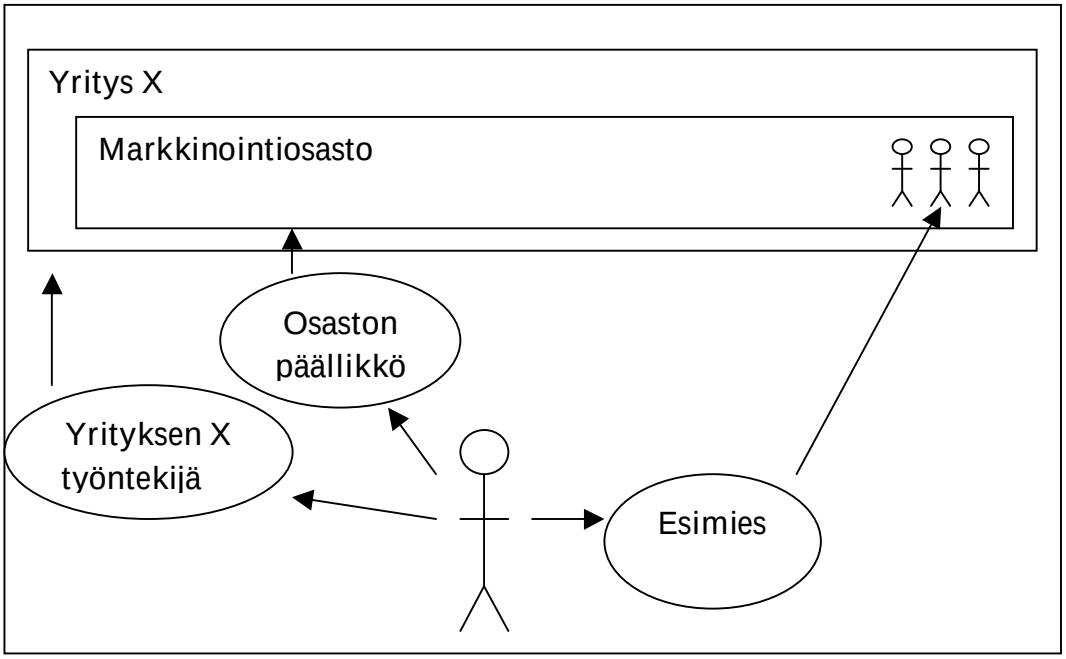
Tämä luku pohjautuu pääosin artikkeliin Role-Based Access Control Models [Sandhu et al., 1996]. Artikkelissa kuvataan hyvin ja melko yksityiskohtaisella tasolla rooliperustaisen pääsynhallinnan perusperiaatteet. Tässä luvussa käydään lävitse mainitussa artikkelissa esitetty RBAC-malli ja tutustutaan RBAC-mallin komposiittilaajennukseen [Park et al., 2004]. Edellisten lisäksi tutustutaan kahteen eri rooliluokittelumalliin ja johdetaan näiden perusteella uusi edelliset yhdistävä malli.

5.1. Rooliperustaisen pääsynhallinnan perusteet

Tietojärjestelmien käyttöoikeuksista puhuttaessa rooleilla tarkoitetaan organisaatiossa määriteltyä työnkuvaa, jossa määritellään roolin mukaisissa tehtävissä toimivan henkilön (käyttö)oikeudet ja velvollisuudet. On kuitenkin huomattava, että henkilön kompetenssi eroaa näistä. Se, mitä henkilö pystyisi tekemään, on eri asia kuin mitä henkilön on hyväksyttävää tehdä tai mitä hänen odotetaan tekevän. Tämä on eräs syy siihen, miksi pääsynhallintaa tarvitaan. [Sandhu et al., 1996]

Henkilöllä voi olla useita eri rooleja organisaation sisällä. Kahdessa seuraavassa esimerkissä valotetaan roolien merkitystä tarkemmin.

Esimerkki 1: Työntekijään liittyvät roolit
Tässä esimerkissä yrityksen X eräs työntekijä esiintyy kolmessa eri roolissa: yrityksen työntekijänä, osaston päällikkönä sekä alaistensa esimiehenä. (Katso kuva 1.)



Kuva 1: Työntekijään liittyvät roolit

Yhdellä henkilöllä voi siis olla useita erilaisia rooleja organisaation sisällä. Esimerkki 1 on rajattu hyvin suppeaksi ja todellisuudessa työntekijällä olisi rooleja todennäköisesti huomattavasti suurempi määrä. Huomattavaa esimerkissä on myös se, että roolit vaihtelevat näkökulman mukaan. Yrityksen X näkökulmasta esimerkin työntekijä voi olla ensisijaisesti yrityksen X työntekijä ja toissijaisesti Markkinointiosaston päällikkö. Vastaavasti Markkinointiosasto näkee työntekijän ensisijaisesti osaston päällikön roolissa ja toissijaisesti markkinointiosastolla työskentelevien henkilöiden esimiehenä. Työntekijän alaiset taas näkevät työntekijän ehkä nimenomaan esimiehenään. Vastaavasti tietojärjestelmienkin sisällä käyttäjä on voitu määritellä kuuluvaksi useaan eri rooliin. Kuitenkin näistä rooleista on yleensä aktiivisina vain osa kerrallaan. Aktiivisina olevat roolit vaihtelevat tilanteen ja tarpeen mukaan. Sandhu ja muut [1996, 41] määrittelevät roolit kuvaavasti seuraavalla tavalla: ”*A role is a named job function within the organization that describes the authority and responsibility conferred on a member of the role.*”

Esimerkki 2: Rooli
Organisaatiossa määritellään rooli Ohjelmistokehittäjä. Tähän rooliin kuuluva käyttäjä saa käyttöönsä pääsyn organisaation työasemille, intranetiin, sähköpostiin, ohjelmistonkehityssovellukseen sekä kulkuoikeudet ohjelmistonkehitysyksikköön.

Rooliperustaisen pääsynhallinnan (RBAC, Role-Based Access Control) keskeisimpiä elementtejä ovat roolit, kuten itse termi jo antaa olettaakin. IAM-järjestelmät ovat rooliperustaisia pääsynhallintajärjestelmiä. Tällaisissa järjes-

telmissä määritellään roolit organisaation työnkuvien mukaan, myönnetään rooleille tarvittavat oikeudet (*permissions*) ja kiinnitetään käyttäjät oikeisiin rooleihin. Näin käyttäjät saavat työtehtäviensä mukaiset oikeudet eri järjestelmiin.

5.2. Rooliperustaiset pääsynhallintamallit

Sandhun ja muiden artikkelissa [1996] esitetään neljä mallia, jotka kuvaavat rooliperustaisen pääsynhallinnan eri tasoja. Ensimmäinen perustaso ($RBAC_0$) kuvaa perusedellytykset ja sitä laajentavat mallit hierarkkiset roolit ($RBAC_1$), rajoitemalli ($RBAC_2$) sekä edelliset yhteen kokoava yhdistetty malli ($RBAC_3$).

5.2.1. Perustaso (*Base model*, $RBAC_0$)

$RBAC_0$ koostuu neljästä perusryhmästä, jotka järjestelmän tulee toteuttaa mahdollistaakseen rooliperustaisen pääsynhallinnan perustason. Nämä perusryhmät ovat: käyttäjät (*users*), roolit (*roles*), oikeudet (*permissions*), istunnot (*sessions*). Kaksi ensimmäistä määritelmää ovat jo tuttuja aiemmista osista tätä tutkielmaa. [Sandhu et al., 1996]

Oikeudella (*permission*) tarkoitetaan pääsyoikeutta (esim. kirjoitusoikeus) yhteen tai useampaan tiettyyn objektiin järjestelmässä. Ne ovat luonteeltaan aina positiivisia, eli ne tuovat jonkin oikeuden lisää järjestelmässä, mutta eivät lisää rajoitteita. Oikeudet liittyvät rooliperustaisessa pääsynhallinnassa yksinomaan rooleihin. Yhteen rooliin voi liittyä useita oikeuksia ja yksittäinen oikeus voi liittyä useaan rooliin. Oikeuksien ja roolien sitomisesta käytetään tässä tutkielmassa termiä oikeuttaminen (*Permission assignment*, PA). Oikeuksiin viitataan eri lähteissä myös termeillä *access rights*, *authorization* ja *privilage*. [Sandhu et al., 1996]

Istunto (*session*) kuvaa ne käyttäjän roolit, jotka ovat kyseisellä ajanjaksolla aktiivisina. Jokainen istunto sitoo *yhden* käyttäjän yhteen tai useampaan aktiiviseen rooliin. Tästä käytetään englanninkielistä termiä *User assignment*, (UA) ja tässä tutkielmassa suomennosta roolittaminen. [Sandhu et al., 1996] Aktiivisena olevat roolit voivat muuttua kesken istunnon.

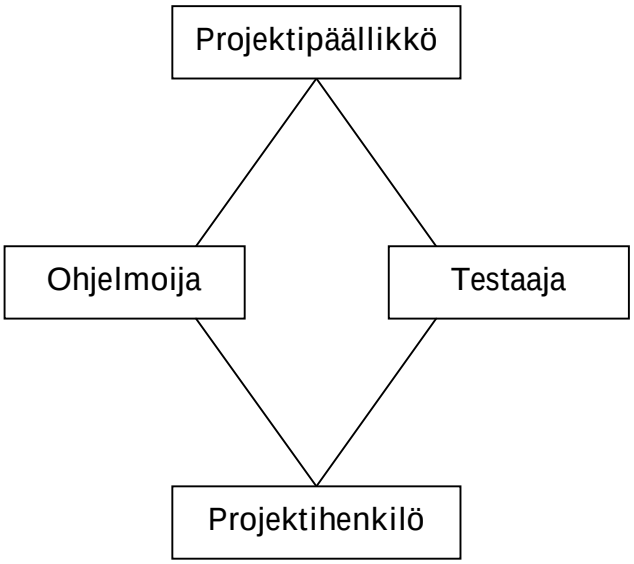
5.2.2. Hierarkkiset roolit (*Role hierarchies*, $RBAC_1$)

Perustasolla $RBAC_0$ roolit ovat toisistaan riippumattomia. Jokaiseen rooliin voi kuulua useita käyttäjiä ja jokainen käyttäjä voi kuulua useaan rooliin. $RBAC_1$ laajentaa $RBAC_0$:ia lisäämällä rooleille mahdollisuuden periä toistensa ominaisuudet.

Esimerkki 3: Roolien periytyminen
Esimerkissä 2 kuvattiin rooli Ohjelmistokehittäjä. Laajennetaan tätä esimerkkiä luomalla uusi rooli Projektipäällikkö. Rooliin kuuluva käyttäjä tarvitsee pääsyn organisaation työasemille, intranetiin, sähköpostiin, ohjelmistonkehityssovellukseen, projektinhallintatyökaluun sekä kulkuoikeudet ohjelmistonkehitysyksikköön. Roolin oikeudet määritellään lähes samoiksi kuin Ohjelmistokehittäjäroolinkin. Koska oikeusjoukko ainoastaan kasvaa (ja tässä tapauksessa vain yhdellä oikeudella), voi hyvinkin olla järkevää määritellä Projektipäällikkörooli siten, että se perii Ohjelmistokehittäjä-roolin oikeudet ja näitä oikeuksia laajennetaan ainoastaan oikeudella käyttää projektinhallintatyökalua.

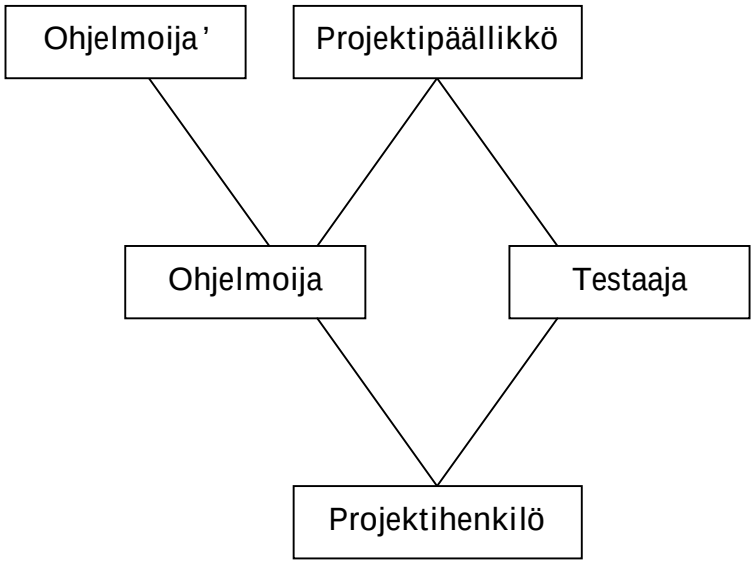
Joissakin tapauksissa kaikkien oikeuksien periytyminen ei ole toivottavaa, vaan periytymistä halutaan rajoittaa. Tarkastellaan tätä seuraavassa esimerkissä.

Esimerkki 4: Yksityinen rooli, mukaillen Sandhun ja muiden [1996] esimerkkiä
Ohjelmistoprojekteja toteuttavassa yrityksessä on määritelty roolit Projektipäällikkö, Testaaja, Ohjelmoija, Projektihenkilö. Kuvassa 2 esitetään roolien perimissuhteet.



Kuva 2: Roolien periytyminen

Yrityksessä todetaan, ettei projektipäällikön ole tarkoituksenmukaista päästä katsomaan tiettyjä osia ohjelmoijien keskeneräistä töistä työrauhan takaamiseksi. Tästä syystä oikeuksien periytymistä muutettiin luomalla uusi Ohjelmoija'-rooli, jonka jälkeen pystyttiin määrittelemään ohjelmoijille oikeuksia, jotka eivät periydy projektipäällikölle. Tämä uudistettu periytymismallin on esitetty kuvassa 3.



Kuva 3: Yksityinen rooli

Kuvan 3 Ohjelmoija'-roolia kutsutaan yksityiseksi rooliksi (*private role*). Tällaisilla rooleilla voidaan estää haluttujen oikeuksien periytyminen ylemmille tasoille.

5.2.3. Rajoitemalli (*Constraints model*, $RBAC_2$)

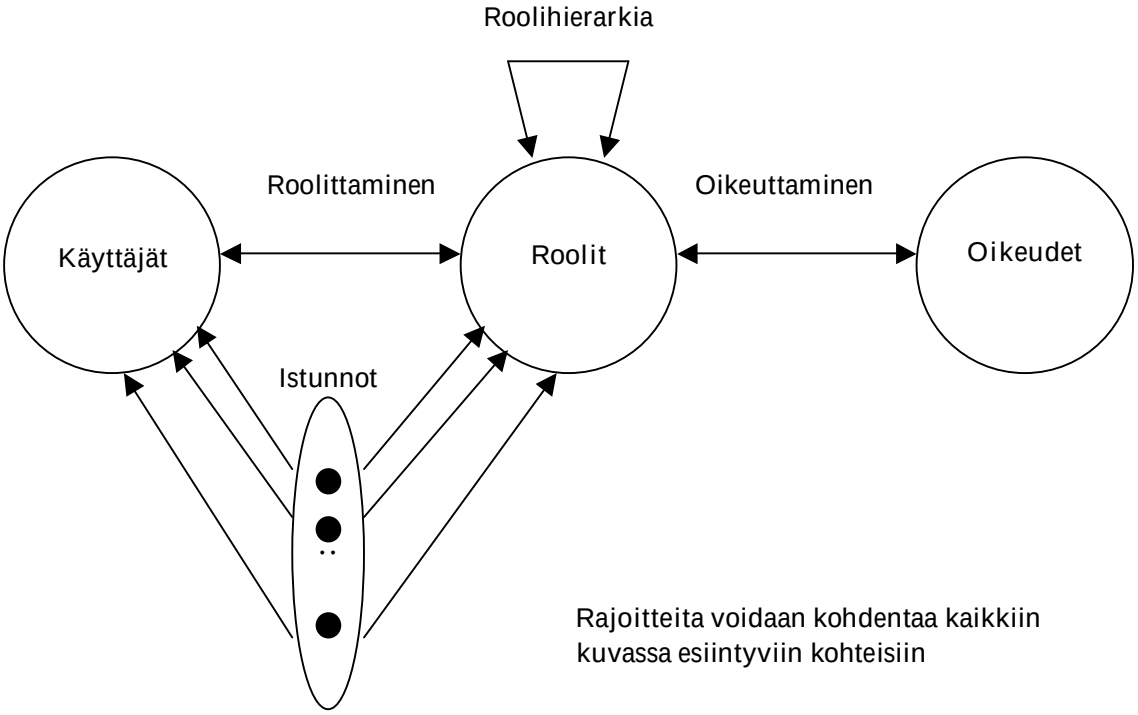
Kohdassa 5.2.1 kerrottiin oikeuksista, jotka ovat luonteeltaan positiivisia, eli oikeuden saaminen kasvattaa aina käyttäjän pääsyoikeuksia. Usein on kuitenkin tarvetta asettaa sääntöjä, jotka rajoittavat oikeuksien saamista. Tällöin on kyse rajoitteista (*constraints*). Tällainen tilanne voi olla yrityksessä, jossa tehdään tilauksia ja tehdyt tilaukset tarkastetaan. Tällöin on luonnollista, ettei sama henkilö voi toimia sekä tilaajana että hyväksyjänä. Tällaisesta vastuita rajaavasta rajoitteesta käytetään englanniksi nimitystä *separation of duties*, johon viitataan usein pääsynhallintaa koskevassa kirjallisuudessa. Käyttöoikeuksista huolehtivan järjestelmän täytyykin tarkistaa, ettei käyttäjän roolittamisessa myönnetä käyttäjälle kiellettyjä rooliyhdistelmiä. Tällaisia rajoitteita voidaan tehdä mm. luomalla roolijoukkoja, joista ainoastaan yksi rooli voi olla kerrallaan roolitettuna käyttäjälle (*mutually exclusive roles*). Roolittamista voidaan rajoittaa myös muillakin säännöillä. Kardinaliteettisäännöllä (*cardinality*) voidaan asettaa lukumäärärajoitteita siihen, kuinka monta käyttäjää voi enintään olla roolitettuna samaan rooliin samanaikaisesti. Rooleille voidaan asettaa myös edellytysvaatimuksia, niin että käyttäjän on täytettävä tietyt ehdot, jotta hänet voidaan roolittaa tiettyyn rooliin. Eräs tällainen edellytys voi olla käyttäjän kuuluminen tiettyyn roolin. Kuvan 3 Projektihenkilö -rooli voisi olla tällainen edellytys Projektipäällikkö -roolille. [Sandhu et al., 1996]

Rajoitteet voivat kohdentua myös roolihierarkiaan, jolloin tiettyjen oikeuksien periytyminen roolilta toiselle on kiellettyä. Tähän esitettiin ratkaisu edellisessä kohdassa ja kuvassa 3. Myös istunnoille voidaan asettaa rajoitteita muun muassa siten, että käyttäjä saa olla liitettynä useampaan tiettyyn rooliin, mutta yhden istunnon aikana aktiivisina saa olla ainoastaan tietyt roolit.

Organisaation tulee määritellä käytettävät rajoitteet ja niitä tulee soveltaa roolien määrittelyssä. Määrittelyn pohjana voidaan käyttää tässä ja kohdassa 5.3.1 esitettyjä rajoitteita. Rajoitteista voidaan tarvittaessa muodostaa monimutkaisiakin sääntökokonaisuuksia valitun identiteetinhallintajärjestelmän ominaisuuksista riippuen.

5.2.4. Yhdistetty malli (*Consolidated model*, $RBAC_3$)

Yhdistetty malli kerää edellä esitetyt pääsynhallintamallit ($RBAC_0$, $RBAC_1$ ja $RBAC_2$) yhteen. Tämä kokonaisuus on esitettynä kuvassa 4, joka on tehty Sandhun ja muiden [1996] artikkelissa esitetyn kuvan pohjalta.



Kuva 4: RBAC-malli

Nykyaikaisten pääsynhallintajärjestelmien tulisi täyttää vähintään edellä kuvattujen pääsynhallintamallien vaatimukset. RBAC on saanut vuoden 1996 jälkeen useampienkin kirjoittajien, muun muassa Sandhun ja Munawerin [1999] luomia laajennuksia. Näistä laajennuksista käsitellään tässä tutkielmassa aino-

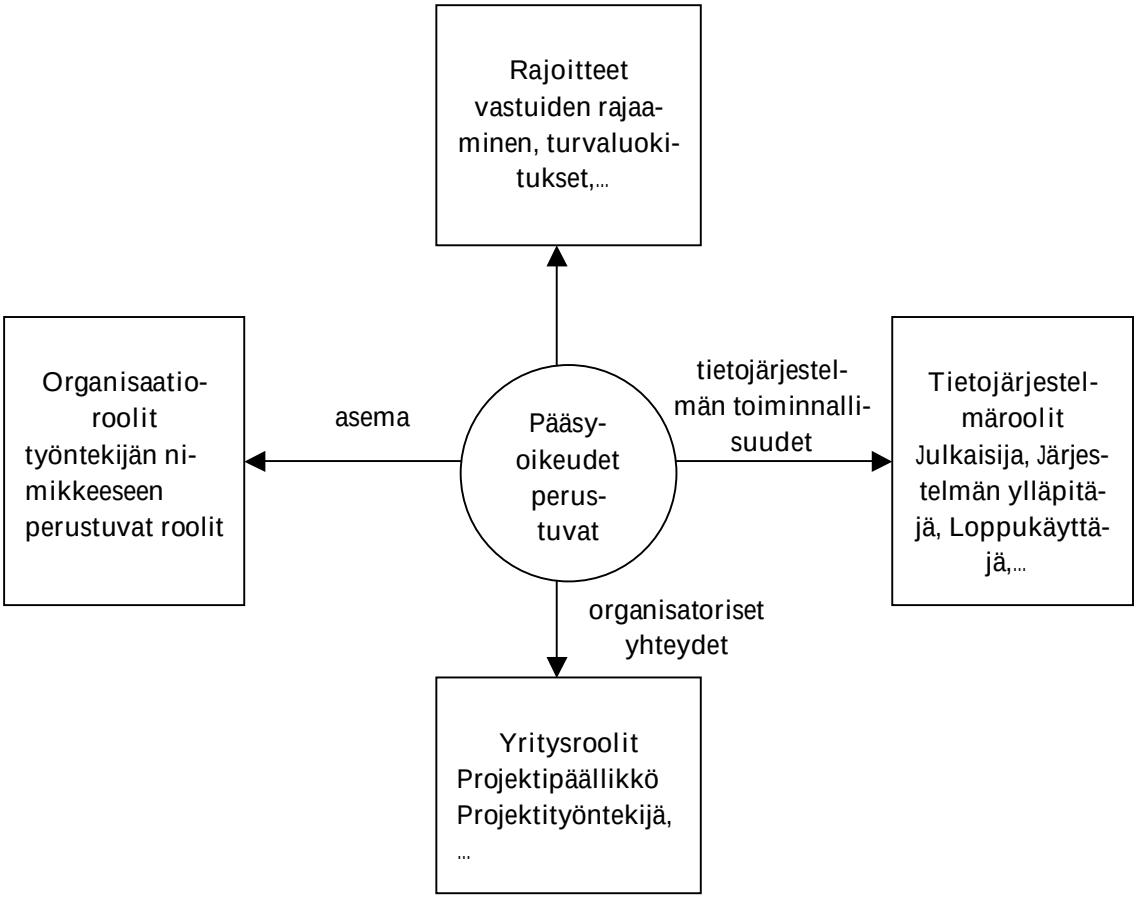
astaan yhtä [Park et al., 2004] kohdassa 5.3 Komposiittimalli. Pääsynhallintajärjestelmää suunniteltaessa tai kehitettäessä kannattaa tutustua myös muihin RBAC-mallin laajennuksiin.

5.3. Komposiittimalli

Laajoissa tietojärjestelmissä voidaan tarvita satoja tai jopa tuhansia rooleja. Vastaavasti organisaation koon kasvaessa erilaisten tehtävänkuvien ja niiden välisten riippuvuuksien kompleksisuus kasvaa. Näistä syistä roolienhallinta tulee sitä haastavammaksi, mitä laajemmasta organisaatiosta ja tietojärjestelmästä (tai tietojärjestelmäkokonaisuudesta) on kyse. Suurien organisaatioiden roolienhallinnan helpottamiseksi on kehitetty komposiittimalli *(Composite RBAC approach*) [Park et al., 2004]. Komposiittimallin perusideana on eriyttää organisatoriset ja tietojärjestelmiin liittyvät roolirakenteet ja tarjota toimiva linkitys näiden välille.

5.3.1. Roolien luokittelu komposiittimallissa

Park ja muut [2004] ovat esittäneet roolien jaottelun kolmeen eri rooliluokkaan: Organisaatioroolit, Yritysroolit ja Tietojärjestelmäroolit. Kuvassa 5 esitetään tämä luokittelu.



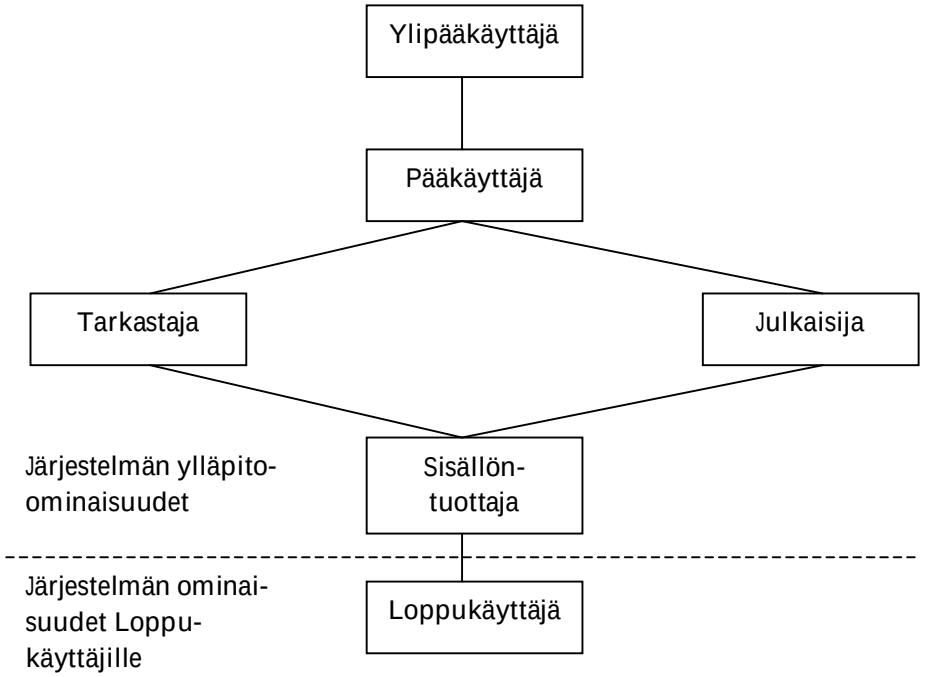
Kuva 5: Roolien luokittelu Parkin ja muiden [2004] mukaan

Kuvassa 5 ylimpänä esitetyistä rajoitteista on jo kerrottu kohdassa 5.2.3. Edellä mainittujen rajoitteiden lisäksi keskeinen rajoite on usein sovellettu vähäisimpien oikeuksien periaate (*least privilege*). Tällä tarkoitetaan sitä, että kullekin käyttäjälle myönnetään vain ne käyttöoikeudet, joita hän työssään todella tarvitsee. Park ja muut [2004] mainitsevat yhdeksi rajoitteeksi myös tiedon turvaluokituksen (*classification*). Turvaluokitusta voidaan käyttää joko lainsäädännön velvoitteesta tai organisaation omasta tahdosta. Tällainen luokitus voi jakaa tiedot esim. luottamuksellisiin ja salaisiin tietoihin.

Organisaatioroolit perustuvat henkilön asemaan organisaatiossa. Organisaatiorooleja voivat olla muun muassa Toimitusjohtaja ja Osastopäällikkö. Tällaiset roolit ovat helposti poimittavissa organisaatiokaavioista. Nykyisin työskentely organisaatioissa on kuitenkin projektiluonteista. Yhteen projektiin voidaan koota työntekijöitä niin organisaation eri yksiköistä kuin myös ulkopuolisista organisaatioista (esimerkiksi toimittajat, ulkopuoliset konsultit ja alihankkijat). Tällöin projektihenkilöstö muodostaa tavallaan oman virtuaalisen organisaationsa, joka toimii perinteisistä organisaatiorajoista riippumatta. Tätä tarkoitusta varten on mahdollista käyttää yritysrooleja [Park et al., 2004].

Yritysroolit (*enterprise roles*) ovat läheistä sukua organisaatiorooleille, jotka kuvaavat tiettyyn organisaatioyksikköön kuuluvan henkilön asemaa. Tämä yksinkertainen malli ei kuitenkaan yksinään riitä useimpien organisaatioiden tarpeisiin, sillä organisaatioissa tehdään usein organisaatioyksiköiden välistä yhteistyötä. Yleinen esimerkki tällaisesta toiminnasta ovat projektit, joihin osallistuu henkilöstöä useasta eri organisaatioyksiköstä. Yritysroolien tärkeys korostuu etenkin silloin, kun projektihenkilöstö on valittu rinnakkaisista yksiköistä. Tällöin hierarkiaan perustuvat organisaatioroolit eivät enää välttämättä päde projektin sisäisessä hierarkiassa ja tarvitaan yritysrooleja. Näitä rooleja voivat olla muun muassa Projektipäällikkö ja Projektihenkilö.

Tietojärjestelmärooleja tarvitaan tietojärjestelmien tarjoamien toiminnallisuuksien synnyttämien tarpeiden vuoksi. Park ja muut [2004] esittivät artikkelissaan kuvassa 6 kuvatun jaottelun tietojärjestelmärooleiksi.



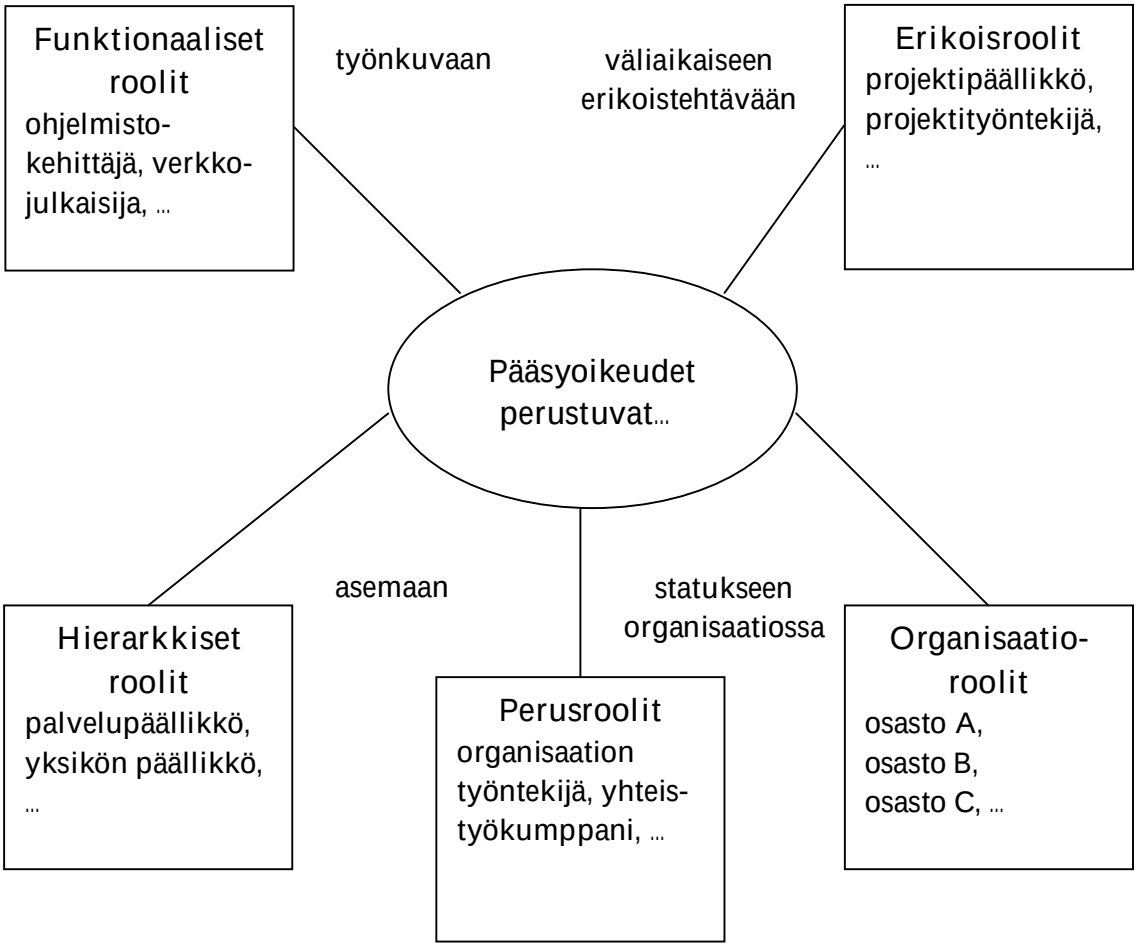
Kuva 6: Tietojärjestelmäroolit Parkin ja muiden [2004] mukaan

Kuvan 6 roolijaossa Ylipääkäyttäjä (*God*) on kaikkein voimakkain ja siihen kuuluvat kaikki järjestelmän käyttöoikeudet. Ylipääkäyttäjän tarkoitus on määritellä kaikki järjestelmässä tarvittavat roolit. Pääkäyttäjän (*System Administrator*) oikeudet ovat samat kuin Ylipääkäyttäjälläkin, mutta tähän rooliin kuuluvat käyttäjät eivät voi muokata järjestelmän rooleihin kuuluvia oikeuksia. Julkaisija (*Publisher*) julkaisee Sisällöntuottajien (*Authors*) tuottamat sisällöt, kun Tarkastaja (*Content Examiner*) on ensin tarkastanut sisältöjen oikeellisuuden. Edellä

mainitut roolit hyödyntävät järjestelmän ylläpito-ominaisuuksia (*Back-End*). Loppukäyttäjä (*End User*) pääsee käyttämään ainoastaan järjestelmän edusta-ominaisuuksia (*Front-End*), jotka ovat tarjolla peruskäyttäjille. [Park et al., 2004] Tämä malli on luotu suurelle yhdysvaltalaiselle Deparment of Homeland Secrurity -turvallisuusorganisaatiolle. Tämä näkyy mallissa myös hierarkkisuutena ja kohonneena byrokratia-asteena. Yhä useammin tietojärjestelmistä pyritään suunnittelemaan sellaisia, että loppukäyttäjät osallistuvat aktiivisesti sisällön tuottamiseen. Useissa järjestelmissä ennakkotarkastamista tehdään vain vähän ja pääosa julkaistusta materiaalista tarkistetaan vasta julkaisemisen jälkeen. Lisäksi tietojärjestelmien ylläpitotehtävät ja niihin liittyvät oikeudet vaihtelevat tietojärjestelmittäin. Näistä syistä johtuen esitetty tietojärjestelmärooli-jako ei todennäköisesti sovellu useimpien organisaatioiden käyttöön sellaisenaan, vaan tietojärjestelmärooleja tarvitaan useita ja niistä syntyy helposti tietojärjestelmäkohtaisia.

5.3.2. Komposiittimallin rooliluokkien arviointi ja uuden rooliluokittelumallin esittely

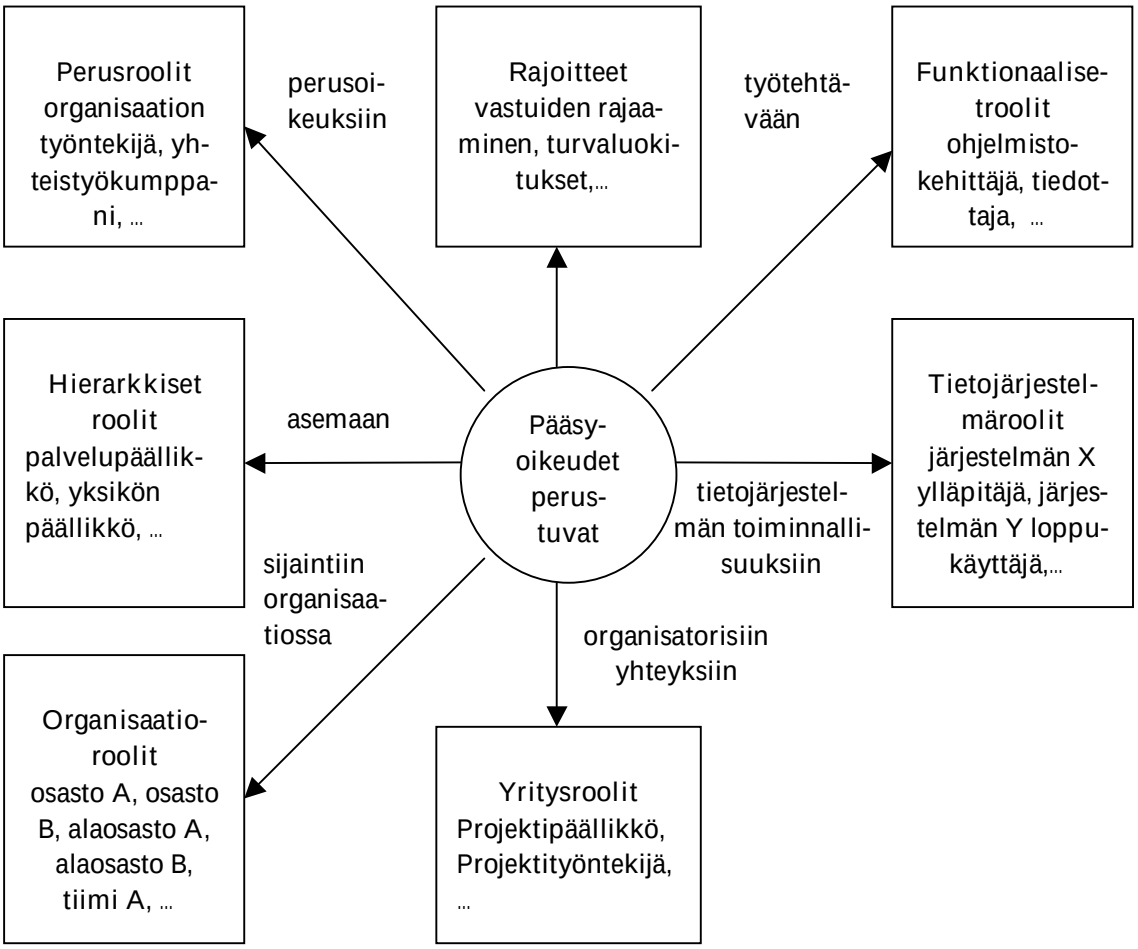
Kohdassa 5.3.1 tarkasteltiin Parkin ja muiden [2004] esittämää roolien jaottelua kolmeen eri rooliluokkaan (katso kuva 5). Roecklen ja muiden [2000] artikkelissa esitetään edellisestä hieman poikkeava jaottelu, joka on kuvassa 7. Siinä roolit on jaettu viiteen eri luokkaan: funktionaalisiin, erikois-, organisaatio- ja perusrooleihin sekä hierarkkisiin rooleihin.



Kuva 7: Roolien luokittelu Roecklen ja muiden [2000] mukaan

Kummassakin jaottelumallissa on lähdetty liikkeelle siitä, mihin pääsyoikeuksien tulisi perustua. Seuraavaksi perehdytään mallien eroihin. Roecklen ja muiden [2000] mallissa keskitytään pelkästään rooleihin, kun taas Park ja muut [2004] huomioivat myös rajoitteet. Myös itse rooliluokissa on selviä eroja. Roecklen ja muiden [2000] mallissa on hyödyllinen perusroolien luokka, jonka rooleilla käyttäjille voidaan antaa yleisimmät käyttöoikeudet. Tällaisia oikeuksia voivat olla mm. oikeus käyttää organisaation työasemia ja sähköpostia sekä selata intranetiä. Kummassakin mallissa on organisaatioroolien luokka, mutta samoista nimistä huolimatta ne tarkoittavat eri asioita. Parkin ja muiden [2004] mallin organisaatioroolit vastaavat pitkälti Roecklen ja muiden [2000] mallin hierarkkisia ja osittain myös funktionaalisia rooleja. Parkin ja muiden mallista taas puuttuvat Roecklen ja muiden [2000] esittämät organisaatioroolit. Mallien erikois- ja yritysroolit ovat hyvin läheistä sukua toisilleen. Erikoisrooleissa tosin korostetaan käyttäjän ja roolin välisen suhteen väliaikaisuutta. Vaikka edellisten havaintojen valossa Roecklen ja muiden [2000] malli vaikuttaa laajemmalta, siitä kuitenkin puuttuu olennainen tietojärjestelmäroolien luokka. Koska

kummassakin mallissa on omat vahvuutensa, esitetään kuvassa 8 edellisten mallien pohjalta luotu uusi, yhdistetty malli roolien luokittelusta.



Kuva 8: Yhdistetty roolien luokittelumalli

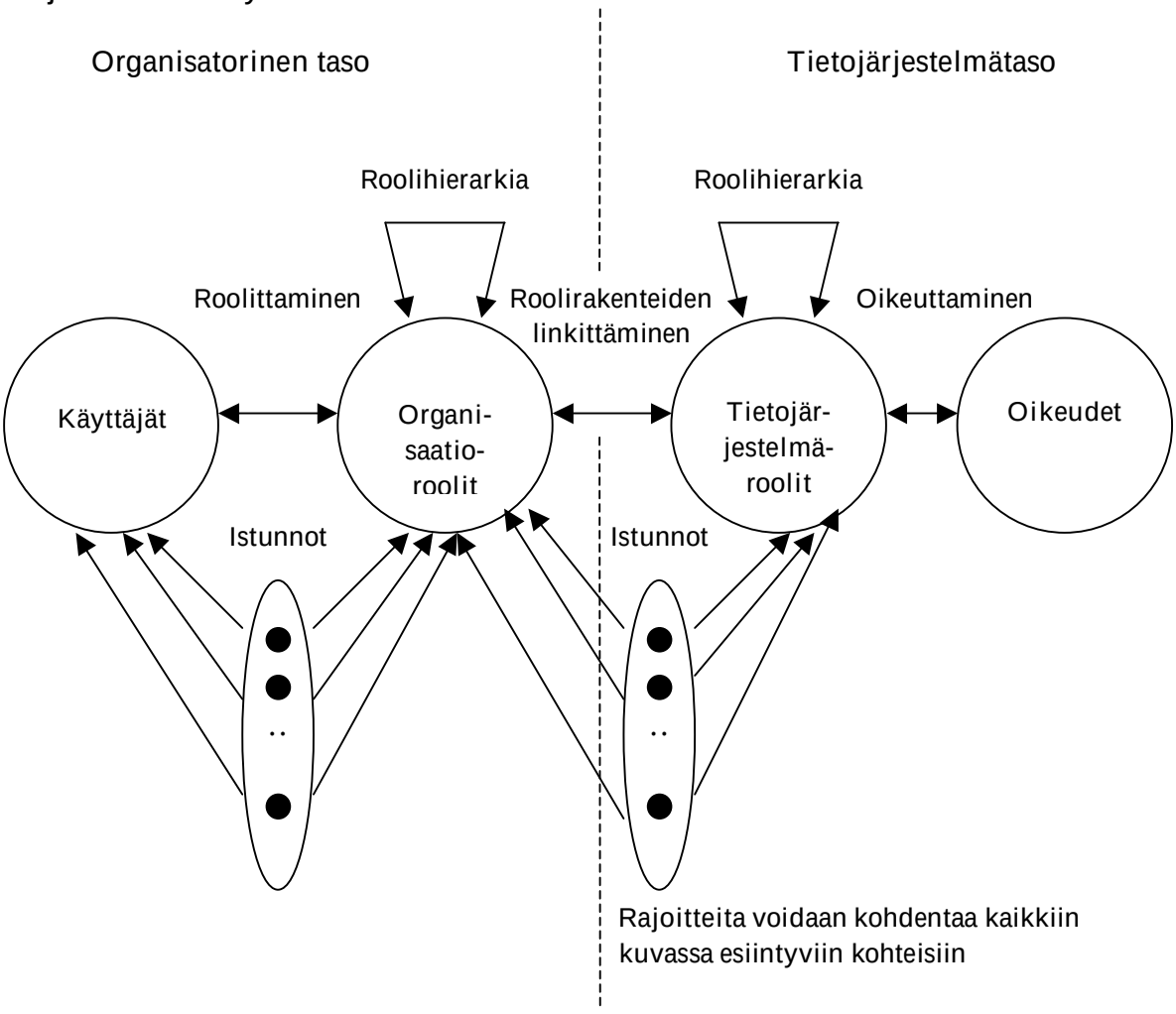
Kuvassa 8 esitetty roolien luokittelumalli yhdistää edellä kuvattujen mallien vahvuudet. Lisäksi siihen on tehty pieniä muutoksia verrattuna edellisiin malleihin. Organisaatiorooleihin on lisätty alaosastojen ja tiimien käsitteet. Tiimit ovat organisaatio-osaston sisäisiä pienempiä työntekijäryhmiä. Tiimit voisivat sijaita myös luokassa yritysroolit, jolloin tiimien jäsenet voisivat olla eri organisaatioyksiköistä. Samoin tietojärjestelmäroolien luonnetta on muutettu. Tässä ehdotetun mallin mukaan tietojärjestelmäroolit muodostettaisiin järjestelmäkohtaisiksi. Tietojärjestelmäroolien kytkemistä muihin rooleihin käsitellään kohdassa 5.3.3.

Kuvan 8 mallia voidaan soveltaa organisaation tarpeita vastaaviksi ja hyödyntää rooleja suunniteltaessa. Rooliluokittelumallia käytettäessä tulee huomioida, että jopa eri luokissa olevien roolien välille syntyy helposti erilaisia riippuvuussuhteita.

5.3.3. Organisaatio- ja tietojärjestelmäroolien yhdistäminen komposiittimallissa

Tässä kohdassa käydään lävitse komposiittimallin keskeisin laajennus perinteiseen RBAC-malliin. Tämä laajennus erottaa organisaatio- ja tietojärjestelmätasot toisistaan ja esittelee toimivan linkityksen näiden välille. Tarve tälle laajennukselle muodostuu siitä, että organisaatioilla on käytössään useita tietojärjestelmiä, joihin kullakin käyttäjällä on eritasoisia käyttöoikeuksia. Suurissa organisaatioissa rooleja on niin paljon, että roolirakenteiden hallitseminen vaikeutuu nopeasti. Komposiittilaajennus helpottaa myös tätä työtä.

Kuvassa 4 esitettiin RBAC-malli. Komposiittilaajennus erottaa RBAC-mallin organisaatio- ja tietojärjestelmätasoiksi. Tämä Parkin ja muiden [2004] artikkelin jako on esitettynä kuvassa 9.



Kuva 9: Komposiittimalli

Kuvassa 4 esitetystä RBAC-mallista poiketen komposiittimallissa (kuva 9) oikeudet sidotaan tietojärjestelmärooleihin, jotka ketjutetaan edelleen organisaatioroolien kautta käyttäjiin.

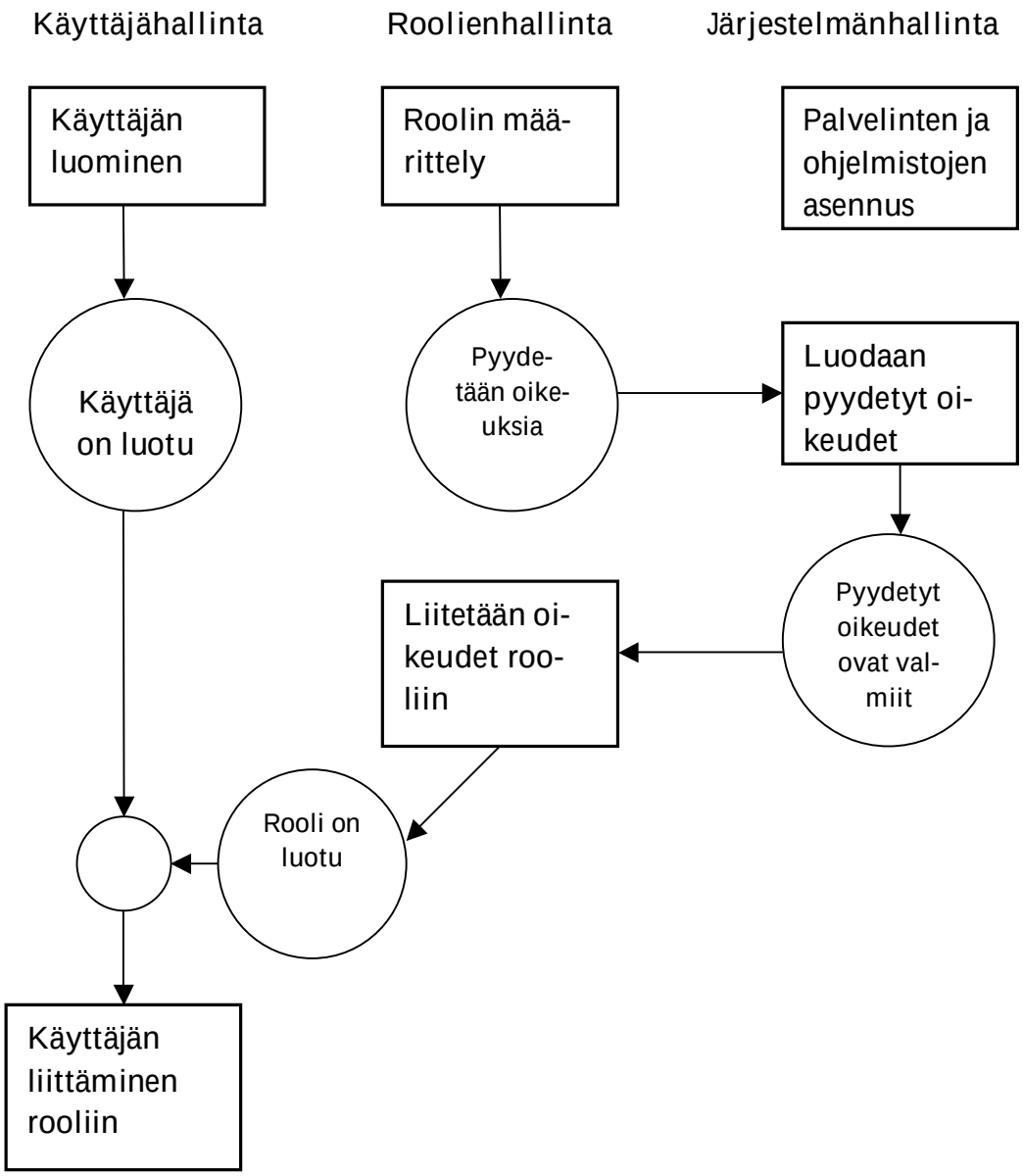
Tässä vaiheessa tutkielmaa tehdään oletus, että tietojärjestelmäroolien käyttäminen muiden roolien ja järjestelmien käyttöoikeuksien välissä tehostaa roolienhallintaa. Tämä oletus perustuu siihen, että kun tietojärjestelmäroolit on luotu, niiden avulla käyttöoikeuksien sitominen rooleihin helpottuu. Tällöin kaikkien ylläpitäjien ei tarvitse tuntea kaikkien järjestelmien käyttöoikeuksia yksityiskohtaisesti. Ylläpitäjät voivat poimia tietojärjestelmäroolit roolihakemistosta muita uusia rooleja muodostaessaan. Roolihakemistoihin palataan tarkemmin kohdassa 7.3. Tehtyä oletusta ei voida kuitenkaan tässä tutkielmassa todistaa tieteellisesti pätevästi todeksi. Tämä todistus puuttuu myös Parkin ja muiden [2004] esityksestä.

6. Roolienhallinta

Edellisessä luvussa keskityttiin rooleihin ja RBAC-malliin. Tässä luvussa läpikäydään roolien löytäminen sekä määrittely. Tämän jälkeen tutustutaan kahteen manuaalista ylläpitotyötä helpottavaan tapaan liittää roolit käyttäjiin.

6.1. Roolien löytäminen ja määrittely

Oikeiden roolien löytäminen ja määrittely on ehkä kaikista haastavin tehtävä identiteetinhallintajärjestelmän käyttöönotossa. Mitä suurempi organisaatio on kyseessä, sitä vaikeammaksi roolien määrittely tulee. Roecklen ja muiden [2000] artikkelissa korostetaan, että suuret ja kompleksiset organisaatiot tarvitsevat työkaluja roolien määrittelyyn ja ylläpitoon, jotta työmäärä pysyisi kohtuullisena. Samassa artikkelissa kuvataan myös karkealla tasolla roolien määrittelyprosessi. Tämä prosessi on esitettynä kuvassa 10.



Kuva 10: Roolien määrittelyprosessi Roecklen ja muiden [2000] mukaan

Seuraavaksi käydään läpi muutamia lähestymistapoja, joita voidaan hyödyntää roolien etsimisessä ja tarkemmassa määrittelytyössä. Esitettävät lähestymistavat täydentävät omalta osaltaan kuvassa 10 esitettyä mallia.

6.1.1. Prosessikeskeinen lähestymistapa

Prosessikeskeisessä lähestymistavassa roolien määrittely aloitetaan valitsemalla ja kuvaamalla organisaation jokin liiketoimintaprosessi. Tämän jälkeen edetään vaihe vaiheelta kohti järjestelmätasoa. Tämä lähestymistapa soveltuu erityisesti funktionaalisten ja hierarkkisten roolien löytämiseen. Prosessikeskeinen lähestymistapa esitetään seuraavaksi kuusivaiheisena.

Prosessikeskeisen lähestymistavan vaiheet:

I Liiketoimintaprosessin kuvaus

Valitaan jokin organisaation liiketoimintaprosessi ja kuvataan se esimerkiksi prosessikaaviona. Kuvaus tulee tehdä niin tarkalle tasolle asti, että toimijoiden keskeisimmät työtehtävät tulevat kuvatuiksi. Tämä mallinnustyö voi viedä paljon aikaa, mikäli organisaation prosesseja ei ole jo aiemmin mallinnettu.

II Toimijoiden ja tehtävien tunnistaminen

Valitaan kuvatusta liiketoimintaprosessista se toimija (*actor*), josta rooli halutaan muodostaa. Tämän jälkeen kerätään kuvauksesta ne toimijaan liittyvät työtehtävät, joissa hyödynnetään tietojärjestelmiä eli käytetään tietojärjestelmien toimintoja. Tässä ja seuraavassa vaiheessa voidaan hyödyntää myös mallinnettuja käyttötapauksia (*use cases*) [Fernandez and Hawkins, 1997].

III Toimijoiden yhdistäminen

Verrataan edellisessä vaiheessa valittua toimijaa muiden liiketoimintaprosessien toimijoihin. Vertailulla pyritään kartoittamaan, onko sama toimija osallisena myös muissa liiketoimintaprosesseissa. Mikäli sama toimija toistuu eri prosesseissa, voidaan toimijat ja työtehtävät yhdistää edellisessä vaiheessa kerättyihin.

IV Roolin luominen

Tarkastetaan, ettei toimijaa vastaavaa roolia löydy jo valmiiksi roolihakemistosta tai voiko uusi rooli periä jonkin aiemmin luodun roolin ominaisuudet. Tämän jälkeen luodaan toimijaa vastaava rooli roolihakemistoon ja dokumentoidaan rooli. Roolihakemistoa käsitellään tarkemmin kohdassa 7.3. Luonnin yhteydessä roolille määrätään sopiva rooliluokka.

V Oikeuksien kiinnittäminen rooliin

Selvitetään, mitkä käyttöoikeudet vaaditaan rooliin kuuluvien työtehtävien suorittamiseen. Tarkastetaan, onko näitä käyttöoikeuksia vastaavia tietojärjestelmärooleja jo aiemmin määriteltyinä. Tarvittaessa määritellään uusia tietojärjestelmärooleja. Kiinnitetään edellisessä vaiheessa luotu rooli näihin tietojärjestelmärooleihin.

IV Roolin testaaminen

Testataan, että rooli toimii ja tarkistetaan, ettei rooli riko rajoitteita. Tarvittaessa voidaan määritellä lisää kyseessä olevaa roolia koskevia rajoitteita. Tämän jälkeen tehdään tarvittaessa korjaukset ja täydennetään roolin dokumentaatio.

6.1.2. Järjestelmäkeskeinen lähestymistapa

Järjestelmäkeskeisessä lähestymistavassa roolien määrittely aloitetaan järjestelmätasolta yksittäisistä käyttöoikeuksista, joita lähdetään koostamaan yhteen. Näitä oikeusjoukkoja yhdistetään myöhemmissä vaiheissa toisiinsa ja täydennetään muilla tiedoilla. Tämä lähestymistapa soveltuu erityisesti tietojärjestelmäroolien löytämiseen, mutta sitä voidaan hyödyntää myös muihin luokkiin kuuluvien roolien määrittelyyn. Järjestelmäkeskeisen lähestymistavan vaiheet esitetään seuraavaksi.

Järjestelmäkeskeisen lähestymistavan vaiheet:

I Järjestelmän toimintojen kartoitus

Valitaan yksi tietojärjestelmä ja kerätään lista kaikista toiminnoista, joita käyttäjien on tarkoitus sillä suorittaa. Toiminnot olisi hyvä purkaa melko yksityiskohtaiselle tasolle saakka.

II Toimintojen käyttöoikeudet

Yhdistetään jokaiseen ensimmäisen vaiheen toimintoon tarvittavat järjestelmätason käyttöoikeudet.

III Toimintojen kokoaminen

Kootaan edellisten vaiheiden toiminnot järkeviksi kokonaisuuksiksi. Esim. intranetin uutisten julkaisuoikeus voi vaatia seuraavat oikeudet: kirjautumisoikeus intranetiin sekä luku- että kirjoitusoikeudet uutissivulle.

IV Tietojärjestelmäroolien luominen

Muodostetaan edellisen vaiheen toiminnallisuuskokonaisuuksia vastaavat tietojärjestelmäroolit ja viedään ne dokumentoituina roolihakemistoon.

V Roolien yhdistäminen

Liitetään luodut tietojärjestelmäroolit muiden rooliluokkien rooleihin.

VI Roolien testaaminen

Testataan, että luodut roolit toimivat ja tarkistetaan, etteivät roolit riko rajoitteita. Tarvittaessa voidaan määritellä lisää kyseessä olevaa roolia koskevia rajoitteita. Tämän jälkeen tehdään tarvittaessa korjaukset ja täydennetään roolin dokumentaatio.

6.1.3. Organisaatiokeskeinen lähestymistapa

Organisaatiokeskeisessä lähestymistavassa roolien määrittely perustuu organisaatiokaavion tutkimiseen. Tämä lähestymistapa soveltuu organisaatioroolien löytämiseen.

Organisaatiokeskeisen lähestymistavan vaiheet:

I Organisaatiokaavion ajantasaisuus

Tarkastetaan, että organisaation organisaatiokaavio on ajan tasalla, eikä siinä ole virheitä. Toisin sanoen kaavioon tulee olla merkittynä kaikki organisaatioyksiköt ja niiden väliset hierarkkiset suhteet.

II Organisaatiokaavion laajentaminen

Etenkin suurissa organisaatioissa on syytä kerätä aliorganisaatioilta omat organisaatiokaavionsa, koska ne todennäköisesti laajentavat yhteistä kaaviota. Laajentamista voidaan jatkaa lisäämällä organisaatiokaavioon myös eri yksiköiden alaiset tiimit.

III Roolien luominen

Luodaan eri organisaatioyksiköitä vastaavat organisaatioroolit roolihakemistoon dokumentaatioineen. Tämä työ on hyvä aloittaa organisaation yläosasta, jolloin hierarkiassa alempana olevien yksiköiden roolit voivat tarvittaessa periä ylempien ominaisuudet.

IV Oikeuksien liittäminen rooleihin

Liitetään luotuihin organisaatiorooleihin halutut tietojärjestelmäroolit. Organisaatiorooleihin voidaan tätä kautta kytkeä muun muassa tarvittavat kulkuoikeudet kunkin yksikön työntekijöille.

V Roolien testaaminen

Testataan, että luodut roolit toimivat ja tarkistetaan, etteivät roolit riko rajoitteita. Tarvittaessa voidaan määritellä lisää kyseessä olevaa roolia koskevia rajoitteita. Tämän jälkeen tehdään tarvittaessa korjaukset ja täydennetään roolin dokumentaatio.

6.1.4. Asemakeskeinen lähestymistapa

Asemakeskeisessä lähestymistavassa roolien määrittely aloitetaan organisaatiokaavioon merkityistä henkilöiden hierarkkisista asemista. Näin ollen tämä lähestymistapa on läheistä sukua prosessikeskeiselle lähestymistavalle. Asemakeskeinen lähestymistapa soveltuu hierarkkisten roolien määrittelyyn.

Asemakeskeisen lähestymistavan vaiheet:

I Ammattinimikkeen valinta

Poimitaan organisaatiokaaviosta jokin sellainen ammattinimike, jota vastaavissa tehtävissä työskentelee useita henkilöitä.

II Ammattinimikkeen yksikäsitteisyyden varmistaminen

Varmistetaan, ettei samalla ammattinimikkeellä tehdä erilaisia työtehtäviä eri osissa organisaatiota. Tämä vaihe todennäköisesti karsii roolikandidaatteina toimivien ammattinimikkeiden määrää huomattavasti.

III Laajennusvaihe

Tarkistetaan, tehdäänkö valittua ammattinimikettä vastaavia työtehtäviä myös muilla ammattinimikkeillä. Mikäli tehdään, nämä samankaltaiset ammattinimikkeet voidaan mahdollisesti yhdistää samaan luotavaan rooliin.

IV Roolin luominen

Tarkastetaan, ettei ammattinimikettä vastaavaa roolia ole jo aiemmin määritelty roolihakemistoon. Samoin tarkistetaan, voiko rooli periä jonkin aiemman roolin ominaisuudet. Hierarkkisia rooleja määriteltäessä perimistä tulisi hyödyntää. Tämän jälkeen luodaan ammattinimikettä vastaava rooli roolihakemistoon ja dokumentoidaan rooli.

V Oikeuksien kiinnittäminen rooliin

Selvitetään, mitkä käyttöoikeudet vaaditaan rooliin kuuluvien työtehtävien suorittamiseen. Tarkastetaan, onko näitä käyttöoikeuksia vastaavia tietojärjestelmärooleja jo aiemmin määriteltyinä. Tarvittaessa määritellään uusia tietojärjestelmärooleja. Kiinnitetään edellisessä vaiheessa luotu rooli näihin tietojärjestelmärooleihin.

IV Roolin testaaminen

Testataan, että rooli toimii ja tarkistetaan, ettei rooli riko rajoitteita. Tarvittaessa voidaan määritellä lisää kyseessä olevaa roolia koskevia rajoitteita. Tämän jälkeen tehdään tarvittaessa korjaukset ja täydennetään roolin dokumentaatio.

6.1.5. Roolinmäärittelyn lähestymistapojen arviointi

Edellisissä alakohdissa esiteltiin neljä erilaista lähestymistapaa roolien määrittelyyn. Mikään lähestymistavoista ei sovellu kaikkien roolien määrittelyyn sellaisenaan, vaan lähestymistapa on valittava rooliluokan mukaan. Tämän lisäksi jokaiseen lähestymistapaan liittyy omat vahvuutensa ja ongelmansa. Prosessikeskeinen lähestymistapa soveltuu parhaiten funktionaalisten roolien määrittelyyn. Lähestymistapa on kuitenkin työläs, mikäli organisaation liiketoimintaprosesseja ei ole jo valmiiksi hyvin mallinnettu. Järjestelmäkeskeinen lähestymistapa on tärkeä tietojärjestelmäroolien määrittelemiseksi ja sitä tarvitaan, jotta muiden roolien käyttöoikeudet saadaan hyvin hallittaviksi. Tietojärjestelmäroolien varjopuolena on se, että ne lisäävät määrittelytyötä. Ilman tietojärjes-

telmärooleja käyttöoikeudet kytkettäisiin suoraa muihin rooleihin, jolloin määrittelyvaihe voisi olla lyhyempi. Organisaatiokeskeinen lähestymistapa auttaa organisaatioroolien määrittelyssä, mutta se ei sovellu muiden luokkien roolien määrittelyyn. Asemakeskeisen lähestymistavan soveltuvuus riippuu pitkälti organisaation rakenteesta. Tämä lähestymistapa toimii paremmin sellaisessa organisaatiossa, jossa kunkin työntekijän työtehtävät on tarkasti määritelty ja samalla tehtäväkuvalla työskentelee useita henkilöitä. Tällaisia organisaatioita voivat olla mm. terveydenhuollon ja maanpuolustuksen organisaatiot.

Esitetyt lähestymistavat eivät välttämättä sovellu suoraa sellaisinaan erilaisten organisaatioiden tarpeisiin, vaan niitä tulee soveltaa tarpeiden mukaan. Organisaation on kuitenkin tärkeää valita itselleen soveltuvat tavat määritellä roolit. Mikäli jokainen rooli suunnitellaan eri tavalla ja roolit jätetään dokumentoimatta, roolien hallinnasta tulee vaikeaa. Tällöin aiemmin luotujen roolien hyödyntäminen voi olla uusia luotaessa hankalaa ja muutostenhallinta vaikeutuu. Dokumentoimattoman roolin muokkaaminen myöhemmin saattaa aiheuttaa ennakoimattomia seurauksia, jos roolin muokkaajalla ei ole tietoa kaikista rooliin liittyvistä riippuvuuksista ja rajoitteista. Lisäksi on lähestymistavasta riippumatta huomattava, että rooleja määritteleviltä henkilöiltä vaaditaan hyvää tuntemusta sekä ammattinimikkeisiin liittyvistä työtehtävistä ja organisaation toiminnasta.

Mikään esitetyistä lähestymistavoista ei sovellu sellaisenaan perusroolien määrittelyyn. Perusroolien määrittely voidaan aloittaa etsimällä sellaisia käyttöoikeuksia, jotka koskevat suurta hajanaista käyttäjäjoukkoa. Esimerkiksi oikeus käyttää organisaation sähköpostia on tällainen käyttöoikeus. Tällaisia yleisesti tarvittuja käyttöoikeuksia voidaan yhdistää haluttuun perusrooliin. Perusrooleja voidaan etsiä myös vertailemalla jo luotuja rooleja ja etsimällä niistä mahdollisimman suurta roolijoukkoa koskevia käyttöoikeuksia.

6.2. Käyttöoikeuksien muokkaaminen päättelysääntöjen avulla

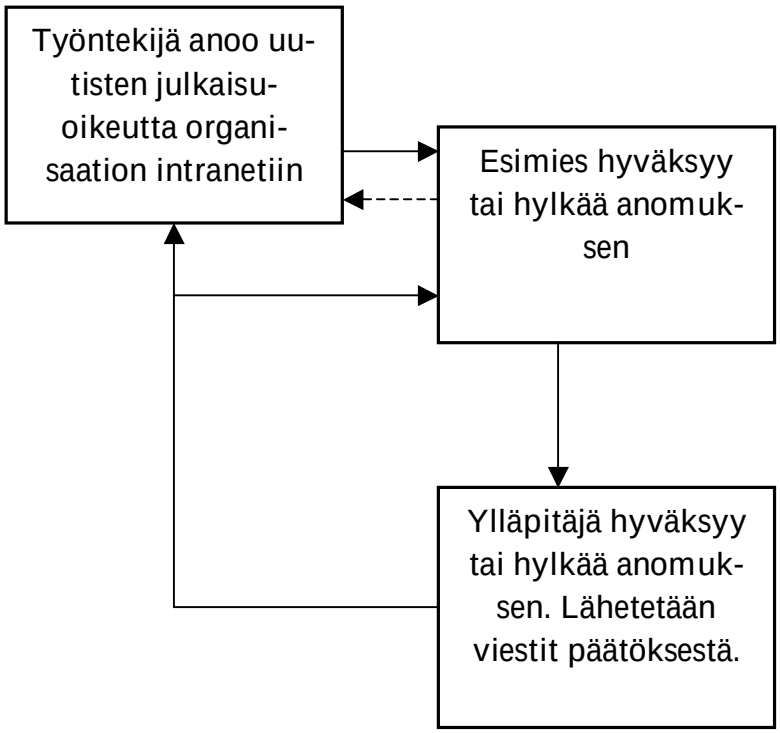
Identiteetinhallintajärjestelmiin voidaan liittää päättelysääntöjä, joiden avulla järjestelmä automaattisesti luo sekä sulkee käyttäjätunnuksia. Tällaisien sääntöjen avulla identiteetinhallintajärjestelmä voidaan valjastaa luomaan uusi käyttäjätunnus, kun organisaation HR-järjestelmään lisätään uusi työntekijä. Tällöin järjestelmä luo käyttäjän sähköisen identiteetin ja käyttäjätunnuksen. Identiteettiin voidaan tässä vaiheessa sitoa myös tiettyjä rooleja esimerkiksi sen mukaan, mihin yksikköön tai millä työnimekkeellä uusi työntekijä on rekrytoitu. Vastaavasti roolien ja tunnuksen voimassaolo voidaan sitoa tällaisiin päättelysääntöihin. Näin ollen käyttäjätunnus voidaan sulkea automaattisesti työsuhteen päättyessä.

Päättelysääntöjen hyödyntäminen vähentää manuaalista ylläpitotyötä. Jotta tällaisten sääntöjen käyttäminen on mahdollista, tulee sääntöjen käsittelemien tietojen olla oikein. Toisin sanoen mikäli HR-järjestelmää käytetään tunnusten luomisessa ja roolittamisessa, täytyy HR-järjestelmästä aina löytyä kaikkien työntekijöiden tiedot oikein tallennettuna. Tämä tarkoittaa käytännössä sitä, että uutta henkilöä rekrytoitaessa hänen tietonsa on tallennettava mahdollisimman pian HR-järjestelmään, koska hänen käyttöoikeutensa riippuvat tästä.

6.3. Käyttöoikeuksien anominen työnkulun avulla

Useat identiteetinhallintatuotteet tarjoavat erilaisia itsepalvelutoiminnallisuuksia. Eräitä tällaisista palveluista ovat työnkulut (*workflow*), joilla käyttäjät voivat hakea itselleen käyttöoikeuksia. Työnkulut ovat yleiskäyttöisiä työkaluja, joita voidaan käyttää apuna halutun lopputuloksen saavuttamiseksi prosessissa, jossa tietoa tai tehtäviä siirretään eri toimijoiden välillä. Hollingsworth [1995, 6] on määritellyt työnkulut seuraavasti: ”*Workflow is concerned with the automation of procedures where documents, information or tasks are passed between participants according to a defined set of rules to achieve, or contribute to, an overall business goal. Whilst workflow may be manually organised, in practice most workflow is normally organised within the context of an IT system to provide computerised support for the procedural automation.*”

Käyttöoikeuksien anomisessa työnkulut toimivat usein siten, että käyttäjä kirjautuu organisaation intranetiin ja avaa tätä kautta käyttöoikeuksien hakemista varten luodun sivun. Sen kautta työntekijä anoo itselleen työtehtävissään tarvitsemia käyttöoikeuksia, joita hänelle ei ole jo aiemmin myönnetty. Käyttöoikeuspyyntö voidaan ohjata työnkulun avulla työntekijän esimiehelle, joka tarkistaa, että anomus on asianmukainen. Esimiehen tekemän hyväksynnän jälkeen anomus voidaan automatisoidusti ohjata sille ylläpitäjälle, joka vastaa sen tietojärjestelmän käyttöoikeuksista, jonka oikeuksia anomus koskee. Kun ylläpitäjä on hyväksynyt pyynnön, käyttöoikeus astuu voimaan. On tärkeää huomata, että vaikka edellä puhuttiinkin ”käyttöoikeuksien anomisesta”, todellisuudessa kyseessä on roolin anominen. Näin ollen yhden anottavan roolin mukana käyttäjälle saattaa tulla suurikin joukko yksittäisen tietojärjestelmän sisäisiä käyttöoikeuksia. Eräs tällainen anottavissa oleva rooli voisi olla ”uutisten julkaisuoikeus intranetissä”. Tämä esimerkki on kuvassa 11.



Kuva 11: Käyttöoikeuden anominen työnkulun avulla

Työnkulkujen hyödyllisiä ominaisuuksia on koottu alla olevaan listaan:

- Kun kaikki työntekijät ohjeistetaan käyttämään työnkulkuja, kaikki käyttöoikeuspyynnöt kulkevat virallista reittiä.
- Edellisen seurauksena esimies on tietoinen alaistensa toimista. Näin kenellekään ei pitäisi tulla sellaisia käyttöoikeuksia, jotka eivät hänelle kuulu.
- Toinen ensimmäisen kohdan seuraus on se, ettei ylläpitäjän tarvitse ottaa oikeuspyyntöjä enää muita reittejä. Tämä helpottaa ja ylläpitäjän työtehtäviä. Ylläpitäjän ei enää tarvitse vastaanottaa oikeuspyyntöjä sähköpostilla tai puhelimitse. Näin työaikaa säästyy ja sosiaalisen hakkeroinnin mahdollisuudet pienenevät.
- Työnkulku kokoaa yhteen kaikki oikeuspyyntöön liittyvät tiedot oikeassa muodossa. Tämä helpottaa päätöksentekoa siitä, tuleeko pyyntö hyväksyä vai hylätä.
- Haettu rooli saadaan automaattisesti kiinnitettyä hakijan käyttäjätunnukseen, kun sen myöntäminen on hyväksytty. Perusylläpitäjän ei tarvitse tehdä käyttöoikeuksien hallintaa enää yksittäisen tietojärjestelmän tasolla. Tämä säästää ylläpitäjän työaikaa, eikä ylläpitäjällä tarvitse olla yhtä syvällistä tuntemusta kohdetietojärjestelmän käyttöoikeuksista kuin aiemmin.
- Työntekijälle voidaan antaa mahdollisuus seurata, missä vaiheessa käsittelyä hänen hakemuksensa on.

- Työntekijälle voidaan lähettää automaattinen sähköpostiviesti, kun haetut oikeudet on asetettu voimaan tai mikäli hakemus on hylätty.
- Työnkulkujen suunnittelu on helppoa kunnollisella työkalulla ja työnkulkuja voidaan muokata nopeastikin käyttötarpeiden muuttuessa.
- Hyväksymisportaiden määrä voi vaihdella nollasta useaan. Hyväksyjät voivat olla myös rinnakkaisia, jolloin yhdenkin hyväksyntä voi riittää.
- Useassa työnkulkusovelluksessa voidaan määritellä myös monimutkaisempia sääntöjä. Esim. rinnakkaisten hyväksyjien hyväksymispäätöksistä voidaan muodostaa äänestys, jonka perusteella lopullinen hyväksymispäätös tehdään.

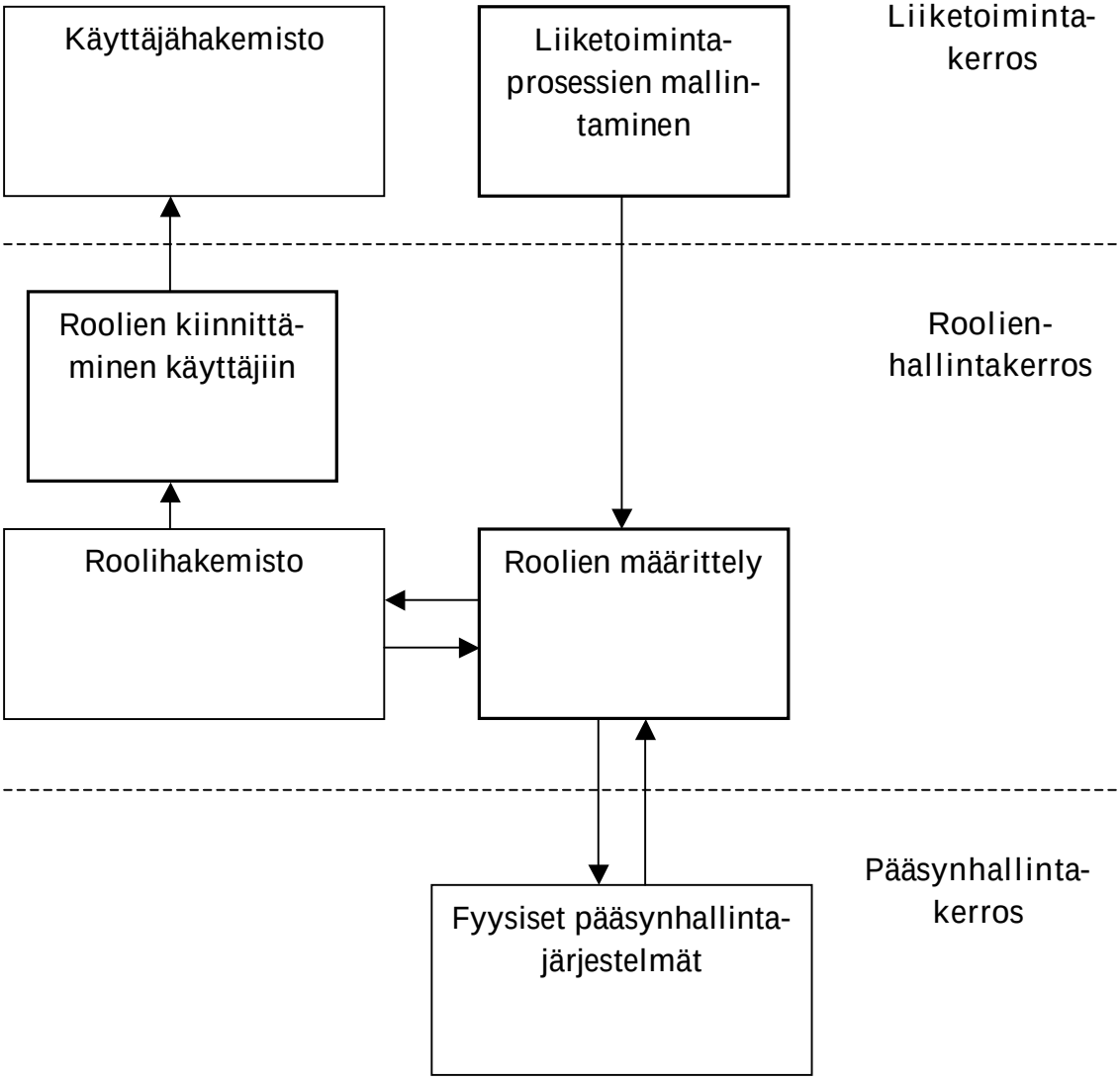
Jotta työnkulut toimisivat, tulee huomioida seuraavat seikat:

- Esimiesten tulee tuntea alaistensa työtehtävät ja niihin liittyvät käyttöoikeustarpeet.
- Anottavissa olevat roolit täytyy kuvata käyttäjille selkeästi, jotta he osaavat hakea aina oikeita tarvitsemiaan käyttöoikeuksia.
- Mikäli kyseessä on suuri organisaatio, jolla on käytössään runsaasti tietojärjestelmiä, on hyvin todennäköistä, ettei kaikkia tietojärjestelmiä saada ainakaan heti identiteetinhallintajärjestelmän käyttöönottovaiheessa integroitua identiteetinhallintaan. Näin ollen kaikkia käyttöoikeuksia ei ole mahdollista anoa työnkulkujen kautta. Organisaation kannattaisikin pyrkiä siihen, että kaikki ne oikeudet, joihin muutospyyntöjä tehdään kaikkein eniten, voitaisiin hallita työnkulkujen kautta.
- Organisaation on laadittava säännöt myös siitä, kuinka sellaisia käyttöoikeuksia anotaan, jotka eivät ole automatisoitujen työnkulkujen avulla anottavissa.
- Työnkuluissa on huomioitava myös se, että työntekijät eivät ole aina paikalla. Etenkin pidempien lomien ajaksi työnkulkujen on voitava joustaa, jottei anomusprosessi pysähdy paikalleen yhden henkilön ollessa lomalla. Tällöin hyväksymistehtävä on siirrettävä loman ajaksi jollekin toiselle työntekijälle.

7. Rooliperustaisen pääsynhallinnan arkkitehtuuri

7.1. Arkkitehtuurikerrokset

Roeckle ja muut [2000] esittävät identiteetinhallintajärjestelmän roolienhallinnan koskevan liiketoiminta-, roolienhallinta- ja pääsynhallintakerrosta. Tämä jaottelu on esitetty kuvassa 12. Esitystä on muutettu alkuperäisestä yleistämällä sitä. Tätä kuvaa voidaan pitää myös abstraktina roolienhallinnan arkkitehtuurikuvauksena.



Kuva 12: Roolienhallinnan arkkitehtuuri

Seuraavissa kohdissa kerrotaan lyhyesti edellä esitettyyn arkkitehtuuriin kuuluvista käyttäjä- ja roolihakemistoista.

7.2. Käyttäjähakemisto

Käyttäjähakemistolla viitataan hakemistoon, jonne organisaation sähköiset identiteetit ja niihin liittyvät käyttäjätunnukset ovat tallennettuina. Ehkä laa-

jimmalle käyttöön levinnyt tällainen tuote on Microsoft Active Directory. Roolit esiintyvät käyttäjähakemistossa identiteetteihin liitettyinä attribuutteina tai käyttöoikeusryhminä.

Kuvassa 12 käyttäjähakemisto on piirretty liiketoimintakerrokselle, koska hakemisto kuvaa koko organisaation käyttäjäkunnan ja on siten erityisen tärkeä organisaation harjoittamalle liiketoiminnalle. [Roeckle et al., 2000]

7.3. Roolihakemisto

Termillä roolihakemisto viitataan tässä tutkielmassa sellaiseen järjestelmään, johon organisaatio on tallentanut kuvaustiedot kaikista pääsynhallintaan käyttämistään rooleista. Joissakin englanninkielisissä lähteissä tällaiseen järjestelmään viitataan termillä *Role-catalog* [Roeckle et al., 2000]. Roolihakemisto on toteutettavissa monella tavalla esimerkiksi LDAP-hakemistona tai relaatiotietokantana. Pääsynhallinnan kannalta roolihakemiston tulisi toteuttaa seuraavat ominaisuudet:

- Roolihakemistosta tulee löytyä kaikki organisaation roolit.
- Jokainen rooli on kuvattuna ja dokumentoituna riittävälle tasolle saakka.
- Jokaiselle roolille on tallennettuna luokkatieto.
- Roolien kuvaukset, rooleihin liittyvät rajoitteet ja suhteet muihin rooleihin ovat dokumentoituna.

Näiden rooleja kuvaavien tietojen avulla roolienhallinta helpottuu. Uusia rooleja perustettaessa voidaan esimerkiksi tarkistaa, onko jo aiemmin luotuja rooleja mahdollista hyödyntää uuden roolin pohjana.

8. Yhteenveto

Tutkielmassa esiteltiin identiteetinhallinnan ja rooliperustaisen pääsynhallinnan perusteet. Esitys eteni peruskäsitteiden määrittelyistä kohti roolienhallintaa ja rooliperustaisen pääsynhallinnan arkkitehtuuria. Seuraavissa alakohdissa kootaan ohjerunko identiteettien ja roolien määrittelemiseksi, arvioidaan tutkimustuloksia esitettyjen tutkimusongelmien valossa sekä pohditaan jatkotutkimusmahdollisuuksia.

8.1. Ehdotus identiteettien ja roolien määrittelemiseksi

Tässä kohdassa kootaan edellisten lukujen perusteella tiivis ohjerunko, jota erilaiset organisaatiot voivat soveltaa identiteettien ja roolien määrittelemisessä. Ohjerunko koostuu kahdeksasta vaiheesta. Organisaatio voi laajentaa annettua runkoa muun muassa tässä tutkielmassa aiemmin esitetyillä yksityiskohdilla tai lisäämällä kokonaan uusia vaiheita. Ohjeessa oletetaan, että organisaation liiketoimintaprosessit on mallinnettu sekä organisaatiolla on jo käytössään identiteetinhallintajärjestelmä.

I Identiteettien määrittely

Määritellään identiteetteihin kuuluvat perustiedot kuten henkilöiden nimi- ja yhteystiedot. Tämän jälkeen valitaan identiteeteille soveltuvat yksilöivät tunnisteet.

Identiteettien määrittelyn tuloksena syntyy skeema, joka kuvaa identiteettien tietorakenteen. Tähän skeemaan on hyvä lisätä tiedot siitä, mistä tietojärjestelmästä mikin identiteettiin kuuluva tieto noudetaan. Esimerkiksi henkilön etunimi noudetaan HR-järjestelmän henkilötietojen tietystä tietokentästä.

II Henkilörekisteriselosteen laatiminen

Edellisessä vaiheessa kerättyjen tietojen perusteella laaditaan henkilörekisteriseloste.

III Yleisten rooleja koskevien rajoitteiden määrittely

Tässä vaiheessa sovitaan organisaation yleiset rajoitteet, joita sovelletaan identiteetinhallinnassa. Rajoitteista selkein on vähäisimpien oikeuksien periaate, joka on siten helposti käyttöönotettavissa. Tämän lisäksi voidaan kirjata ylös organisaation ydinliiketoiminnasta ja keskeisimmistä tietojärjestelmistä nousevia rajoitteita.

Uuden roolin lisääminen saattaa mahdollistaa ei-toivottuja rooliyhdistelmiä. Tästä syystä rajoitteita tulee täydentää sitä mukaa, kun uusia rooleja määritellään.

IV Tietojärjestelmäroolien määrittely

Määritellään keskeisimmät tietojärjestelmäroolit järjestelmäkeskeistä lähestymistapaa apuna käyttäen.

V Perusroolien määrittely

Määritellään yleisimmät perusroolit. Tällaisia voivat olla muun muassa ”organisaation työntekijä” ja ”organisaation asiakas” -roolit. Perusrooleja voidaan lisätä sitä mukaa, kun tarve niille havaitaan muita rooleja määritellessä.

VI Muiden roolien määrittely

Määritellään muiden rooliluokkien roolit. Tässä määrittelytyössä voidaan hyödyntää kaikkia tutkielmassa aiemmin esitettyjä roolien määrittelyyn liittyviä lähestymistapoja.

VII Roolien suhteiden määrittely

Määritellään roolien väliset suhteet. Keskeistä tässä vaiheessa on määritellä tietojärjestelmä- ja perusroolien linkittyminen muihin rooleihin.

VIII Päättelysääntöjen ja työnkulkujen määrittely

Määritellään päättelysäännöt, joilla roolit liitetään identiteetteihin. Esimerkiksi luodaan päättelysääntö, joka yhdistää ”organisaation työntekijä” -roolin kaikkien yrityksen työntekijöiden sähköisiin identiteetteihin. Lisäksi määritellään työnkulut, joilla käyttäjät voivat hakea käyttöoikeuksia.

8.2. Tulosten arviointi

8.2.1. Roolien määrittelyn kehittäminen

Tutkielman ensimmäinen tutkimusongelma oli ”Miten roolien määrittelyä voitaisiin kehittää ad hoc -tyyppisestä toiminnasta kohti harkittua suunnitteluprosessia?”. Tätä tarkoitusta varten tutkielmassa esiteltiin neljä erilaista lähestymistapaa roolien määrittelyyn, uusi roolienluokittelumalli sekä ohjerunko tukemaan identiteettien ja roolien määrittelyä. Roolien määrittelyyn tarjottiin erilaisia lähestymistapoja, koska selvästikään ei ole olemassa yhtä tapaa, joka soveltuisi hyvin kaikkien eri luokkien roolien määrittelyyn.

Johdannossa luvattiin tutkielman toimivan ohjeena roolien määrittelyssä sellaisille organisaatioille, jotka suunnittelevat identiteetinhallintajärjestelmän käyttöönottoa. Aiemmin asetettujen rajausten puitteissa tämä tavoite voitaneen katsoa saavutetuksi. Tutkielma ei ole täydellinen opas siitä, kuinka identiteetinhallintajärjestelmä tulisi käyttöönottaa, mutta siinä kuitenkin esitetään ohjerunko

roolien määrittelyyn ja jaetaan runsaasti ohjeita koskien identiteettejä ja rooleja. Annettujen ohjeiden ei ole tarkoitus olla kiveen hakattuja totuuksia, vaan antaa identiteetinhallintaan tutustuvalle organisaatiolle eväitä aloittaa oma keskustelunsa näistä aiheista. Lopullinen tavoite on se, että kukin organisaatio löytäisi itselleen parhaimman tavan toteuttaa identiteetinhallinta.

Tutkielmassa esitetyt ratkaisut eivät ole täysin valmiita käyttöönotettaviksi, vaan ne tulee sovittaa kunkin organisaation toimintaan sopiviksi. Esitettyihin malleihin on tarkoituksella jätetty liikkumavaraa organisaatioille. Tämä linjaus on valittu, koska organisaatiokulttuureissa on suuria eroja, eikä esitettäviä malleja haluttu lukita vain tietyntyyppisten organisaatioiden käyttöön. Mallien mahdollisena heikkoutena voi pitää sitä, ettei niiden toimivuutta ole testattu suurella testiaineistolla. Etenkin tietojärjestelmäroolien määrittelyä ja käyttöä olisi hyvä testata. Testeillä tulisi selvittää, tuleeko tietojärjestelmäroolien käyttö lopulta taloudellisemmaksi kuin käyttöoikeuksien kiinnittäminen suoraan muihin rooleihin. Tutkielmassa on lähdetty siitä oletuksesta, että tietojärjestelmäroolien käytöllä on mahdollista saavuttaa säästöjä roolienhallintatyössä. Tätä oletusta ei ole näytetty toteen tieteellisesti pitävillä perusteluilla. Oletuksen paikkansapitävyyteen vaikuttavia tekijöitä ovat ainakin organisaation koko ja rakenne, tietojärjestelmien ja niiden tarjoamien toiminnallisuuksien lukumäärä, roolienhallintaan osallistuvien henkilöiden määrä sekä heidän osaamisensa. Tietojärjestelmäroolien käyttäminen lisää todennäköisesti aluksi määrittelytyömäärää, mutta niiden hyödyntämisen pitäisi helpottaa muiden luokkien roolien määrittelyä.

Esitetty roolienluokittelumalli yhdistää aiemmin kehitettyjen mallien hyviä puolia yhteen. Aiemmin esitettyjen mallien vahvuudet myös kumoavat toistensa heikkouksia. Näin ollen uusi luokittelumalli voi soveltua paremmin erilaisten organisaatioiden tarpeisiin kuin kumpikaan alkuperäisistä malleista yksinään.

8.2.2. Identiteettien yksilöivät tiedot

Toinen tutkielman tutkimusongelma oli ”Miten sähköiset identiteetit tulisi erotella toisistaan, eli millaisia yksilöiviä tunnistetietoja identiteetteihin tulisi liittää?”. Esitetyt identiteettien yksilöivien tietojen vahvuudet ja heikkoudet voivat vaikuttaa triviaalitiedoilta. Tästä huolimatta organisaatiot käyttävät hyvin usein eri tietojärjestelmissä erilasia tunnistetietoja, jotka vaikeuttavat järjestelmäintegraatioiden toteuttamista ja henkilötietojen synkronointia eri järjestelmien välillä. Mikään identiteettien yksilöivä tieto ei osoittautunut täysin ongelmattomaksi. Mikäli järjestelmässä käsitellään ainoastaan Suomen kansalaisten tietoja, luontevin valinta tunnisteeksi on henkilötunnus. Erityisesti tällöin on kuitenkin huomioitava tietosuoja ja lainsäädäntö. Jos järjestelmässä on tarkoi-

tus käsitellä muidenkin kohteiden kuin Suomen kansalaisten identiteettejä, organisaation on parasta sopia oma yksilöivä tunnisteensa. Oli tunnistetieto mikä tahansa, se tulee levittää jokaiseen järjestelmään, joissa identiteetteihin liittyviä tietoja käsitellään. Mikäli tämä ei ole mahdollista, eräs kompromissiratkaisu on liittää identiteetteihin useampia tunnisteita. Tällöin kukin identiteetinhallintajärjestelmään integroitu järjestelmä voi hyödyntää sopivaa tunnistetta.

Tutkielmassa esitettiin erilaisiin yksilöiviin tunnistetietoihin liittyvät keskeisimmät vahvuudet ja heikkoudet. Näiden tietojen pohjalta organisaatiot voivat valita, minkä tyyppinen tunniste soveltuu organisaation käyttöön.

8.3. Jatkotutkimus

Jatkotutkimusta varten tutkielmassa esitettyjä rajauksia voitaisiin purkaa. Tarkastelun ulottaminen koskemaan useamman organisaation välistä toimintaa tuo vastaan uusia mielenkiintoisia tutkimushaasteita. Samoin tämän tutkielman ulkopuolelle rajattu käyttäjien tunnistaminen sisältää haasteita, joita ei ole täydellisesti vielä ratkaistu. Esimerkiksi mikäli organisaatio tarjoaa sähköisiä palveluita suurelle heterogeeniselle asiakaskunnalle, asiakkaiden tunnistaminen ja sitä kautta kytkeminen sähköisiin identiteetteihin on haasteellista. Lisäksi jatkotutkimuskohteena voisi olla esitettyjen mallien testaaminen suurilla tietomassoilla.

Viiteluettelo

[Ailisto et al., 2005] Heikki Ailisto, Pasi Ahonen, Mikko Lindholm, Biometrisen tunnistamisen tietoturvallisuus ja yksityisyydensuoja. Liikenne ja viestintäministeriö, 2005. Artikkeli luettavissa http://www.lvm.fi/fileserver/Julkaisuja%2080_2005.pdf, tark. 7.5.2008.

[Alkula, 2007] Riitta Alkula, Terveydenhuollon kansallinen tietojärjestelmäarkkitehtuuri KANTA-jatkomäärittely, 2007, Sosiaali- ja terveysministeriö. Artikkeli luettavissa http://www.stm.fi/Resource.phx/vastt/tietoh/jatkomaar.htx.i479.pdf, tark. 16.5.2008.

[Asetus 198, 1970] Väestökirja-asetus 198/1970. Vanhentunut asetus luettavissa http://www.finlex.fi/fi/laki/alkup/1970/19700198, tark. 7.5.2008.

[Asetus 288, 2001] Valtioneuvoston asetus yritys- ja yhteisötietojärjestelmästä, 29.3.2001/288. Ajantasainen asetus luettavissa http://www.finlex.fi/fi/laki/ajantasa/2001/20010288, tark. 7.5.2008.

[Asetus 473, 1963] Sairausvakuutusasetus 473/1963. Vanhentunut asetus luettavissa http://www.finlex.fi/fi/laki/alkup/1963/19630473, tark 7.5.2008.

[ATK-sanakirja, 2004] Tietotekniikan liitto, *Tietotekniikan liiton ATK-sanakirja*, Talentum, 2004.

[Bishop, 2003] Matt Bishop, *Computer Security: Art and Science*, Addison-Wesley, 2002.

[Cameron, 2005] Kim Cameron, The Laws of Identity, 2005. Artikkeli luettavissa http://www.identityblog.com/stories/2005/05/13/TheLawsOfIdentity.pdf, tark. 7.5.2008.

[DWP, 2006] Department for Work and Pensions, Applying for a National Insurance number. Luettavissa http://www.dwp.gov.uk/lifeevent/benefits/ni_number.asp, tark. 7.5.2008

[Fernandez and Hawkins, 1997] E. B. Fernandez and J. C. Hawkins, Determining Role Rights from Use Cases. In: *Proceedings of the second ACM workshop on Role-based access control*, Fairfax, Virginia, United States, November 6-7, 1997, ACM, 121-125.

[Hollingsworth, 1995] David Hollingsworth, The Workflow Reference Model, Workflow Management Coalition, 1995. Artikkeli luettavissa http://www.wfmc.org/standards/docs/tc003v11.pdf, tark. 7.5.2008.

[JHS 159, 2006] JHS 159, 2006. Suositus luettavissa http://www.jhs-suositukset.fi/suomi/jhs159, tark. 7.5.2008.

[Järvinen, 2006] Petteri Järvinen, Sähköinen identiteetti – avain sähköisiin palveluihin. Luettavissa http://www.sahkoinenhenkilokortti.fi/vrk/fineid/files.nsf/files/57C341ACB2C847A0C225721B002FDF0B/$file/01_PJ.pdf, tark. 7.5.2008.

[Leenes et al., 2007] Ronald Leenes, Jan Schallaböck, Marit Hansen, PRIME white paper v2, Privacy and Identity Management for Europe, 2007. Artikkeli luettavissa https://www.prime-project.eu/prime_products/whitepaper, tark 7.5.2008.

[Longstaff et al., 2003] Jim Longstaff, Mike Lockyer, John Nicholas, The Tees Confidentiality Model: an Authorisation Model for Identities and Roles. In: *SACMAT 03,* Como, Italy, June 2–3, 2003, ACM, 125-133.

[L523, 1999] Henkilötietolaki 523/1999. Ajantasainen laki luettavissa http://www.finlex.fi/fi/laki/ajantasa/1999/19990523, tark. 7.5.2008.

[Microsoft, 2004] MIIS 2003 Product Overview. Luettavissa http://www.microsoft.com/technet/miis/evaluate/overview.mspx, tark. 7.5.2008.

[Park et al., 2004] Joon S. Park, Keith P. Costello, Teresa M. Neven, Josh A. Diosomito, A composite rbac approach for large, complex organizations. In: *Proceedings of the ninth ACM symposium on Access control models and technologies,* Yorktown Heights, New York, USA, 2004, 163-172.

[Pfitzmann and Hansen, 2008] Andreas Pfitzmann, Marit Hansen, Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management – A Consolidated Proposal for Terminology, Technische Universität Dresden, 2008. Artikkeli luettavissa http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.31.pdf, tark. 7.5.2008.

[RFC 1777, 1995] RFC 1777 – Lightweight Directory Access Protocol. Luettavissa http://www.faqs.org/rfcs/rfc1777.html, tark 7.5.2008.

[RFC 2460, 1998] RFC 2460 – Internet Protocol, Version 6 (Ipv6) Specification. Luettavissa http://www.faqs.org/rfcs/rfc2460.html, tark 7.5.2008.

[RFC 791, 1981] RFC 791 – Internet Protocol Darpa Internet Program Protocol Specification. Luettavissa http://www.faqs.org/rfcs/rfc791.html, tark. 7.5.2008.

[Roeckle et al., 2000] Haio Roeckle, Gerhard Schimpf, Rupert Weidinger, Process-Oriented Approach for Role-Finding to Implement Role-Based Security Adminstration in a Large Industrial Organization. In: *Symposium on Access Control Models and Technologies, Proceedings of the fifth ACM workshop on Role-based access control*, Berlin, Germany, 2000, ACM, 103-110.

[Sandhu et al., 1996] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, Charles E. Youman, Role-Based Access Control Models. *Computer* 29, 2 (Feb. 1996), 38-47.

[Sandhu and Munawer, 1999] Ravi Sandhu, Qamar Munawer, The ARBAC Model for Administration of Roles. In: *15th Annual Computer Security Applications Conference (ACSAC '99)*, 1999, IEEE Computer Society, p. 229.

[Shaw, 2007] Jackson Shaw, Tenet of Identity Management, Quest Software, 2007. Artikkeli luettavissa http://www.quest.com/common/registration.aspx?requestdefid=12751, tark. 7.5.2008.

[Slone, 2004] Skip Slone, Identity Management. The Open Group, 2004. Artikkeli luettavissa http://www.opengroup.org/bookstore/catalog/w041.htm, tark. 7.5.2008.

[Tietosuojavaltuutetun toimisto] Tietosuojavaltuutetun toimisto, Tietosuoja- ja rekisteriselosteet. Luettavissa http://www.tietosuoja.fi/2584.htm, tark. 7.5.2008.

[Väestörekisterikeskus] Väestörekisterikeskus, Henkilötunnus. Luettavissa http://www.vaestorekisterikeskus.fi/vrk/home.nsf/pages/32C708B9CA6C5824C2256CB70038D048?Opendocument, tark. 7.5.2008.

[Windley, 2005] Phillip Windley, Digital Identity. O'Reilly, 2005.