# ERÄÄN SYSTEEMITYÖMALLIN ARVIOINTI TIETOTURVANÄKÖKULMASTA

**8. Turvallisuusjohdon koulutusohjelma**
**Koulutuskeskus Dipoli**
**Tutkielma 1.2.2006**
**Jari Pirhonen**
**japi @ iki.fi**

8. Turvallisuusjohdon koulutusohjelma

Jari Pirhonen

Erään systeemityömallin arviointi tietoturvanäkökulmasta

## Tiivistelmä

Tietotekniikan ja tietoverkkojen sekä niihin pohjautuvien palvelujen jatkuva kehittyminen korostaa useissa organisaatiossa tietoturvallisuuden roolia turvallisuusjohtamisen kentässä. IT-alan kypsymättömyys verrattuna perinteisiin aloihin ja jatkuva tekniikan kehittyminen tarkoittaa, että parhaita käytäntöjä ja kokemusperäistä osaamista ei ole vielä riittävästi. Tietoturvaratkaisut perustuvat liian usein reaktiivisiin ratkaisuihin.

Palvelusovellusten tietoturvallisuutta ei voi jättää erillisten tietoturvatuotteiden ja jälkikäteen tehtävän tietoturvatarkastuksen varaan. Sovelluksista itsessään on tehtävä väärinkäyttöyrityksiä sietäviä. Tämä vaatii tietoturvallisuuden kokonaisvaltaista huomioimista jo sovelluskehitysvaiheessa ja koko tietojärjestelmän elinkaaren ajan.

Tutkimukset ja kokemukseni ovat osoittaneet, että tietojärjestelmiä kehitettäessä tietoturvatarpeita ei määritellä tarpeeksi suunnitelmallisesti systeemityön alkuvaiheissa. Samoin tietoturvaosaaminen sovellussuunnittelussa, ohjelmoinnissa ja testauksessa on usein riittämätöntä.

Organisaation systeemityömalli ohjaa tietojärjestelmien kehitystä. Systeemityöammattilaisten riittävän tietoturvakoulutuksen lisäksi on välttämätöntä, että käytettävä systeemityömalli ohjaa ja antaa riittävät työkalut tietoturvallisuuden huomioimiseen kaikissa systeemityön vaiheissa.

Tutkielmassa arvioidaan erään kaupallisen systeemityömallin soveltuvuutta organisaatiomme tietoturvakriittisten pankkijärjestelmien tuottamiseen. Tutkielma antaa organisaatiomme systeemityövastaaville arvion systeemityömallin sisältämien tietoturvatehtävien riittävyydestä sekä ehdotuksia systeemityömallin kehittämiseen.

Tutkielman johtopäätös on, että arvioitu systeemityömalli ei riittävästi tue tietoturvakriittisten sovellusten tuottamista. Systeemityömallia tulee kehittää tietoturvallisuuden paremmin huomioivaksi.

# Sisällysluettelo

*1 Johdanto* ..... *4*

*2 Tutkielman tavoite ja toteutustapa* ..... *9*

*3 Sovellusten ja systeemityön tietoturvavaatimuksia* ..... *11*

**3.1 Tietoturvallisen sovelluksen määritelmä** ..... **11**

**3.2 Tutkimuksia sovellustietoturvasta** ..... **15**

**3.3 Pankkisovellusten tietoturvavaatimukset** ..... **18**

**3.4 Tietoturvastandardeja ja suosituksia** ..... **20**

3.4.1 Comprehensive, Lightweight Application Security Process (CLASP) ..... 20
3.4.2 Control Objectives for Information and related Technology (COBIT) ..... 21
3.4.3 ISF Standard of Good Practice for Information Security (ISF SoGP) ..... 21
3.4.4 ISO/IEC 15408, Evaluation Criteria for Information Technology Security (Common Criteria, CC) ..... 22
3.4.5 ISO/IEC 17799:2005 ja ISO/IEC 27001:2005 ..... 23
3.4.6 ISO 21827, Systems Security Engineering Capability Maturity Model (SSE-CMM) ..... 24
3.4.7 NIST Security Considerations in the Information System Development Life Cycle (800-64) ..... 24
3.4.8 OWASP Guide to Building Secure Web Applications ..... 24
3.4.9 Rahoitustarkastuksen standardi 4.4b operatiivisten riskien hallinta ..... 25
3.4.10 Valtionhallinnon keskeisten tietojärjestelmien turvaaminen ..... 26
3.4.11 Valtionhallinnon tietojärjestelmäkehityksen tietoturvallisuussuositus ..... 26
3.4.12 Valtionhallinnon tietotekniikkahankintojen tietoturvallisuuden tarkistuslista ..... 27
3.4.13 VISA Payment Card Industry Data Security Standard (PCI DSS) ..... 27

*4 Tietoturvallisuus systeemityössä* ..... *29*

**4.1 Parhaat käytännöt systeemityön eri vaiheissa** ..... **29**

4.1.1 Esitutkimus ..... 30
4.1.2 Määrittely ..... 30
4.1.3 Suunnittelu ..... 31
4.1.4 Toteutus ..... 32
4.1.5 Testaus ..... 33
4.1.6 Käyttöönotto ..... 34
4.1.7 Ylläpito ..... 34

**4.2 Sovelluksen tietoturvariskien hallinta** ..... **35**

*5 Tutkittavan systeemityömallin arviointi ja parannusehdotukset* ..... *38*

**5.1 Systeemityömallin yleiskuvaus** ..... **38**

**5.2 Vaatimusmäärittely** ..... **40**

5.2.1 Kuvaus ..... 40
5.2.2 Arviointi ..... 41
5.2.3 Parannusehdotuksia ..... 42

**5.3 Määrittely** ..... **43**

5.3.1 Kuvaus ..... 43
5.3.2 Arviointi ..... 44
5.3.3 Parannusehdotuksia ..... 44

**5.4 Suunnittelu** ..... **45**

5.4.1 Kuvaus ..... 45
5.4.2 Arviointi ..... 47
5.4.3 Parannusehdotuksia ..... 47

**5.5 Toteutus** ..... **48**
5.5.1 Kuvaus ..... 48
5.5.2 Arviointi ..... 49
5.5.3 Parannusehdotuksia ..... 49

**5.6 Testaus** ..... **49**
5.6.1 Kuvaus ..... 49
5.6.2 Arviointi ..... 50
5.6.3 Parannusehdotuksia ..... 50

**5.7 Käyttöönotto** ..... **50**
5.7.1 Kuvaus ..... 50
5.7.2 Arviointi ..... 51
5.7.3 Parannusehdotuksia ..... 52

**5.8 Ylläpito** ..... **52**
5.8.1 Kuvaus ..... 52
5.8.2 Arviointi ..... 53
5.8.3 Parannusehdotuksia ..... 53

**5.9 Erillinen tietoturvaohje** ..... **53**
5.9.1 Kuvaus ..... 53
5.9.2 Arviointi ..... 53
5.9.3 Parannusehdotuksia ..... 54

**5.10 Katselmointi** ..... **54**
5.10.1 Kuvaus ..... 54
5.10.2 Arviointi ..... 54
5.10.3 Parannusehdotuksia ..... 55

***6 Yhteenveto ja johtopäätökset*** ..... ***56***

***Lähdeluettelo*** ..... ***60***

# 1 Johdanto

*”We’re going to tell people that even if it means we’re going to break some of your apps, we’re going to make things more secure. You’re just going to have to go back and fix it”*
*-- Craig Mundie*

Organisaatiomme on pankkiasiakkaidemme tietojärjestelmätoimittaja, palveluintegraattori ja palvelutoimittaja. Palvelumme oleellisena osana ovat pankkitietojärjestelmät peruspankkipalveluista verkkopalveluihin. Tietoturvallisuus ja tietojärjestelmien turvallisuus ovat ehdottomia menestymisen edellytyksiä meille ja asiakkaillemme. Organisaatiossamme onkin pitkät perinteet pankkijärjestelmien kehityksessä ja korkea tietoturvatietoisuus.

Turvallisuusjohtamisen kannalta tietoturvallisuuden korostunut merkitys tarkoittaa erityishuomion kiinnittämistä palveluidemme ja tietojärjestelmiemme tietoturvalliseen toteuttamiseen. Samoin ulkoisten tietoturvastandardien, -määräysten ja -ohjeiden vaatimusten täyttäminen edellyttää tietoturvallisuuden kokonaisvaltaista huomioimista.

Tietotekniikan jatkuva kehittyminen sekä tietoverkkojen ja verkkopalvelujen entistä suurempi hyödyntäminen vaativat useimmilta muiltakin organisaatioilta merkittävää tietoturvallisuuteen panostamista. Keskuskauppakamarin yritysturvallisuustutkimukseen osallistuneista jopa 63% arvioi tulevaisuudessa painottavansa turvallisuuden kehittämisessä tietoturvallisuutta nykyistä enemmän [19].

Tietojärjestelmien tuottamisessa kehitystyötä ohjaava systeemityömalli on oleellisessa asemassa. Systeemityön tarkoituksena on tuottaa organisaation tarpeen pohjalta haluttu lopputulos eli tietojärjestelmä, joka sisältää paitsi sovelluksen, myös sitä tukevat laitteet, tuotteet, määritykset, dokumentit ja ohjeet. Systeemityötä ohjaa tietojärjestelmäarkkitehtuuri, joka koostuu teknisestä, järjestelmä- ja tietoturva-arkkitehtuurista. Organisaation projekti- ja systeemityömalli ohjaavat systeemityötä.

Perinteisesti systeemityömalleissa korostetaan tietoturvallisuuden sijaan muita ominaisuuksia, kuten helppokäyttöisyyttä, tehokkuutta ja monipuolista toiminnallisuutta. Laa-

tukriteereistä unohdetaan usein tietoturvallisuus ja tietojärjestelmien turvaamisessa luotetaan jälkikäteen rakennettuihin ratkaisuihin.

Sovellusten tietoturvallisuus on viime vuosien aikana noussut yleisenkin huomion kohteeksi. Lisääntynyt haittaohjelmien, palvelunestohyökkäysten ja tietomurtojen määrä on herättänyt huomaamaan, että sovelluksen turvaamiseksi ei riitä suojamuurin rakentaminen sen ympärille, vaan myös itse sovelluksen on oltava tietoturvallinen ja väärinkäyttöyrityksiä sietävä. USA:n presidentille laaditussa raportissa todetaan sovellusvirheiden ja –haavoittuvuuksien olevan vakava kansallinen uhka [32] ja myös Suomen kansallisessa tietoturvastrategiassa on huomioitu ohjelmistovirheiden aiheuttamat tietoturvaongelmat [21]. Tietoturvastrategiassa todetaan, että tietoturva-asioita ei tulisi tarkastella pelkästään irrallisina ohjelmistoina tai toimintoina, vaan luonnollisena osana yritysten liiketoimintaa ja tuotekehitystä toimialasta riippumatta.

Tietoturvakoulutukseen ja -sertifiointiin keskittyvä SANS julkaisee listaa Internetin 20 pahimmasta tietoturvaongelmasta. Marraskuussa 2005 julkaistussa listassa huomion kiinnittää sovellusongelmien suuri määrä. Listalla on mainittu mm. web-selaimet, toimistotyökalut, pikaviestimet ja tietokannat [36]. Väärinkäyttöyrityksiä kohdistetaan siis erityisesti perussovelluksiin. Myös yrityksiin kohdistetuissa palvelunestohyökkäyksissä on massiivisten verkkokuormitushyökkäysten sijaan entistä enemmän näkynyt suuntauksena palvelunestoyritykset sovellusongelmien kautta [31].

Tietoturvallisuuteen suhtaudutaan usein reaktiivisesti. Tietoturvallisuutta ei määritellä ja suunnitella palveluihin ja toimintoihin alusta pitäen, vaan siihen kiinnitetään huomiota pahimmillaan vasta käyttöönottovaiheessa. Valitettavasti samaa lähestymistapaa noudatetaan usein myös systeemityössä eli sovelluksia pyritään suojaamaan jälkikäteen erillisillä ratkaisuilla ja tietoturvallisuudesta koetetaan vakuuttua tekemällä tietoturvatarkastuksia ja -testauksia vasta valmiille sovellukselle. Toki näinkin saadaan korjattua osa ongelmista, mutta sovelluksesta ei ole tietoisesti ja järjestelmällisesti rakennettu tietoturvallista, sovelluksen tietoturvatarpeita ei ole määritelty eikä tietoturvaratkaisuja ole perusteltu.

Vuonna 2002 koettiin eräs käänteentekevä hetki, kun Microsoft ilmoitti panostavansa sovellustensa tietoturvallisuuteen ja esitteli tuotekehityksensä tietoturvaperiaatteet [24]:

- Tietoturvallisuuden suunnittelu: sovellukset täytyy suunnitella ja toteuttaa itseään ja käyttämiään tietoja suojaavaksi sekä väärinkäyttöyrityksiä sietäväksi.
- Oletusarvoinen tietoturvallisuus: tietoturvapiirteiden täytyy olla oletusarvoisesti käytössä ja tietoturvallista käyttöä täytyy edistää.
- Käytönaikainen tietoturvallisuus: sovellukseen täytyy liittää riittävät työkalut ja ohjeet sovelluksen tietoturvalliseen käyttöön ja hallintaan.
- Tietoturvallisuuden kommunikointi: tietoturvaongelmien ja –uhkien löytymiseen täytyy varautua. Niistä täytyy kommunikoida avoimesti ja vastuullisesti siten, että tuotteiden käyttäjät ja ylläpitäjät voivat suojautua löydettyjen uhkien varalta.

Microsoftin panostuksen syynä on tietoturvallisuuden korostuminen ja sen myötä asiakkaiden vaatimukset. On luonnollista, että yritykset priorisoivat toimintojaan asiakasvaatimusten perusteella. Organisaatioiden täytyy selkeästi vaatia sovellustoimittajiltaan paitsi tietoturvallisia sovelluksia, myös perusteita ja vakuuksia tietoturvallisuuden toteutumiselle.

Organisaatioille esitetään enenevässä määrin myös ulkopuolisia vaatimuksia tietojärjestelmien tietoturvallisuudelle. Esimerkiksi Sarbannes-Oxley, Basel ja VISA vaatimukset edellyttävät tietoturvallisuuden huomioimista systeemityön kaikissa vaiheissa ja tietojärjestelmän koko elinkaaren ajan. Myös tietoturvastandardeja ja ohjeita kehittävät tahot, kuten ISO/IEC, ISACA ja Information Security Forum esittävät vastaavat vaatimukset.

Systeemityötapaa ja erityisesti siihen liittyviä ihmisiä ei kuitenkaan muuteta hetkessä. Systeemityöhön osallistuvien ammattilaisten, kuten projektipäälliköiden, suunnittelijoiden, ohjelmoijien ja testaajien kouluttaminen tietoturvaan sekä tietoturvatehtävien integroiminen systeemityöhön on aikaa vievä asenne- ja kulttuurimuutos. Erityisen haasteen tuo se, että systeemityö, kuten IT-projektityö yleensäkin, on vielä nuori ja kehittymätön alue, jolle ei ole kehittynyt samanlaista ammatillista osaamista ja parhaita työkäytäntöjä kuin perinteisille aloille [45].

Vuonna 2002 tehtiin tutkimus 45 merkittävän sovelluksen tietoturvallisuudesta [1]. Tulokset olivat hälyttäviä – tietoturvaongelmia löytyi paljon ja jopa 70% näistä olisi vältetty paremmalla määrittelyllä ja suunnittelulla. Tyypilliset ongelmat liittyivät käyttäjän todennukseen ja valtuutukseen, istunnon hallintaan ja syötteen tarkistamiseen – siis aivan perusasioihin.

Tutkimusten mukaan tietoturvaongelmien löytäminen ja korjaaminen suunnitteluvaiheessa vähentää merkittävästi kustannuksia ja siten panostus systeemityön kehitykseen maksaa itsensä takaisin. Virheen korjaaminen tuotantokäytössä olevasta sovelluksesta voi olla jopa 100 kertaa kalliimpaa kuin saman virheen korjaaminen suunnitteluvaiheessa [40].

Olen useiden vuosien aikana ollut mukana seuraamassa läheltä sovelluskehitystä ja systeemityötä useissa eri organisaatioissa. Käytäntö on osoittanut, että usein systeemityötä tekevillä ei ole intressejä tai mahdollisuuksia tehtäviinsä liittyvien tietoturva-asioiden opiskeluun. Tämä on ymmärrettävääkin jatkuvan tekniikoiden ja työkalujen kehittymisen takia, mutta toisaalta tietoturvallisuuden perusteellinen huomioiminen on ainoa tapa tuottaa laadukkaita sovelluksia. Systeemityössä on edellytettävä myös tietoturvallisuuden osaamista ja huomioimista osana tehtäviä ja systeemityömallin on annettava tähän tukea ja työkaluja.

Organisaatiomme on arvioimassa kaupallista systeemityömallia ja sen käyttöönottomahdollisuutta aiemmin käytössämme olleen mallin tilalle. Uudella systeemityömallilla haetaan tehokkuutta erityisesti uusien sovellustekniikoiden käyttöön.

Tutkielman tavoitteena on arvioida tämä systeemityömalli tietoturvanäkökulmasta ja selvittää, onko systeemityömallissa huomioitu tietoturvallisuus riittävästi organisaatiomme tietoturvakriittisten tietojärjestelmien tarpeisiin.

Luku kaksi esittelee tarkemmin tutkielman tavoitteen ja toteutustavan. Luvussa kolme määrittelen tietoturvallisen sovelluksen. Lisäksi esittelen sovellustietoturvan tilaa kuvaavia tutkimuksia, pankkisovellusten erityisvaatimuksia sekä sovellusten ja systeemityön tietoturvallisuuteen kantaa ottavia standardeja ja suosituksia. Luvussa neljä esittelen arvioinnin pohjaksi hyviä käytäntöjä ja suosituksia tietoturvallisuuden huomioimiseksi systeemityön eri vaiheissa. Luku viisi sisältää arvioitavan systeemityömallin ku-

vauksen, arvioinnin ja parannusehdotukset systeemityövaiheittain. Luvussa kuusi esitän yhteenvedon ja johtopäätökset.

## 2 Tutkielman tavoite ja toteutustapa

*"We can't solve problems using the same kind of thinking we used when we created them"*
*-- Albert Einstein*

Tutkielman tavoitteena on selvittää organisaatiossamme arvioitavan kaupallisen systeemityömallin sopivuus tietoturvakriittisten pankkijärjestelmien tuottamiseen sekä kuinka tietoturvallisuus huomioidaan systeemityömallissa ja mitä apuvälineitä se antaa kehittäjille. Tutkielman tavoitteena on myös toimia organisaatiomme systeemityöammattilaisten ohjeena systeemityömallin jatkokehittämiseen tietoturvallisuuden osalta, joten esitän havaitut puutteet ja parannusehdotukset.

Tutkielmassa ei esitetä yksityiskohtaisia tietoturvamenetelmiä tai -ratkaisuja, vaan annetaan raamit, jonka pohjalta systeemityömallia voidaan kehittää. En arvioinut systeemityömallin yleistä sopivuutta organisaatiomme systeemi- ja projektityöhön. Se on organisaatiomme systeemityöammattilaisten tehtävä huomioiden tässä tutkielmassa esitetyt näkökohdat.

Tutkielmassa en myöskään käsittele tietoturvallisen systeemityön ympäristö- ja projektityön vaatimuksia, kuten koulutusta, työvaiheiden erottamista, testiaineistojen suojaamista, tietojen luokittelua, dokumenttien turvaamista, tms.

Arvioitava systeemityömalli on laaja sisältäen yli 350 dokumenttia tai dokumenttipohjaa. Dokumentaatio on jaettu systeemityön eri vaiheiden mukaan, jotka käsittelen vaihe kerrallaan. Dokumentaatiossa on erilliset osiot liittyen ydinprosesseihin, liiketoiminnan kehittämiseen, arkkitehtuuriin ja katselmointiin. Lisäksi mukana on koulutus- ja esittelymateriaalia. Systeemityön vaiheiden lisäksi arvioin katselmointiosuuden, joka on tietoturvallisuudenkin kannalta tärkeä. Suunnitteluvaiheen ohjeistuksessa on erillinen tietoturvaohje, jonka myös käsittelen erikseen.

Ennen arvioitavaan systeemityömalliin syventymistä tutustuin systeemityön tietoturvallisuuden standardeihin, ohjeisiin ja vaatimuksiin. Tarkoituksena oli selvittää vaatimukset ja parhaat käytännöt tietoturvallisten sovellusten toteuttamiseksi. Lisäksi halusin palauttaa mieleeni aiheesta lukemani lukuisat artikkelit ja kirjat sekä käytännön kokemuk-

seni sovellusten tietoturva-arkkitehtuureiden suunnittelusta, sovellusmääritysten ja suunnitelmien katselmoinnista sekä pitämistäni sovellusammattilaisten tietoturvakoulutuksista ja heidän kanssaan käymistäni keskusteluista.

Minulla on vuosien takaa ohjelmointi- ja systeemityötaustaa, mutta en ole systeemityön vaan tietoturvallisuuden ammattilainen. Viime vuosina olen perehtynyt sovelluksiin ja sovelluskehitykseen erityisesti tietoturvanäkökulmasta ja olen edistänyt tietoturvallisuuden integroimista systeemityöhön omissa ja asiakasorganisaatioissani sekä nykyisessä että aiemmissa tehtävissäni.

Näkemykseni on, että systeemityöammattilaisten täytyy hallita oman osa-alueensa tietoturvatehtävät. Tietoturvallisuuden huomioiminen täytyy olla luonnollinen osa kutakin tehtävää eikä erillinen ponnistus. Erityisesti korostan tietoturvallisuuden määrittelyn ja suunnittelun periaatetta. Sovelluksen tietoturvatarpeet on määriteltävä ja ratkaisut suunniteltava aivan kuten muidenkin sovellusten ominaisuuksien.

Systeemityömallin toimittaja suhtautui myötämielisesti tähän tutkimukseen. Toimittajan toivomuksesta heidän ja itse systeemityömallin nimeä ei tutkimuksessa mainita.

# 3 Sovellusten ja systeemityön tietoturvavaatimuksia

*"Reliable software does what it is supposed to do. Secure software does what it is supposed to do and nothing else."*
*-- Ivan Arce*

## 3.1 Tietoturvallisen sovelluksen määritelmä

Tietoturvallinen sovellus on hankala määriteltävä. Tarkoitetaanko sillä

- todistettavaa tietoturvallisuutta?
- virheettömyyttä?
- virhetilanteista selviytymistä?
- väärinkäytön vaikeutta?
- häiriöttömyyttä?
- selviytymistä ennalta arvaamattomissa tilanteissa?

Tavoitteet voivat olla jopa ristiriitaisia. Sovellus, joka mahdollisen virheen tai tietoturvauhkan havaitessaan lopettaa toimintansa estää kyllä väärinkäytön, mutta avaa samalla mahdollisuuden helpolle palvelunestohyökkäykselle.

Erään määritelmän mukaan tietoturvallinen sovellus [47]:

- Kestää ja sietää suurimman osan ennakoiduista väärinkäyttötapauksista ja hyökkäyksistä
- Toipuu nopeasti ja mahdollisimman pienin vahingoin kekseliäimmistä ja taitavimmista ennakoiduista väärinkäyttötapauksista ja hyökkäyksistä

Sovellus ei voi luottaa pelkästään ulkopuolisiin tietoturvaratkaisuihin, esim.:

- Palomuurin tehtävä on estää muu kuin sovellukselle tarkoitettu verkkoliikenne. Palomuuri ei suojaa sovellusvirheiltä.
- VPN salaa verkkoliikenteen ja rajoittaa tahoja, jotka voivat käyttää sovellusta, mutta ei suojaa sovellusvirheiltä.

- SSL salaa verkkoliikenteen ja varmentaa käyttäjälle sovelluksen oikeellisuuden. Sovellusta vastaan voidaan kuitenkin hyökätä yhtä hyvin käyttäen salattua verkkoyhteyttä kuin salaamatontakin.
- Hyökkäysten havainnointiin ja estoon tarkoitetut tuotteet suojaavat yleisiltä, tunnetuilta ongelmilta, mutta eivät auta sovellusten erityisominaisuuksiin liittyviin tai logiikkavirheisiin.

Tietoturvatuotteet ovat hyödyllisiä rajaamaan tietoturvariskiä, mutta sovelluksen täytyy itsessään olla riittävän turvallinen ja luotettava suojautuakseen suoraan sitä vastaan kohdistetuilta väärinkäyttöyrityksiltä.

Tietoturvallisen sovelluksen tietoturvavaatimukset on määriteltävä tietoturvariskianalyysin kautta, jotta riskitaso voidaan formaalisti hyväksyä. Toisaalta kaiken toiminnallisuuden tietoturvallinen suunnittelu ja toteutus ovat myös tärkeitä. Useat tietoturvaongelmat johtuvat puutteellisen määrittelyn tai ohjelmointivirheiden aiheuttamista sovelluksessa olevista ylimääräisistä toiminnoista tai sivuvaikutuksista.

Tutkielmaa varten määrittelen tietoturvallisen sovelluksen seuraavasti, korostaen tietoturvariskianalyysin, tietoturvavaatimusten ja tietoturvasuunnittelun merkitystä systeemityössä.

**Määritelmä:**

Tietoturvallinen sovellus toteuttaa tietoturvariskianalyysin perusteella määritellyt tietoturvavaatimuksensa siten, että jäännösriski on hyväksytty ja hallinnassa.

Muiden vaatimusten osalta sovellus toteuttaa tietoturvallisesti vain ja ainoastaan määritellyt toiminnot.

Systeemityössä on siis huomioitava sekä sopivien tietoturvaratkaisujen löytäminen ja niiden oikea toteutus että halutun toiminnallisuuden oikea ja tietoturvallinen toteutus.

Tyypillisiä syitä tietoturvallisuuden vaarantaviin ohjelmointivirheisiin ovat [7]:

- Ohjelmoijien riittämätön koulutus
- Ohjelmointikielten tietoturvaton käyttö

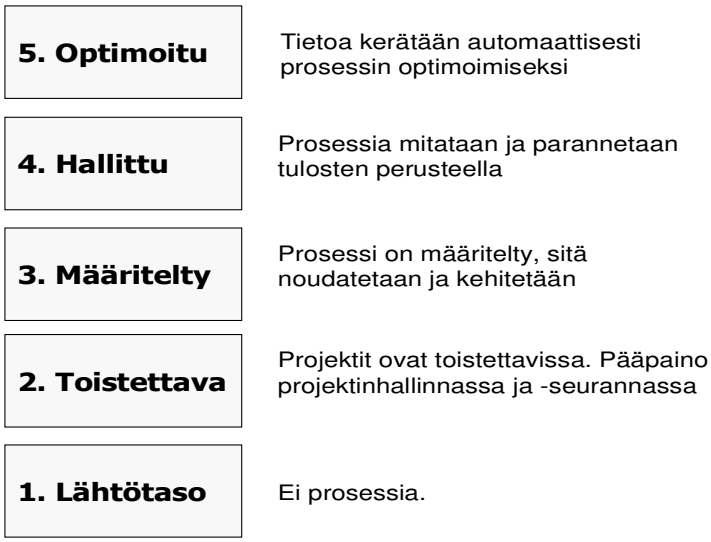
- Riittämättömät tietoturvamäärittelyt tai laatuvaatimukset
- Riittämätön katselmointi ja testaus
- Työkalujen virheellinen käyttö

ISO/IEC 9126 – standardi määrittelee tuotteen päälaatukriteereiksi toiminnallisuuden, luotettavuuden, helppokäyttöisyyden, tehokkuuden, ylläpidettävyyden ja siirrettävyyden. Tietoturvallisuus on määritelty toiminnallisuuden alakohdaksi. Tästä voidaan päätellä, että tietoturvaton sovellus ei täytä toiminnallisuuden laatuvaatimusta [37]. Tietoturvallisuus eroaa ominaisuuksiltaan muista toiminnallisista ja ei-toiminnallisista tavoitteista: on mahdotonta listata kaikki mitä sovellus *ei* saa tehdä ja on mahdotonta testauksella osoittaa sovelluksen toimivan tietoturvallisesti uuden, ennestään tuntemattoman virheen löytyessä [35].

Mielestäni sovellusprojekteissa tietoturvallisuus on nostettava yhdeksi päälaatukriteeriksi. Tietoturvatavoitteita, kuten jatkuvakäyttöisyys, tietojen ja järjestelmien eheys, luottamuksellisuus, tapahtumien jäljitettävyys ja luotettavuus on vaikea saavuttaa, ellei niitä huomioida kaikissa systeemityön vaiheissa. Verkottuneessa maailmassa, missä alihankkijoilla ja kumppaneilla on merkittävä rooli yrityksen toiminnassa, luotettavuuden osoittaminen on erityinen haaste. Kattavan, tietoturvallisuuden huomioivan systeemityömallin säännönmukainen käyttö osoittaa myös sidosryhmille yrityksen sitoutumisen tietoturvallisuuteen.

Hyväkään systeemityömalli ei yksin riitä, jos organisaation tietoturvatietoisuus ja –käyttäytyminen ei yllä systeemityöhön. Usein tarvitaan organisaatiossa kulttuurimuutos, jotta tietoturvallisuus osataan huomioida systeemityön kaikissa vaiheissa ja jotta tietoturvallisuutta arvostetaan myös käytännössä tasavertaisena tavoitteina muiden tavoitteiden kanssa.

Tietoturvallisuuden huomioiminen systeemityössä ei saa olla satunnaista ja henkilöriippuvaista. Tietoturvatyön on oltava toistettavaa, tehokasta, kehittyvää ja luotettavaa. Systems Security Engineering Capability Maturity Model (SSE-CMM) määrittelee organisaatioiden kyvykkyyden tuottaa laadukkaita, tietoturvallisia sovelluksia viidellä tasolla [17].




Organisaation siirtyminen tasolta seuraavalle vaatii tyypillisesti jopa vuosien työn ja korkeimman tason omaavat organisaatiot ovat harvassa.

Tietoturvallisuuden integrointi systeemityöhön voidaan käynnistää seuraavasti [12]:

1. Suunnitellaan, kuinka tietoturvallisuus saadaan vaiheittain osaksi nykyistä systeemityötä.
2. Otetaan yksittäiset tietoturvallisuutta parantavat käytännöt mukaan systeemityöhön pikkuhiljaa. Pilotoidaan menetelmiä ja annetaan organisaatiolle aikaa oppia.
3. Varmistetaan systeemityöhön osallistuvien tietoturvaosaaminen riittävällä koulutuksella: projektipäälliköt, arkkitehdit, suunnittelijat, ohjelmoijat ja testaajat.
4. Laaditaan mittarit. Ilman mittaamista saavutettuja hyötyjä on vaikea havaita.
5. Parannetaan toimintatapoja jatkuvasti.

Tässäkin asiassa eteneminen pienin askelin on usein paras vaihtoehto.

Microsoftin mukaan parhaat tulokset tietoturvallisuuden integroimisessa systeemityöhön saadaan ottamalla tietoturvatoimenpiteet käyttöön vaihe vaiheelta seuraavassa järjestyksessä [23]:

1. Tietoturvatavoitteiden määrittely
2. Sovellusarkkitehtuurin ja – suunnitelmien tietoturvakatselmointi
3. Uhkamallinnus
4. Sovelluskoodin tietoturvakatselmointi
5. Käyttöönoton tietoturvakatselmointi
6. Hyvien tietoturvakäytäntöjen huomioiminen suunnittelussa
7. Tietoturvatestaus

Gary McGraw esittää toimenpiteiden tehokkuuden hiukan erilaisessa järjestyksessä [22]. Hän mm. nostaa koodikatselmoinnin tärkeimmäksi ja tuo esille murtotestien ja väärinkäyttötapausten hyödyntämisen tärkeyden. Käsitykseni on, että Microsoftin näkemys on tietoturvamielessä tehokkaampi, mutta McGrawn esittämät toimenpiteet ja niiden järjestys voivat olla joillekin organisaatioille helpompi etenemistapa.

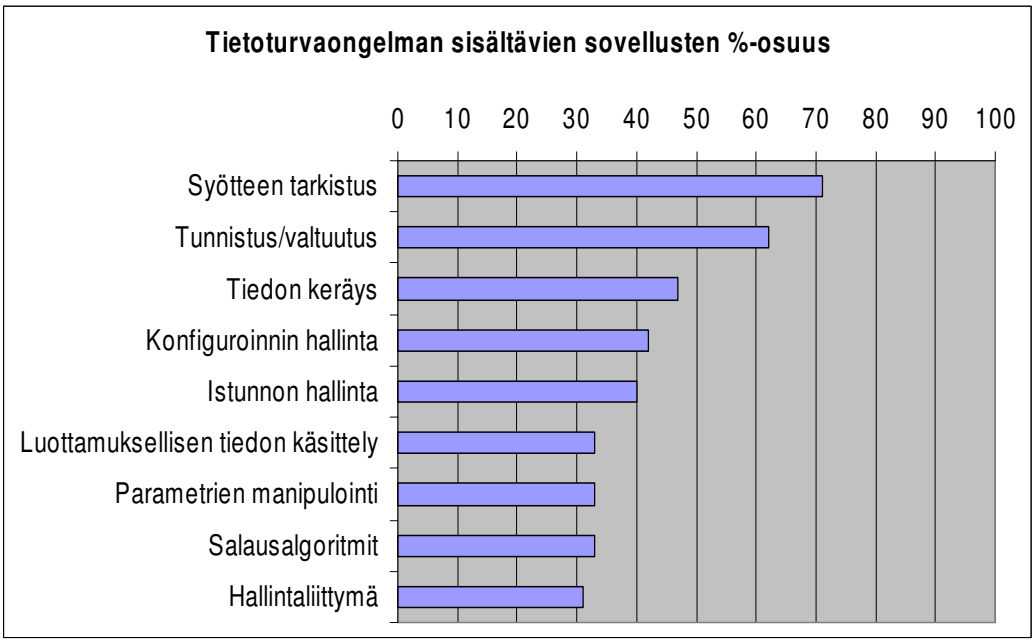
### 3.2 Tutkimuksia sovellustietoturvasta

Sovellusten tietoturvallisuudesta tehdyt tutkimukset ovat surullista luettavaa:

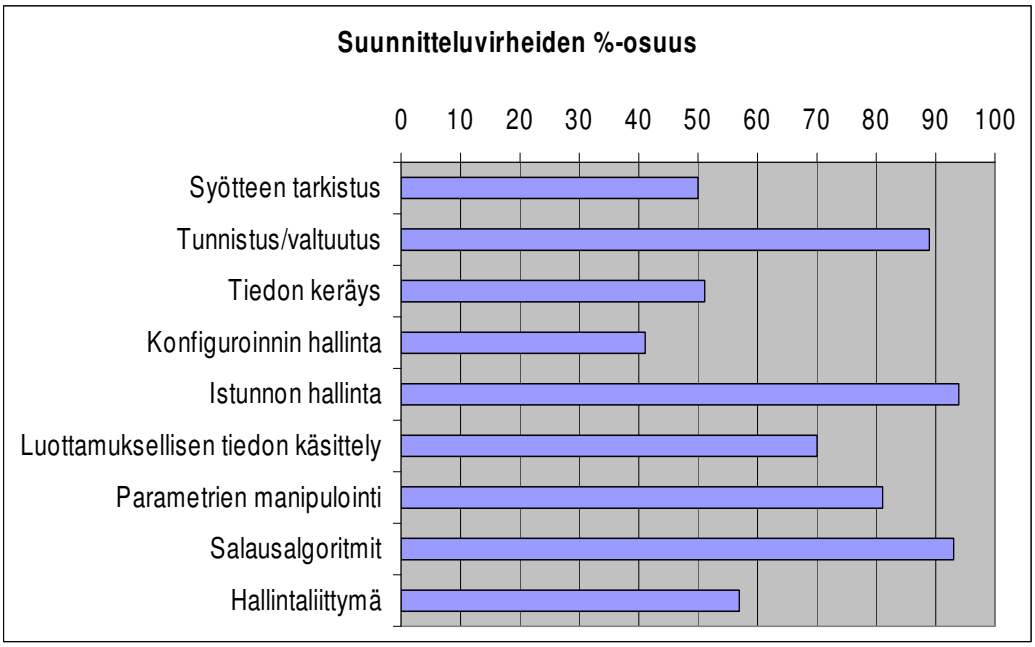
- Gartner ja Microsoft ovat arvioineet, että järjestelmien toimintakatkoksista 40% on sovellusvirheiden aiheuttamia. Gartner varoittaa lisäksi, että ohjelmisto-ongelmien aiheuttamat vuosittaiset häiriöajat kolminkertaistuvat 15 prosenttiin vuoteen 2008 mennessä niiden yritysten osalta, jotka eivät suhtaudu tietoturvallisuuteen ennaltaehkäisevästi rakentaessaan ja ostaessaan ohjelmistoja. [4]
- Carnegie Mellon University Software Engineering Institute arvioi sovelluksissa olevan keskimäärin 1-7 suunnittelu- tai toteutusvirhettä 1000 uutta tai muutettua koodiriviä kohti. [26]
- Microsoftin arvion mukaan 50% sovellusten tietoturvaongelmista on suunnitteluvirheitä. [26]
- CERT/CC arvioi, että 90% sovellusten tietoturvahaavoittuvuuksista johtuvat tunnetuista ongelmista ja 10 yleisintä syytä aiheuttavat 75% sovellusten haavoittuvuuksista. [26]

- @staken tutkimuksessa 45 merkittävästä sovelluksesta löytyi lähes 500 merkittävää tietoturvaongelmaa, keskimäärin 10 kpl per sovellus. Näistä suuri osa oli vakavia suunnitteluvirheitä.[1]

Edellä mainitusta @staken tutkimuksesta on huomioitava, että tutkitut sovellukset olivat yrityskriittisessä käytössä. Sovellukset olivat merkittävien ohjelmistotalojen kaupallisia tuotteita ja sähköisen kaupankäynnin sovelluksia. Seuraavassa kaaviossa esitetään prosenttiosuuksina, kuinka suuresta osasta sovelluksia löytyi tietyn ongelmaluokan tietoturvavirhe.




Näistä tutkimuksessa löydetyistä virheistä suunnitteluvirheiden osuus prosentteina oli ongelmaluokkakohtaisesti seuraava.




Tutkimuksessa havaittiin myös, että eri yritysten tekemien sovellusten välillä oli selkeät erot tietoturvatasossa. Tutkimuksessa arvioitiin, että hyvä tietoturvallisuus saavutettiin nimenomaan hyvien tietoturvakäytäntöjen kautta suunnittelussa, toteutuksessa ja käyttöönotossa. Työkaluvalikoimalla ei niinkään ollut merkitystä.

Tutkimus tunnisti erityisesti kuusi toimintamallia parhaiten suoriutuvilla yrityksillä:

1. Käyttäjän tunnistuksen, todennuksen ja valtuutuksen suunnittelu aikaisessa vaiheessa
2. Syötteiden huolellinen tarkistaminen ennen käyttöä
3. Tietojen salaaminen koko käsittelyketjun läpi
4. Tietojen huolellinen suojaaminen kaikkialla ja kaikissa vaiheissa
5. Oletusasetusten muuttaminen ja ylläpitovirheiden minimointi
6. Tietoturvallisuuden huomioiminen laatukriteerinä laadunvarmistusprosessissa

Web-sovellusten tietoturvallisuuden kehittämiseen keskittyvä, vapaaehtoisvoimin toimiva projekti, The Open Web Application Security Project (OWASP) ylläpitää listaa kymmenestä pahimmasta sovellusten tietoturvaongelmasta. Marraskuussa 2005 listalla oli pääosin vastaavat ongelmat kuin @staken tutkimuksessa havaitut [30].

Tutkimuksista voi tehdä johtopäätöksen, että systeemityössä ei huomioida tietoturvallisuutta kokonaisvaltaisesti. Vaikuttaa siltä, että sovellusten tietoturvaratkaisut jätetään usein ohjelmoijan tai sovelluksen ylläpitäjän vastuulle. Erityisen huolestuttavaa on, että yleisimmät ongelmat ovat tietoturvallisuuden perustoimintoja, kuten käyttäjän todennus ja valtuutus sekä istunnonhallinta. Samoin syötteen ja parametrien oikeellisuuden tarkastaminen tulisi olla rutiinia.

Monet listatuista ongelmista ovat olleet tiedossa vuosia, elleivät vuosikymmeniä, kuten puskuriylivuoto-ongelmat. Käytettävät tekniikat ja sovellusympäristö ovat kuitenkin samalla muuttuneet ja kehittyneet merkittävästi, joten ilmeisesti systeemityöammattilaiset ovat keskittyneet muihin heille tutumpiin haasteisiin kuin tietoturvallisuuteen. On tärkeää huomata, että samalla kun sovellusympäristö muuttuu, myös tietoturvavaatimukset muuttuvat. Nykyään tietoturvahaasteita tuovat sovellusten ja yritysten verkottuminen sekä sovellusympäristöjen monimutkaisuus ja laajennettavuus. Asiakkaiden, kumppaneiden ja oman henkilökunnan sovelluskäyttöä ei välttämättä voi enää eristää omiin järjestelmiinsä, vaan sovellusten tietoturvaratkaisujen täytyy mahdollistaa samojen järjestelmien käyttö eri välineillä, eri näkökulmista ja eri oikeuksilla.

### 3.3 Pankkisovellusten tietoturvavaatimukset

Pankkisovellusten tietoturvavaatimukset eivät juurikaan poikkea muiden elintärkeiden sovellusten vaatimuksista:

- **Käytettävyys**: varautuminen häiriöihin ja poikkeusoloihin täytyy huomioida sovelluksessa sen liiketoimintakriittisyyden perusteella. Sovelluksen täytyy olla häiriöitä sietävä.
- **Luottamuksellisuus**: pankkisalaisuuden toteutuminen on huomioitava sovelluksen tietojen käsittelyssä, tallentamisessa ja itse sovelluksen hallintamenettelyissä.
- **Eheys**: pankkijärjestelmiin ja sen käsittelemiin tietoihin on voitava luottaa. Sovelluksen on huolehdittava tietojen oikeellisuudesta ja tarvittavista eheystarkistuksista.

- **Kiistämättömyys, todistettavuus:** tehdyt toimenpiteet ja tapahtumat on pystyttävä osoittamaan tapahtuneiksi.
- **Jäljitettävyys:** pankkijärjestelmien tapahtumia on pystyttävä aukottomasti jäljittämään henkilötasolle asti.
- **Luotettavuus:** asiakkaille, kumppaneille ja viranomaisille on pystyttävä perustelemaan sovellusten laadukkuus ja tietoturvallisuus. Kattava, hyvin tietoturvallisuuden huomioiva sovellusten ja tietojärjestelmien systeemityömalli toimii tässä tukena.
- **Tietosuoja:** asiakkaiden ja toimihenkilöiden tietosuojasta on huolehdittava sovelluskäytössä ja ylläpidossa.

Pankkijärjestelmissä tietoturvallisuuden täytyy toteutua häiriöttömänä ja luotettavana liiketoimintana myös erityisolosuhteissa. Valtionhallinnolla on tahto turvata yhteiskunnan elintärkeät toiminnot. *Valmiuslaki* [48] edellyttää rahoituslaitoksilta tehtäviensä mahdollisimman häiriötöntä hoitamista myös poikkeusoloissa ennakkoon varautumalla ja valmiussuunnittelulla. *Laki Luottolaitostoiminnasta* [20] asettaa vahvat vaatimukset asiakkaiden tunnistamiselle ja pankkisalaisuuden säilymiselle. Suomen Pankkiyhdistys on julkaissut *pankkisalaisuusohjeet* [42].

Lakien toteutumista ohjataan usean eri tahon toimesta. Valtioneuvoston asettaman Puolustustaloudellisen suunnittelukunnan Rahoitushuoltotoimikunnan julkaisema *Rahoitusmarkkinoiden varautumisohje* [33] asettaa vaatimuksia poikkeusoloihin varautumiselle. Lähtökohtana on toiminnan jatkaminen ja palveluiden tarjonta normaalilla tavalla niin pitkään kuin mahdollista sekä nopea toipuminen vakavistakin häiriöistä. *Rahoitustarkastuksen standardi 4.4b, Operatiivisten riskien hallinta* [34] antaa määräyksiä ja ohjeita myös tietoturvallisuuteen, jatkuvuussuunnitteluun ja tietoturvallisten palveluiden rakentamiseen.

Tietoturvallisuuden toteutumista valvotaan sisäisen tarkastuksen ja Rahoitustarkastuksen toimesta. Myös sidosryhmät enenevässä määrin vaativat todennettua tietoturvallisuutta. Luottokorttiyhtiöt, kuten Visa ja Mastercard, vaativat luottokorttitietoja käsitte-

leviltä tahoilta kolmannen osapuolen tekemää tietoturva-tarkastusta, jossa arvioidaan käytössä olevien sovellusten ja luottokorttitietojen tietoturvallisuus [53].

Omat haasteensa tuovat *Henkilötietolaki* [8] ja *Sähköisen viestinnän tietosuojalaki* [44]. Tietosuojaa on vaalittava samalla kun tapahtumatietoa on kattavasti kerättävä tietojen oikeellisuuden ja tapahtumien kiistämättömyyden osoittamiseksi.

Tutkielman kannalta ei ole oleellista syventyä edellä mainittujen lakien, määräysten, ohjeiden ja vaatimusten yksityiskohtiin. Tavoitteena on pankkisovellusten häiriöttömyys ja tieturvallisuus lähes kaikissa olosuhteissa sekä kansalaisten luottamuksen säilyminen pankkitoimintaan. Systeemityössä tietoturvallisuuteen on suhtauduttava erityisellä huolellisuudella sen kaikissa vaiheissa sekä käytettävien menetelmien, työkalujen ja toimintatapojen on tuettava tietoturvallisuuden toteutumista.

## 3.4 Tietoturvastandardeja ja suosituksia

Seuraavassa arvioidaan lyhyesti muutamia tietoturvallisuuden standardeja ja ohjeita sekä niiden soveltuvuutta ohjaamaan tietoturvallisuuden huomioimista systeemityössä.

### 3.4.1 Comprehensive, Lightweight Application Security Process (CLASP)

CLASP sisältää 30 systeemityössä huomioitavaa tietoturvatoimenpidettä. Toimenpiteet on suunniteltu integroitaviksi olemassa olevaan systeemityömalliin siten, että organisaatio voi valita mukaan tarpeellisiksi katsomansa osiot. CLASP on kehitetty käytännön projekteissa ja siinä on kuvattu myös tietoturvallisuuteen liittyviä rooleja. [41]

CLASP perustuu seuraaviin hyviksi havaittuihin käytäntöihin:

- Sovellustietoturvatietoisuuden kehittämiseen
- Sovelluksen tietoturva-arviointiin
- Sovelluksen tietoturvavaatimusten varmistamiseen
- Systeemityön käytännön tietoturvaohjeista ja tietoturvatoimenpiteistä huolehtimiseen
- Tietoturvaongelmien korjaamisen kehittämiseen
- Tietoturvamittareiden kehittämiseen ja seuraamiseen

CLASP on hyödyllinen apuväline tietoturvallisuuden integroimiseksi systeemityömalliin.

### 3.4.2 Control Objectives for Information and related Technology (COBIT)

COBIT on IT-keskeinen malli IT-kontrollien suunnitteluun, käyttöönottoon ja tarkastamiseen. Malli selittää, miten organisaation toimintaan saadaan IT prosesseista tarpeellinen tieto arvioiden asiaa kriittisten menestystekijöiden, prosessien tavoiteindikaattorien ja prosessien suoritusindikaattorien suhteen. COBITin 34 IT-prosessia jaetaan neljään luokkaan: suunnittelu ja organisointi, hankinta ja toimeenpano, toimitus ja tuki sekä tarkastaminen. Jokaiselle prosessille on päävalvontatavoite ja sille alisteisia valvontatavoitteita. Valvontatavoitteiden saavuttamista arvioidaan kypsyystasomallia käyttäen. [14]

Ainoastaan yksi COBITin prosesseista on suoraan tietoturvallisuuteen liittyvä, mutta hankinta ja toimeenpano luokassa otetaan kantaa myös systeemityön tietoturvakontrolleihin mm. seuraavasti:

- Tietoturvavaatimukset täytyy löytää, hyväksyä ja dokumentoida projektin määrittelyvaiheessa. Tietoturvaratkaisujen täytyy olla oikein mitoitettuja.
- Jatkuvuusvaatimusten määrittelyllä varmistetaan ratkaisun oikea häiriöiden sietokyky.
- Systeemityömallin täytyy varmistaa tarvittavat jäljitysketjut.
- Ohjelmistoasennusten turvallisuus on varmistettava.

COBIT on tärkeä malli, koska sitä käytetään usein IT-tarkastuksen perustana. Se ei anna riittävästi apuvälineitä tietoturvallisuuden integroimiseksi systeemityömalliin, mutta systeemityömalli kannattaa joka tapauksessa arvioida COBITin vaatimuksia vasten.

### 3.4.3 ISF Standard of Good Practice for Information Security (ISF SoGP)

Information Security Forum on loppukäyttäjäorganisaatioiden järjestö, joka kehittää jäsenistön kokemuksiin perustuvaa tietoturvastandardia. ISF SoGP käsittelee kokonaisvaltaisesti ja käytännönläheisesti organisaation tietoturvallisuutta [13].

Standardi on jaettu viiteen luokkaan: tietoturvallisuuden hallinta, kriittiset liiketoimintasovellukset, tietojärjestelmäasennukset, tietoverkot sekä tietojärjestelmien kehittäminen.

Tietojärjestelmien kehittäminen on jaettu aliluokkiin ja niissä on tietoturvakannanotot seuraavasti:

- Kehittämisen hallinta: roolit ja vastuut, kehittämismalli, laadun varmistaminen, kehitysympäristö
- Tietoturvallisuuden hallinta: koordinointi, tietoturvatietoisuus, tietoturvallisuuden tarkastus
- Liiketoimintavaatimukset: tietoturvavaatimukset, luottamuksellisuusvaatimukset, eheysvaatimukset, jatkuvuusvaatimukset, tietoturvallisuuden riskiarvio
- Suunnittelu ja toteutus: tietojärjestelmän suunnittelu, sovelluskontrollit, yleiset tietoturvakontrollit, tuoteturvallisuus, toteutus, web-sovellukset
- Testaus: testiprosessi, hyväksymistestaus
- Käyttöönotto: käyttöönottovalmiuden testaus ja tarkastus, asennusprosessi, jälkitarkastus

ISF SoGP sopii hyvin systeemityön tietoturvatoimenpiteiden yleiseksi kehittämismalliksi. Ohjeet ovat kattavia ja käytännönläheisempiä kuin monessa muussa standardissa.

### 3.4.4 ISO/IEC 15408, Evaluation Criteria for Information Technology Security (Common Criteria, CC)

ISO/IEC standardi 15408 määrittelee tietoturvallisuuden arvioinnin kriteerit ja yhdistää USA:ssa (TCSEC), EU:ssa (ITSEC) ja Kanadassa (CTCPEC) kehitetyt arviointikriteeristöt. Standardi tunnetaan myös nimellä Common Criteria. [15].

Standardin mukaisesti tietojärjestelmille voidaan laatia tietoturvallisuusprofiili, joka kuvaa järjestelmän tietoturvallisuuden toiminnallisuuden ja vakuuttavuuden. Evaluoija voi tutkia täyttääkö tietojärjestelmä tietyn tietoturvallisuusprofiilin asettamat vaatimukset.

Standardi tarjoaa seitsemän tuotteiden tietoturvaluokkaa (EAL1-EAL7) sen mukaan, kuinka formaalisti ja menetelmällisesti tuotteen tietoturvallisuus on suunniteltu, testattu ja arvioitu.

Standardi on tarkoitettu tuotesertifiointiin, joten se ei suoraan sovellu systeemityömallin kehittämisen apuvälineeksi. Standardista löytyviä tietoturvatoiminnallisuuksia voi kuitenkin hyödyntää sovelluksen tietoturvavaatimuksia määritellessään.

### 3.4.5 ISO/IEC 17799:2005 ja ISO/IEC 27001:2005

Standardit antavat mallin organisaation tietoturvallisuuden hallinnalle ja kriteeristön tietoturvasertifioinnille.

17799 on ohje organisaation tietoturvastandardien kehittämiselle ja tietoturvallisuuden hallinnalle [16]. 27001 määrittelee vaatimukset tietoturvaratkaisuille ja tietoturvakäytännöille [18]. Organisaatio voi sertifioitua nimenomaan 27001 vaatimuksia vasten.

Standardeissa on oma osuutensa liittyen tietojärjestelmien hankintaan, kehitykseen ja ylläpitoon. Systeemityöhön liittyviä tietoturvakannanottoja ovat mm.:

- Tietoturvavaatimukset täytyy löytää, hyväksyä ja dokumentoida projektin määrittelyvaiheessa
- Syöttötietojen tarkastuksesta täytyy huolehtia
- Tietojen oikeellisuudesta täytyy varmistua
- Salausmekanismit ja avainten hallinta täytyy suunnitella huolellisesti
- Tuotantoasennusten täytyy olla kontrolloituja
- Testiaineiston täytyy olla kontrolloitu
- Lähdekoodin täytyy olla suojattu
- Ympäristön muuttuessa kriittisille sovelluksille täytyy tehdä tietoturva-arviointi
- Tietojen vuotaminen täytyy estää

ISO 17799 ja 27001 ovat tärkeitä standardeja, koska niitä käytetään organisaation tietoturva-arvioinnin ja – sertifioinnin perustana. Ne eivät anna käytännön apuvälineitä tie-

toturvallisuuden integroimiseksi systeemityömalliin, mutta systeemityömalli kannattaa joka tapauksessa arvioida näiden standardien vaatimuksia vasten.

### 3.4.6 ISO 21827, Systems Security Engineering Capability Maturity Model (SSE-CMM)

Malli sisältää menetelmäohjeistuksen, jonka avulla voidaan arvioida organisaation tietoturvallisuuden toteuttamistavan kypsyyttä järjestelmätuotantomenetelmässä. SSE-CMM mittaa organisaation tietoturvaprosesseja sekä normaaliin projektitoimintaan liittyviä prosesseja tuotteen tai järjestelmän koko elinkaaren ajan aina määrittelystä hallittuun alasajoon [17].

Standardi määrittää organisaatioille viisi kypsyystasoa sen mukaan, kuinka hyvin prosessit ovat toistettavissa, määriteltävissä, hallittavissa ja optimoitavissa.

SSE-CCM on työläs käytettävä systeemityömallin kehittämisessä, mutta kypsyystason mittaamiseen siitä voi saada työkaluja.

### 3.4.7 NIST Security Considerations in the Information System Development Life Cycle (800-64)

Ohje sisältää yleisiä tietoturvallisuusnäkökohtia tietojärjestelmän elinkaaren aikana. Ohjeessa korostetaan mm. seuraavia asioita [27]:

- Tietojärjestelmän liiketoiminnan vaikutusanalyysi tietoturvallisuuden kannalta
- Karkea riskiarviointi aikaisessa vaiheessa
- Perinpohjainen riskianalyysi suunnittelu ja toteutusvaiheessa
- Huolellinen tietoturvasuunnittelu huomioiden kustannustehokkuus
- Tietoturvaratkaisujen evaluointi ja testaus
- Tietoturvallisuuden tarkastus

Dokumentti sopii hyvin yleisohjeeksi ja taustalukemistoksi.

### 3.4.8 OWASP Guide to Building Secure Web Applications

OWASP (Open Web Application Security project) on Open Source projekti, joka kehittää ohjeita ja työkaluja Web-sovellusten tietoturvallisuuden parantamiseen. Projektin tuotokset ovat vapaasti käytettävissä ja toimintaa rahoitetaan mm. vapaaehtoisilla jä-

senmaksuilla. Esim. VISA tukee OWASP toimintaa ja viittaa omassa PCI-standardissaan OWASP-ohjeisiin.

Projektin tärkein tuotos on systeemityöhön osallistuville tarkoitettu tietoturvaohje: *A Guide to Building Secure Web Applications* [29]. Tämä dokumentti sisältää käytännönläheisiä ohjeita mm.

- Tietoturvalliseen ohjelmointiin
- Sovelluksen tietoturvauhkien mallinnukseen
- Henkilötietojen kalastelun estämiseen
- Luottokorttitietojen turvalliseen käsittelyyn
- Salaukseen
- Virheiden käsittelyyn
- Merkistöjen käsittelyyn
- Käyttäjien todennukseen ja valtuutukseen
- Istunnon hallintaan

OWASP tietoturvaohje sopii hyvin systeemityömallin käytännön toimenpiteiden kehittämiseen.

### 3.4.9 Rahoitustarkastuksen standardi 4.4b operatiivisten riskien hallinta

Standardi antaa määräyksiä Rahoitustarkastuksen valvottaville organisaatioille. Standardi antaa velvoitteita operatiivisten riskien hallinnan järjestämiseen sekä eri osa-alueille, kuten jatkuvuussuunnittelu, tietojärjestelmät, henkilöstö ja tietoturvallisuus [34].

Standardissa painotetaan tietoturvariskien tunnistamista ja hallintakeinojen dokumentointia sekä esitetään mm. vaatimus järjestelmäkehityksen ja tuotantotehtävien erottamisesta. Standardissa ei ole yksityiskohtaisia teknisiä vaatimuksia, vaan hyviä käytäntöjä ja periaatteita.

Standardin noudattaminen on vaatimus rahoitusalalla toimiville organisaatioille.

### 3.4.10 Valtionhallinnon keskeisten tietojärjestelmien turvaaminen

Dokumentti keskittyy erittäin vaativan tietoturvatason järjestelmien tietoturvallisuuteen kattaen kaikki tietoturvallisuuden osa-alueet. Dokumentin lähtökohtana oletetaan tietämys perustietoturvatason ja vaativan tietoturvatason tietojärjestelmien vaatimuksista [49].

Erityisesti on hyvä huomioida sovellustoimittajalle asetetut dokumentointivaatimukset tietoturvaratkaisujen hyvyyden todentamiseksi:

- Sovelluksen määrittely huomioiden organisaation tietoturvavaatimukset
- Tietoturvaratkaisut yksityiskohtaisesti
- Tietoturvaratkaisujen riskianalyysi
- Perustelut tietoturvaratkaisujen sopivuudesta organisaation esittämiin vaatimuksiin
- Tietoturvaratkaisujen testisuunnitelma ja – tulokset

Dokumentti toimii tarkistuslistatyyppisesti organisaation omien toimintojen vertaamisessa valtionhallinnon vaatimuksiin.

### 3.4.11 Valtionhallinnon tietojärjestelmäkehityksen tietoturvallisuussuositus

Suosituksen tarkoituksena on [50]:

- parantaa valtionhallinnon tietojärjestelmien tietoturvallisuuden tasoa yhtenäistämällä tietoturvallisuuden tavoitteiden määrittelyä ja toteuttamista järjestelmäkehityksen eri vaiheissa
- tukea valtion organisaatioita järjestelmäkehityshankkeiden läpivienteihin ja valmisohjelmistojen hankintoihin liittyvissä tietoturvallisuustehtävissä

Suosituksessa otetaan kantaa tietojärjestelmäkehityksen kaikkiin vaiheisiin, vaikka painopiste onkin tietojärjestelmän elinkaaren alkupäässä. Mukana on myös suosituksia tietoturvavastuiden työnjaosta, projektityön tietoturvallisuudesta, valmisohjelmien hankinnasta ja sopimuksiin liittyvistä tietoturvanäkökohdista..

Suositus kuvaa hyvin tietojärjestelmän elinkaaren vaiheet sekä kuhunkin vaiheeseen liittyvät tietoturvallisuustehtävät ja niiden päätulokset.

Suositus on käyttökelpoinen systeemityömallin kehittämisen tukena erityisesti tehtävä- ja tarkistuslistojen osalta.

### 3.4.12 Valtionhallinnon tietotekniikkahankintojen tietoturvallisuuden tarkistuslista

Ohje on tarkoitettu tietotekniikkatuotteiden ja -palvelujen ostajan tietoturvallisuuden tarkistuslistaksi sisältäen myös valmisohjelmistohankinnat ja sovelluskehityksen [51].

Ohje ei toimi systeemityömallin kehittämisen suorana tukena, mutta erityisesti valmisohjelmistohankintoihin ja sovelluskehitykseen liittyvät kysymykset on hyvä käydä läpi ja miettiä vastauksia oman organisaation kannalta. On oletettavaa – ja toivottavaa – että enenevässä määrin sovellusten, tietojärjestelmien ja palvelujen ostajien tietoturvatietoisuus kasvaa ja toimittajalta vaaditaan vakuudet ratkaisun tietoturvallisuudesta.

### 3.4.13 VISA Payment Card Industry Data Security Standard (PCI DSS)

VISA PCI DSS dokumentoi VISAn asettamat vaatimukset luottokorttitietoja vastaanottaville, käsitteleville, välittäville ja tallettaville tahoille. Tietojärjestelmät on tarkastettava säännöllisesti VISAn vaatimustenmukaisuuden noudattamiseksi. Pienillä toimijoilla riittää VISAlle toimitettava itsearviointi, isoilla toimijoilla vaaditaan VISAn nimeämän tahon tekemä kattava tarkastus.

Standardi koostuu kuudesta pääperiaatteesta [53]:

1. Rakenna ja ylläpidä tietoturvallinen tietoverkkoympäristö
2. Suojaa kortinhaltijan tiedot
3. Varmista sovellushaavoittuvuuksien hallinta
4. Käytä vahvoja pääsynhallintamekanismeja
5. Monitoroi ja testaa tietoverkkojen tietoturvallisuutta säännöllisesti
6. Ylläpidä tietoturvallisuusperiaatteita

Kolmas periaate, sovellushaavoittuvuuksien hallinta, edellyttää mm. tuotannon eristämistä testauksesta ja kehittämisestä sekä teknisesti että henkilötasolla, erillisen testiai-

neiston käyttöä, tietoturvatarkastusta ennen sovelluksen käyttöönottoa ja OWASP-projektin listaamien yleisimpien tietoturvaongelmien välttämistä.

Standardi ei sovi systeemityömallin tietoturvallisuuden kehittämiseen. Standardin noudattaminen on kuitenkin organisaatiollemme välttämättömyys, joten se toimii tarkistuslistana.

# 4 Tietoturvallisuus systeemityössä

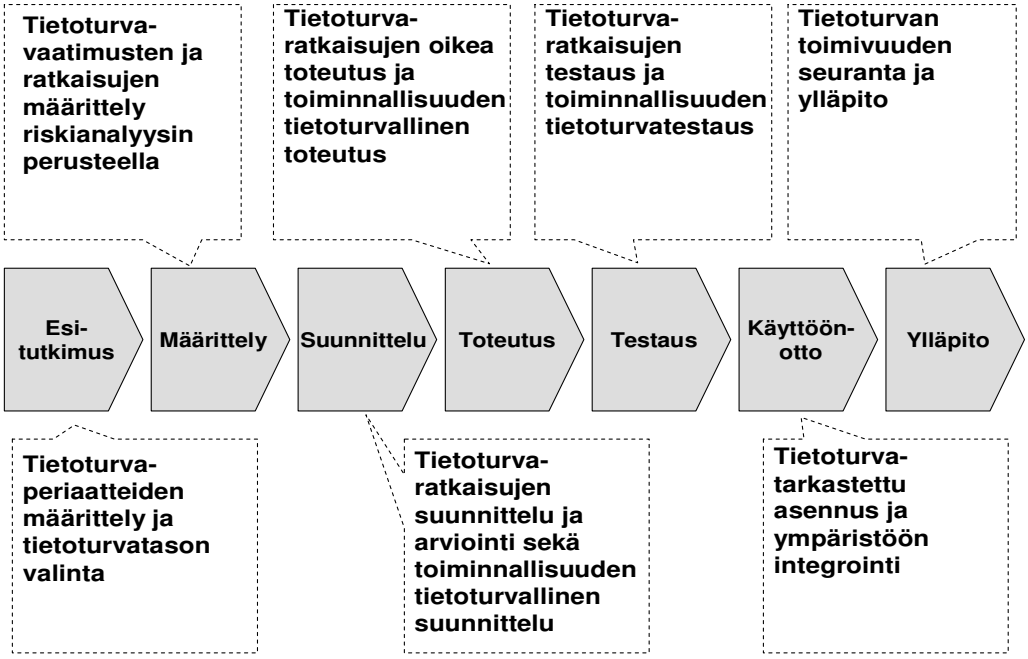
"*We already have enough fast, insecure systems. We don't need another one.*"

-- Ferguson, Schneier

## 4.1 Parhaat käytännöt systeemityön eri vaiheissa

Tässä luvussa esitellään lyhyesti yleisiä systeemityömallin parhaita käytäntöjä sovelluksen tietoturvallisuuden varmistamiseksi. Esiteltävät käytännöt ovat kirjallisuuden, tutkimusten ja kokemusteni kautta koottuja. Tarkoituksena on hahmottaa parhaat käytännöt ennen varsinaista tutkittavana olevan systeemityömallin arvioimista.

Päätason vaatimukset voidaan kuvata seuraavasti:




Usein systeemityömalliin liitetään myös vaihe *käytöstä poisto*. Tutkittavan systeemityömallin rakenteesta johtuen tämä tärkeä vaihe on jätetty pois myös tästä yleisestä mallista.

### 4.1.1 Esitutkimus

Esitutkimus kohdistuu liiketoimintatarpeisiin ja arvioi, miksi toimintaa pitäisi kehittää. Esitutkimuksessa selvitetään vaihtoehdot ja tehdään investointilaskelma arvioiden hyödyt ja kustannukset. Esitutkimuksessa täytyy varmistaa, että valittu etenemistapa on liiketoiminta- ja tietohallintostrategian mukainen.

**Tietoturvatehtävät:**

- Laaditaan liiketoiminnan vaikutusanalyysi. Analyysi osoittaa mahdolliset toimintaympäristön muutokset ja herättää huomaamaan kokonaisvaikutukset. Tietojärjestelmän tietoturvallisuus on aina suhteessa toimintaympäristöön.
- Arvioidaan jatkuvuus- ja toipumisaikavaatimukset. Korkeat häiriöttömyys- ja jatkuvuusvaatimukset vaikuttavat järjestelmän määrittelyyn ja suunnitteluun.
- Varmistetaan lakien, asetusten, määräysten ja ohjeiden asettamat vaatimukset.
- Varmistetaan sopimusten ja sidosryhmien asettamat vaatimukset.
- Varmistetaan organisaation tietoturvallisuusperiaatteiden ja – ohjeiden sekä tietoturva-arkkitehtuurin vaatimukset.
- Kartoitetaan tietoturvauhat ja arvioidaan karkeasti tietoturvariskit.
- Tietoturvariskiarvion perusteella määritetään järjestelmän tietoturvaperiaatteet ja haluttu tietoturvataso.

### 4.1.2 Määrittely

Määrittely kohdistuu tietojärjestelmään ja siinä arvioidaan ja dokumentoidaan mitä tietojärjestelmältä odotetaan. Dokumentoidaan tieto-, toiminto- ja tietoturvavaatimukset.

**Tietoturvatehtävät:**

- Tunnistetaan suojattavat tiedot ja kohteet
- Laaditaan tietojärjestelmän uhkamalli huomioiden myös väärinkäyttötapaukset.
- Laaditaan tietoturvariskianalyysi.
- Valitaan suojaukset ja kontrollit tietoturvariskianalyysin perusteella.

- Dokumentoidaan toimintaympäristön tietoturvallisuuteen vaikuttavat oletukset.
- Määritellään tietoturvavaatimukset selkeästi ja riittävän yksityiskohtaisesti niin, että ne voidaan suunnitella ja niiden toteutuminen varmistaa.

Kunnollinen tietoturvariskianalyysi on ehdottomasti tehtävä jo määrittelyvaiheessa, jotta suojaukset ja kontrollit voidaan valita oikein.

Määrittelyvaiheessa tehtyä tietoturvariskianalyysia on tarkennettava systeemityön edetessä. Valitut tietoturvaratkaisut voivat myös aiheuttaa uusia tietoturvariskejä.

### 4.1.3 Suunnittelu

Suunnitteluvaiheessa suunnitellaan ja dokumentoidaan, miten määrittelyn vaatimukset toteutetaan. Tässä vaiheessa suunnitellaan mm. käyttöliittymät, sovellusarkkitehtuuri, tietokantaratkaisu, tietoliikenneratkaisu, varusohjelmistot, liittymät, testaustarpeet ja tietoturvaratkaisut.

**Tietoturvatehtävät:**

- Varmistetaan organisaation tietoturva-arkkitehtuurin noudattaminen.
- Käytetään ratkaisuissa tunnettuja ja hyväksi havaittuja tietoturvamalleja.
- Noudatetaan tunnettuja tietoturvallisuuteen tähtääviä suunnitteluperiaatteita [52]:
    - Suojaa heikoin kohta ensin
    - Rakenna useita puolustuslinjoja
    - Turvaa virhetilanteet
    - Anna sovellukselle mahdollisimman vähän oikeuksia
    - Erottele toiminnallisuudet eri turvatasoihin
    - Pyri yksinkertaisiin ratkaisuihin
    - Huolehdi yksityisyyden suojasta
    - Älä luota salaisuuksien säilymiseen
    - Luota säästeliäästi
    - Käytä testattuja välineitä ja komponentteja

- Varmistetaan sovelluslogiikan tietoturvallisuus.
- Eristetään tietoturvakriittiset osiot siten, että niiden katselmointi on helppoa.
- Huomioidaan ratkaisuissa testattavuus.
- Tehdään suunnitelman uhka-analyysi, jossa esim. uhkapuiden avulla arvioidaan suunnitelman tietoturvaratkaisujen riittävyys.

### 4.1.4 Toteutus

Toteutusvaiheessa kirjoitetaan sovelluksen ohjelmakoodi. Ohjelmoijien ei tule tehdä ratkaisuja sovelluksen toiminnallisuudesta, arkkitehtuurista ja toteutustavasta. Ohjelmoijan tehtävä on toteuttaa annettu suunnitelma tehokkaasti ja tietoturvallisesti. Ohjelmointivaiheessa huomatut puutteet ja epäselvyydet on palautettava suunnittelu- ja mahdollisesti jopa määrittelyvaiheeseen.

**Tietoturvatehtävät:**

- Toteutuksessa tarvitaan tuote- ja työvälinekohtaista tietoturvaosaamista. Eri ohjelmointikielissä, varusohjelmistoissa, tietokannoissa ja työvälineissä on omat tietoturvallisuutta heikentävät ja parantavat piirteensä.
- Toteutuksessa on noudatettava organisaation ohjelmointistandardeja, joissa varmistetaan, että valittujen toteutusvälineiden ja ohjelmointikielien vaarallisia piirteitä ei käytetä.
- Tietoturvakriittiset osiot eristetään siten, että niiden testaaminen ja katselmointi on helppoa.
- Tietoturvaratkaisuihin liittyvä ohjelmakoodi katselmoidaan.
- Katselmoidaan muuhun toiminnallisuuteen liittyvä koodi ja varmistetaan tietoturvallinen toteutus. Ohjelmakoodia kirjoitettaessa ja katselmoidessa tarvitaan erilaisia näkökulmia: toiminnallisuuden toteuttaminen, virhetilanteiden käsittely ja tietoturvallisuuden varmistaminen. Kaikkien näkökulmien yhtäaikainen käsittely voi olla ylivoimaista, joten koodi on syytä katselmoida erikseen eri näkökulmista.

### 4.1.5 Testaus

Testausvaiheessa varmistetaan sovelluksen määritysten mukainen toimivuus. Testauksessa täytyy varmistaa myös sovelluksen selviytyminen kovassa kuormituksessa, virhetilanteissa ja väärinkäyttöyrityksissä.

**Tietoturvatehtävät:**

- Varmistetaan, että sovellus toteuttaa *vain ja ainoastaan* määrityksen mukaisen toiminnallisuuden ilman yllättäviä sivuvaikutuksia.
- Varmistetaan tietoturvavaatimusten toteutuminen sekä valittujen suojausten ja kontrollien virheetön toteutus.
- Testataan erityisesti komponenttien väliset rajapinnat.
- Käytetään tietoturvatestaukseen erikoistuneita työkaluja:
    - Ohjelmakoodianalysaattorit
    - Satunnaissyötteen generoijat
    - Tunnettuja ongelmia testaavat työkalut
- Toimitaan väärinkäytöstä yrittävän tavoin:
    - Rikotaan suunnittelijan tai ohjelmoijan olettamukset. Huijataan sovellusta.
    - Testataan sovelluksen normaalia rajapintaa alemmalla tasolla

Testaus on tärkeä vaihe sovelluksen toimivuuden ja turvallisuuden osoittamiseksi, mutta sovelluksen tietoturvallisuus ei saa perustua etsi-ja-korjaa periaatteeseen. Andersson esittää kirjassaan tilastollisen esimerkin, joka osoittaa yksittäisen henkilön hallitsevan isoa organisaatiota kilpailussa tietoturvavirheiden etsinnässä. Vaikka iso organisaatio suurilla resursseillaan löytää ja korjaa paljon enemmän virheitä, kuin yksittäinen henkilö löytää vapaa-ajallaan, niin silti vain pienellä todennäköisyydellä organisaatio löytää saman virheen kuin tämä henkilö [2].

Testauksen suunnittelussa voidaan käyttää apuna esim. Hoglundin ja McGrawn dokumentoimaa 49 hyökkäysmallia [9].

### 4.1.6 Käyttöönotto

Käyttöönotossa sovellus siirretään tuotantovastuullisille. Sovelluksen tarvitsemat laitteet varusohjelmistoineen asennetaan ja luodaan muu tarvittava käyttö- ja toimintaympäristö:

**Tietoturvatehtävät:**

- Tarkastetaan projektin luovutusdokumentaatio tietoturvavaatimusten osalta.
- Laaditaan asennussuunnitelmat ja -dokumentaatio huomioiden tietoturvanäkökohdat.
- Huolehditaan järjestelmien tietoturvakovennuksista.
- Asennetaan tietoturvakorjaukset käyttöjärjestelmään ja varusohjelmistoihin.
- Tarkastetaan varusohjelmistojen ja sovellusten asetusten oikeellisuus ja tietoturvallisuus.
- Tarkastetaan kokonaisuuden tietoturvallisuus ennen käyttöönottoa. Apuna käytetään tarkistuslistoja ja tehtävään tarkoitettuja työkaluja. Tarkastuksessa on hyvä käyttää myös ulkopuolisia asiantuntijoita.

### 4.1.7 Ylläpito

Sovelluksen ollessa ylläpidossa varmistetaan kapasiteetin riittävyys ja huolehditaan tarvittavista päivityksistä.

**Tietoturvatehtävät:**

- Seurataan tietoturvallisuuden toteutumista.
- Päivitetään sovellusmuutosten yhteydessä määrittely ja suunnitteludokumentaatio.
- Ylläpidetään tietoturvariskianalyysiä ympäristön, käyttäjäkunnan, käyttötavan tai toiminnallisuuden muuttuessa.
- Seurataan tietoturvakorjausten ilmestymistä sekä asennetaan tarvittavat korjaukset ja päivitykset.
- Pidetään kirjaa muutoksista.

- Varaudutaan tietoturvapoikkeamien käsittelyyn.

## 4.2 Sovelluksen tietoturvariskien hallinta

Sovelluksen tietoturvaratkaisujen oikeaksi mitoittamiseksi sovelluksen tietoturvariskit on arvioitava ja analysoitava. Tietoturvariskien arviointi tarkoittaa toimenpidettä, jolla pyritään tunnistamaan sovellukseen tai sovelluksen kautta organisaatioon kohdistuvat uhat ja niiden mahdolliset vaikutukset karkealla tasolla. Sovelluksen tietoturvariskianalyysi on yksityiskohtainen tutkintamenetelmä, jolla selvitetään sovellukseen kohdistuvat yksittäiset, tekniset uhat. Tunnistettujen ja arvioitujen tietoturvariskien perusteella voidaan päättää sopivista hallintakeinoista, tietoturvatoimenpiteistä ja – ratkaisuista [25].

Sovelluksen tietoturvariski voidaan yleisesti kuvata kaavalla

$$Riski = \left[(Uhka \times Haavoittuvuus) \div Suojaustoimenpiteet\right] \times Vaikutus$$

missä

*Uhka* = Sovellukseen kohdistuva vaara vai väärinkäytösmahdollisuus

*Haavoittuvuus* = Sovelluksen tai sovellusympäristön heikkous, joka mahdollistaa uhkan toteutumisen

*Suojaustoimenpiteet* = Toimenpiteet, suojaukset ja kontrollit, joilla pienennetään Uhkaa tai suojaudutaan haavoittuvuudelta

*Vaikutus* = Uhkan toteutumisen vaikutus, esim. rahan, asiakkaiden tai maineen menetys

Käytännössä suojaustoimenpiteillä pienennetään uhkan toteutumisen todennäköisyyttä, joten voidaan käyttää myös kaavaa

$$Riski = Uhka \times Todennäköisyys \times Vaikutus$$

missä

*Todennäköisyys* = Uhkan toteutumisen mahdollisuus huomioiden tehdyt suojaustoimenpiteet

Tyypillisesti tietoturvariskien arvioinnissa käytetään organisaatiossa sovittua luokitusta. Esimerkiksi todennäköisyys arvioidaan luokituksella *todennäköinen – mahdollinen - epätodennäköinen* ja vaikutus luokitellaan vastaavasti *suuri – keskimääräinen – pieni*.

Löydetyt tietoturvariskit on pyrittävä poistamaan, minimoimaan tai siirtämään. Tietoturvariskien luokittelu ohjaa toimenpiteiden priorisoinnissa. Jäljelle jääviä, hyväksyttäviä ja hallittavia riskejä kutsutaan jäännösriskiksi..

Systeemityöhön kuuluva tietoturvariskien arviointi sekoitetaan usein organisaatiossa tehtävään muuhun riskien arviointityöhön, kuten projektiriskien arviointiin. Systeemityössä arvioidaan ja analysoidaan tietoturvariskejä systeemityön kohteena olevan sovelluksen kannalta esim. seuraavasti:

- Esitutkimusvaiheessa tunnistetaan uhat sekä arvioidaan karkeasti kunkin uhkan todennäköisyys ja toteutumisen vaikutukset tietoturvatavoitteiden määrittämiseksi.
- Määrittelyvaiheessa tehdään perusteellinen tietoturvariskianalyysi selkeiden tietoturvavaatimusten saamiseksi.
- Suunnitteluvaiheessa tietoturvariskianalyysia tarkennetaan teknisen suunnitelman tasolle huomioiden esim. suunnitellut tuotteet, rajapinnat, protokollat, verkkoarkkitehtuurit, liittymät, jne.
- Testaustulosten perusteella tarkistetaan tietoturvariskianalyysia.
- Tietoturvariskianalyysi ylläpidetään sovelluksen koko elinkaaren ajan huomioiden sovelluksen kehityspiirteiden ja sovellusympäristön muutosten vaikutukset.

Tietoturvariskien dokumentoinnissa kannattaa käyttää organisaation yleistä riskien dokumentointitapaa, mutta tietoturvariskien tunnistamisessa ja arvioinnissa voi hyödyntää erityisesti tähän suunniteltuja malleja. Microsoft esittää STRIDE/DREAD mallin, jossa uhat sijoitetaan kuuteen luokkaan (S-T-R-I-D-E) ja kutakin uhkaa arvioidaan viidestä eri näkökulmasta (D-R-E-A-D) määritellen niille vakavuus asteikolla 1-10 [43].

| Uhkien luokittelu (STRIDE) | Näkökulmat (DREAD) |
|---|---|
| **S**poofing<br>- harhauttaminen, tekeytyminen | **D**amage potential [1-10]<br>- vahinkopotentiaali |
| **T**ampering<br>- väärentäminen, tiedon peukalonti | **R**eproducibility [1-10]<br>- uusittavuus, toimivuus |
| **R**epudiation<br>- kiistäminen | **E**xploitability [1-10]<br>- hyödynnettävyys, työmäärä |
| **I**nformation disclosure<br>- tiedon paljastuminen | **A**ffected users [1-10]<br>- käyttäjävaikutus, laajuus |
| **D**enial of service<br>- palvelun esto | **D**iscoverability [1-10]<br>- havaittavuus |
| **E**levation of privilege<br>- käyttöoikeustason luvaton nosto | |

Tietoturvauhkien tunnistamiseen voidaan käyttää myös uhkapuita [39] ja erityisesti rajapintoihin liittyvien tietoturvauhkien tunnistamiseen voidaan käyttää edellä mainittua STRIDE/DREAD-mallia tietovirtakaavioon yhdistettynä.

Tietoturvariskianalyysin jatkuva ylläpito systeemityön edetessä on tärkeää myös siksi, että tietyn tietoturvariskin hallitsemiseksi valittu tietoturvaratkaisu voi aiheuttaa uusia riskejä. Tietoturvariskianalyysin vaiheita on yleisesti kuvattu myös seuraavasti [38]:

1. Mitä tietoja halutaan suojata?
2. Mitä tietoturvariskejä tietoihin kohdistuu?
3. Millä ratkaisuilla tietoturvariskejä pienennetään?
4. Kuinka hyvin valitut ratkaisut pienentävät tietoturvariskiä?
5. Mitä uusia tietoturvariskejä ratkaisut aiheuttavat?
6. Mitä kompromisseja valitut tietoturvaratkaisut edellyttävät?

# 5 Tutkittavan systeemityömallin arviointi ja parannusehdotukset

*"There are two ways of constructing a software design. One way is to make it so simple that there are obviously no deficiencies. And the other way is to make it so complicated that there are no obvious deficiencies."*
*--Tony Hoare*

## 5.1 Systeemityömallin yleiskuvaus

Arvioitava systeemityömalli sisältää liiketoiminnan kehittämisen, sovelluskehityksen ja arkkitehtuurin kehittämisen. Tutkielmassa keskityn sovelluskehitykseen liittyvään osuuteen. Sovelluksen sovittaminen olemassa olevaan tietoturva-arkkitehtuurin on tärkeää, joten myös se huomioidaan, vaikka itse arkkitehtuurin kehittämisen ohjeistukseen ei otetakaan kantaa

Systeemityömallin systemaattinen noudattaminen parantaa sovellustyön ja sovellusten laatua. Laadukkuuteen ja virheettömyyteen tähtäävä systeemityö parantaa myös sovellusten tietoturvallisuutta, vaikka se ei olisikaan fokuksena. Tämän tutkielman tarkoituksena on kuitenkin painottaa erityisesti tietoturvakriittisten sovellusten tarpeita ja tietoturvallisuutta parantavia erityiskeinoja.

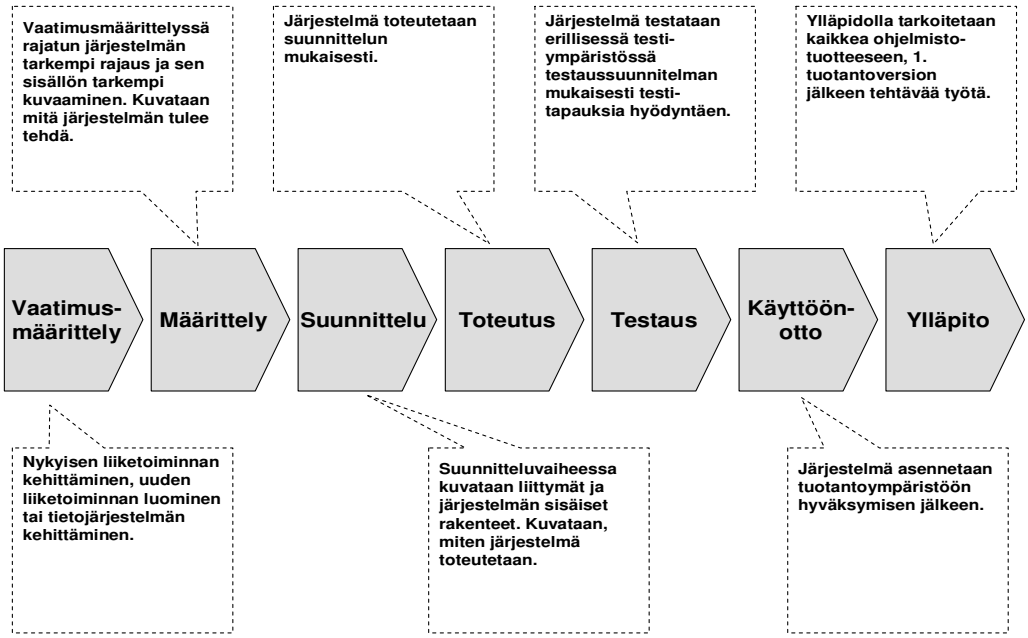
Systeemityömallin dokumentaatio sisältää varsinaisen ohjeistuksen ja dokumenttipohjien lisäksi myös koulutus- ja esittelymateriaalia. Systeemityömalli on kehitetty erityisesti oliopohjaisen sovelluskehityksen tarpeisiin, painottaen Java, J2EE ja web-sovelluksia. Mallinnukseen käytetään UML-kuvaustapaa (Unified Modeling Language). Dokumentaatio sisältää myös ohjeita Rational Rose ja SODA – työkalujen käyttöön UML-mallinnuksessa.

Esittelymateriaalissa tietoturvallisuus tuodaan esille tärkeänä tekijänä mm. mainiten, että tietoturvallisuus on yksi laadun kivijaloista toiminnallisuuden, luotettavuuden ja suorituskyvyn ohella. Esittelymateriaalissa korostetaan myös liiketoiminnan asettamien tietoturvallisuuden erityisvaatimusten huomioimista.

Systeemityömalli jakaa sovelluskehitysprosessin tehtäviin perinteisellä vaihejaolla. Esitutkimus-termiä ei kuitenkaan käytetä, vaan ensimmäisenä vaiheena on vaatimusmäärit-

tely. Vaatimusmäärittelyä oletetaan edeltävän liiketoiminnan kehittämisprosessi, jossa on tehty tietojärjestelmän esiselvitys.

Seuraavassa kuvassa esitetään systeemityömallin päätehtävät vaiheittain:




Malli sisältää kustakin vaiheesta ohjeet, dokumenttipohjia ja esimerkkejä. Kustakin vaiheesta on erillinen prosessikuvaus, joka kuvaa vaiheen tehtävät. Kustakin tehtävästä kuvataan:

- roolit ja vastuut
- tarvittava lähtödokumentaatio
- lopputulokset
- suositeltavat menetelmät
- tarvittavat ohjeet
- suositeltavat työvälineet
- mittarit

Tutkielman rajauksen kannalta ei ole tarkoituksenmukaista tutustua kaikkeen materiaaliin perinpohjaisesti. Lähtökohtana pidän prosessikuvauksia ja niissä mainittuja ohjeita, joista saan riittävän kuvan systeemityömallin tietoturvallisuuteen liittyvistä tehtävistä.

Kuvaan ja arvioin systeemityövaiheet tietoturvanäkökulmasta. Lisäksi teen päätelmät tietoturvatehtävien kattavuudesta ja annan kehitysehdotuksia tietoturvallisuuden kannalta.

Systeemityömalli sisältää myös erillisen tietoturvaohjeen joka arvioidaan erikseen.

Kussakin vaiheessa prosessikuvaus viittaa ohjeisiin, dokumentteihin ja tarkistuslistoihin, jotka on sijoitettu toisen systeemityövaiheen dokumenttiosioon. Pääasiassa käsittelen dokumentit siihen liittyvässä vaiheessa. Esim. määrittelyvaiheen testauksen suunnittelussa viitataan testausvaiheen dokumentaatioon, jonka käsittelen testausvaiheen tarkastelussa.

## 5.2 Vaatimusmäärittely

### 5.2.1 Kuvaus

Vaatimusmäärittelyä oletetaan edeltävän liiketoiminnan tai nykyisen järjestelmän kehittämis- ja/tai kannattavuustarkastelu, jonka perusteella päätetään lähteä sovelluskehitysprojektiin systeemityömallin mukaisesti.

Vaatimusmäärittelyn vaiheet:

1. Valmistelu: varmistetaan ja tarvittaessa täydennetään riittävät lähtötiedot
2. Vaatimusten määrittely: nykyongelmat ja kehitystarpeet sekä järjestelmävaatimukset
3. Toiminnallisuuden kartoitus: kuvataan järjestelmäliittymät sekä kartoitetaan käyttäjät, toiminnot ja tietosisältö
4. Arkkitehtuurin valinta: järjestelmä- ja integraatioarkkitehtuuri
5. Konversiotarpeiden kartoitus
6. Jatkokehittämisen suunnittelu: määrittelykokonaisuuksien valinta
7. Lopputulosten katselmointi

Tietoturvakannanotot prosessikuvauksessa:

- Tietoturva-asiantuntija avustaa järjestelmä- ja integraatioarkkitehtuurien valinnassa. Oletuksena on, että arkkitehtuurivaihtoehdot ovat jo olemassa ja kuvattu.

Tietoturvakannanotot ohjeissa:

- Vaatimusesimerkkeinä: tiedon eheysvaatimukset, jatkuvuusvaatimukset, käyttötilanteiden suojaustaso, hallintakäytön suojausmenettely, varmuuskopiointi, turvallisuusrikkomusten käsittely, virhetilanteisiin reagointi
- Käyttötapaukset ja roolit
- Arkkitehtuuri järjestelmänhallinnan osalta: esimerkkeinä valvonta, käytettävyysvaatimukset (24x7), toipuminen ennakoiduissa virhetilanteissa
- Varusohjelmisto- ja laitetason tietoturva-arkkitehtuuri: esimerkkeinä tietoliikenteen salaaminen, palomuuri, käyttäjien tunnistus ja todennus, järjestelmätason erityisvaatimukset
- Sovellustason tietoturva-arkkitehtuuri: käyttäjähallinta, istunnonhallinta, pääsynvalvonta, turvalokimekanismit, turvarikkeiden havainnointi ja niihin reagointi
- ”protokollatason tietoturvaa ei välttämättä tarvita luotetussa lähiverkkoympäristössä, palomuuriratkaisu on yleensä tarpeen vain Internet-ympäristössä”

Tietoturvakannanotot dokumenttipohjissa:

- Ei mainintoja

Tietoturvakannanotot tarkistuslistoissa:

- Käyttäjäroolien kuvaus

### 5.2.2 Arviointi

Tietoturvavaatimukset otetaan esille esimerkkeinä, eikä formaalia tapaa tunnistaa tietoturvariskejä ja -vaatimuksia ole esitetty. Erityisesti järjestelmän tietoturvariskiarvion puuttuminen vie pohjan tietoturvallisuuden oikeantasoiselta toteutumiselta.

Arkkitehtuuriosiossa huomioidaan verkkotason ja käyttöjärjestelmätason suojaukset. Normaalien pääsynvalvontavaatimusten (käyttäjien tunnistus, todennus, istunnonhallinta, käyttöoikeudet) lisäksi kiinnitetään huomiota myös käyttäjien hallintaan, turvalokeihin ja turvarikkeiden havainnointiin.

Tietoturva-asiantuntija on mukana vain arkkitehtuureihin liittyvissä tehtävissä.

Dokumentaation lause ”protokollatason tietoturvaa ei välttämättä tarvita luotetussa lähiverkkoympäristössä, palomuuriratkaisu on yleensä tarpeen vain Internet-ympäristössä” ei pidä paikkaansa nykyaikaisessa, tietoturvakriittisessä ympäristössä. Harvassa ympäristössä halutaan antaa kaikkea tietoa kaikkien saataville, jolloin verkkoympäristön lohkominen, palvelujen eristäminen ja ”luotetun verkon” tietoliikenteen salaaminen ovat tarpeellisia suojauskeinoja.

### 5.2.3 Parannusehdotuksia

Mikäli vaatimusmäärittelyvaihetta edeltänyt esitutkimus/kehittämistarkastelu ei ole tuottanut liiketoiminnan vaikutusanalyysia tai siinä ei ole huomioitu tietoturvavaikutuksia, täytyy vaikutusanalyysi tehdä. Joka tapauksessa järjestelmän kriittisyys organisaation liiketoiminnalle on arvioitava organisaation omien kriteerien mukaan, koska kriittisyys vaikuttaa merkittävästi järjestelmälle asetettaviin vaatimuksiin.

Menetelmä tarvitsee mallin ja ohjeet järjestelmän jatkuvuus- ja toipumisvaatimusten arviointiin sekä tietoturvariskiarvion laatimiseen. Dokumenttipohjassa täytyy olla selkeät paikat em. vaatimusten kirjaamiselle.

Lakien ja ohjeiden lisäksi nykyään myös sidosryhmät asettavat tietoturvavaatimuksia. Dokumenttien täytyy ohjata selvittämään ja täyttämään myös tarvittavat ulkopuoliset vaatimukset. Dokumenteissa täytyy myös viitata organisaation omiin tietoturvallisuusperiaatteisiin ja – ohjeisiin, jotta varmistetaan järjestelmän yhteensopivuus organisaation omiin vaatimuksiin.

ISO/IEC 17799:2005 standardi tukee edellä mainittuja vaatimuksia todetessaan, että tietoturvavaatimukset saadaan kolmesta päälähteestä [16]:

1. Riskianalyysista
2. Lakien, määräysten, sopimusten, sidosryhmien, tms. asettamista vaatimuksista

3. Organisaation omista periaatteista, toimintatavoista ja vaatimuksista

Vaiheen tuloksena täytyy saada selkeä kuvaus järjestelmän kriittisyydestä, jatkuvuusvaatimuksista, tietoturvariskeistä ja tietoturvaperiaatteista, jotka ohjaavat koko järjestelmän suunnittelua ja toteuttamista.

Tietoturvaosaamista tarvitaan ainakin hallinnollisen tietoturvallisuuden, sidosryhmävaatimusten ja tietoturva-arkkitehtuurin osalta.

## 5.3 Määrittely

### 5.3.1 Kuvaus

Määrittelyvaiheessa rajataan ja kuvataan rakennettava järjestelmä ja arkkitehtuuri sillä tarkkuudella, että tuloksia voidaan käyttää suunnittelun ja toteutuksen pohjana. Vaatimusmäärittelyssä ratkaisua vasta hahmotellaan painottaen liiketoimintavaatimuksia ja määrittelyvaiheessa tavoitteet konkretisoidaan. Tekniset ratkaisut valitaan vasta seuraavassa suunnitteluvaiheessa.

Määrittelyn vaiheet:

1. Arkkitehtuurien tarkentaminen
2. Toiminnallinen määrittely: toiminnallisuus, tietosisältö, ulkoiset liittymät, käyttöliittymät, tulosteet ja määrittelyn mallien integrointi
3. Jatkokehittämisen suunnittelu: alustava toteutussuunnitelma sekä testauksen ja käyttöönoton suunnittelu
4. Lopputulosten katselmointi

Tietoturvakannanotot prosessikuvauksessa:

- Tietoturva-asiantuntija avustaa järjestelmä- ja integraatioarkkitehtuurien suunnittelussa
- Tuotetaan käyttövaltuusmatriisi

Tietoturvakannanotot ohjeissa:

- Arkkitehtuurin osalta samat kuin vaatimusmäärittelyvaiheessa

- Käyttötapausten kuvauksessa mukana poikkeustilanteiden (virheiden) käsittely ja ei-toiminnallisissa vaatimuksissa mainitaan tietoturvallisuus ja käyttöoikeudet

Tietoturvakannanotot dokumenttipohjissa:

- Järjestelmäarkkitehtuuri sisältää otsikot tietoturva-arkkitehtuuri, protokollatason tietoturva, palomuuri, käyttäjien todennus, erityisvaatimukset käyttöjärjestelmätason tietoturvalle. Lisäksi huomioidaan yleisellä tasolla lokit, skaalautuvuus, valvonta ja toipumistekniikat.

Tietoturvakannanotot tarkistuslistoissa:

- Käyttäjäroolit, käyttötapaukset

### 5.3.2 Arviointi

Vaihe oikeastaan vain tarkentaa vaatimusmäärittelyvaihetta, joten edellisen vaiheen kommentit pätevät myös tähän.

Tietoturvallisuutta edistävät käyttötapausten kuvausmallissa olevat kohdat poikkeustilanteiden käsittelylle ja ei-toiminnallisissa vaatimuksissa mainituille tietoturva- ja käyttöoikeusvaatimuksille.

### 5.3.3 Parannusehdotuksia

Jatkuvuus- ja toipumisvaatimuksia tarkennetaan sekä tehdään kattava tietoturvariskianalyysi edellisen vaiheen riskiarvion pohjalta. Tietoturvariskianalyysin perusteella dokumentoidaan selkeät tietoturvavaatimukset. Tietoturvariskianalyysi ja tietoturvavaatimukset täytyy formaalisti hyväksyä. Yritysjohdon, projektin johtoryhmän tai tietoturvallisuudesta vastaavien on käytännössä mahdotonta ottaa kantaa toteutettavan tietojärjestelmän tietoturvatasoon ilman heille esitettyä selkeää riskianalyysia ja siitä johdettua vaatimusmäärittelyä.

Vaiheen tuloksena saadaan tarkempi tietoturvariskianalyysi, jossa myös esitetään tietoturvavaatimuksia tietoturvariskien pienentämiseksi. Tietoturvaperiaatteet konkretisoituvat tietoturvavaatimuksiksi.

Organisaation tietoturva-arkkitehtuurin ja – käytäntöjen asettamat vaatimukset järjestelmäarkkitehtuurille täytyy dokumentoida, esim. keskitetty käyttäjähakemisto, keski-

tetty valtuushallinta, sallitut todennustavat, verkkojen luokittelu palveluiden kriittisyyden mukaan, sallitut tietoliikenneprotokollat, salausvaatimukset, jäljitettävyysvaatimukset, integrointivaatimukset tietoturvallisuuden hallintaprosesseihin, jne. Mahdolliset poikkeamat täytyy hyväksyttää ja dokumentoida.

Laaditaan väärinkäyttötapauksia, joissa hahmotellaan, miten järjestelmää voitaisiin väärinkäyttää ja miten väärinkäytöksiltä suojaudutaan.

Dokumentoidaan tietoturvallisuuteen merkittävästi vaikuttavat oletukset, esim. ylläpitohenkilöstöön luotetaan, tiettyyn verkko-osioon luotetaan, jne. Tämä on tärkeää, jotta ympäristön mahdollisesti myöhemmin muuttuessa tiedetään ratkaisun vahvuudet ja heikkoudet.

Tietoturvaosaamista tarvitaan ainakin hallinnollisen tietoturvallisuuden, tietoturva-arkkitehtuurin ja tietoverkkojen osalta.

## 5.4 Suunnittelu

### 5.4.1 Kuvaus

Suunnitteluvaiheessa kuvataan, miten järjestelmä toteutetaan annettujen vaatimusten mukaisesti.

Suunnittelun vaiheet:

1. Arkkitehtuuritason suunnittelu: määrittelyn arviointi ja tarkentaminen, sovellusarkkitehtuurin ja sovellusrungon kuvaaminen
2. Yksityiskohtainen suunnittelu: oliomalli, ulkoiset liittymät, integraatioratkaisut, eräajot, käyttöliittymät, tulosteet, tietokanta, sovelluksen käyttöympäristö sekä tietoturvallisuus ja sen valvonta
3. Toteutustyön suunnittelu: toteutuserät, testaus, tietokonversiot, suunnittelumallien integrointi ja projektisuunnitelman täydentäminen
4. Lopputulosten tarkastus

Tietoturvakannanotot prosessikuvauksessa:

- Tietoturva-asiantuntija avustaa sovellusarkkitehtuurin kuvaamisessa

- Tietoturva-asiantuntija avustaa tietoturvaratkaisujen ja valvonnan suunnittelussa
- Tietokannan suojaukset ja käyttöoikeudet sekä tietokannan varmistaminen ja palauttaminen
- Järjestelmän tietoturvaratkaisujen kuvaus: verkkotopologia, sovellustyypit, tietoturvamenetelmät, tunnistus, todennus, valtuutus, salaaminen, allekirjoitus, tietoturvamallit (patterns), Java-ohjelmointikielen vaikutukset, tietoturvallisuuden valvonta ja raportointi
- Tietovarastojen varmistukset, tietoliikenteen varmistukset ja varajärjestelmien käyttö
- Jäljitysketjun (audit trail) kuvaus

Tietoturvakannanotot ohjeissa:

- Tietoturvallisuuden ja valvonnan suunnittelu perustuu erilliseen tietoturvaohjeeseen, joka on arvioitu erikseen.
- Sovellusintegraatiossa: fyysisten ja loogisten virhetilanteiden hallinta, eheysvaatimukset
- J2EE-referenssiarkkitehtuuri esittelee J2EE-maailman vaihtoehtoja käyttäjän todennukseen, istunnonhallintaan, syötteen tarkistukseen, pääsynvalvontaan, virheiden käsittelyyn, lokin keräämiseen ja skaalautuvuuteen.
- Käyttöympäristön kuvauksessa tietoturvaratkaisut ovat omana kohtanaan, tosin esimerkkinä vain tunnistus, todennus ja valtuutus. Samoin oma kohtansa löytyy varajärjestelmälle ja jäljitysketjulle
- Tietokannan suunnitteluohje huomioi tietokannan suojaukset, käyttöoikeudet, varmistukset ja jatkuvuutta parantavat mahdollisuudet
- Web-istunnon hallinnasta on oma ohje ja esimerkki

Tietoturvakannanotot dokumenttipohjissa:

- Käyttöympäristön kuvauksessa on otsikko tietoturvaratkaisuille, toipumismekanismeille, varajärjestelmälle, varmistuksille ja jäljitysketjulle

- Sovellusarkkitehtuurin kuvauksessa on otsikko käyttäjän todennukselle, istuntomekanismille, syötteen tarkistukselle, valtuutukselle ja sovellustason tietoturvalle
- Tietokantakuvaus sisältää otsikot em. tietokannan suunnitteluohjeen mukaisesti
- Suunnittelupohja esittää tehtäväksi erillisen tietoturvadokumentin tietoturvaohjeen mukaisesti

Tietoturvakannanotot tarkistuslistoissa:

- Tietokanta-tarkistuslista huomioi tietokannan suunnitteluohjeessa esitetyt asiat

### 5.4.2 Arviointi

Tietokannan ja käyttöympäristön tietoturvallisuus huomioitiin hyvin.

Tietoturvasuunnittelu keskittyi erillisen tietoturva-ohjeen noudattamiseen, joka käsitellään erikseen.

Yleisimpiä tietoturvanäkökohtia on huomioitu, mutta apuvälineitä tietoturvasuunnitteluun ei ole. Samoin määrittelyvaiheen tietoturvariskianalyysin puuttuminen aiheuttaa sen, että selkeitä perusteita tietoturvaratkaisujen ja kontrollien valinnalle ei ole.

### 5.4.3 Parannusehdotuksia

Valitaan tietoturvaratkaisut ja kontrollit tietoturvariskianalyysin avulla saatujen tietoturvavaatimusten perusteella. Tietoturvariskianalyysia edelleen tarkennetaan käsittämään suunnitteluvaiheen tulokset.

Kuvataan tunnetut tietoturvallisen suunnittelun periaatteet sekä valitaan käyttöön sopivat tietoturvamallit ja suunnittelumenetelmät. Haetaan ratkaisuja esim. uhkapuiden ja tietovirtakaavioiden avulla.

Huomioidaan ratkaisuissa organisaation verkko- ja tietoturva-arkkitehtuurin vaikutukset.

Huomioidaan valittujen ratkaisujen mahdollisesti aiheuttamat uudet tietoturvariskit.

Dokumentoidaan tietoturvallisuuteen vaikuttavat suunnitteluoletukset.

Tietoturvaosaamista tarvitaan ainakin tietoturva-arkkitehtuurin, tietoverkkojen, suunnitteluperiaatteiden, tietokantojen, käyttöjärjestelmien ja käytettävien ohjelmistojen osalta.

## 5.5 Toteutus

### 5.5.1 Kuvaus

Toteutusvaiheessa järjestelmän eri komponentit toteutetaan suunnitelmien mukaisesti. ja suoritetaan komponenttien yksikkötestaus.

Toteutuksen vaiheet:

1. Kehitysympäristöjen pystytys ja testauksen suunnittelu
2. Järjestelmän toteutus: sovellusrunko, käyttöliittymät, tulosteet, liiketoimintalogiikka, tietokanta, liittymät, eräajot, yksikkötestaus, aloitustestauksen suunnittelu ja staattinen analyysi
3. Aloitustestaus
4. Lopputulosten katselmointi

Tietoturvakannanotot prosessikuvauksessa:

- Kuvauksessa viitataan testaus- ja katselmointiohjeisiin, jotka on arvioitu erikseen.
- Kiinnitetään huomiota katselmoinnin, staattisen analyysin ja testauksen tärkeyteen.

Tietoturvakannanotot ohjeissa:

- Web Services ohjeessa lyhyt tietoturva-kappale, jossa maininta käyttäjien tunnistuksesta ja tietoliikenteen salauksesta.

Tietoturvakannanotot dokumenttipohjissa:

- Ei mainintoja

Tietoturvakannanotot tarkistuslistoissa:

- Ei tarkistuslistoja.

#### 5.5.2 Arviointi

Mukana ollut Java-tyyliohje ohjaa yhtenäiseen ohjelmointitapaan, mikä on tärkeää katselmoinnin ja ylläpidon kannalta. Toteutusohjeet eivät opasta tietoturvalliseen ohjelmointiin. Vähintäänkin ohjelmointiohjeissa pitäisi mainita yleisimmät ohjelmoijan keinot ja sudenkuopat sekä opastaa tietoturvallisen, testattavan sovelluskoodin tuottamiseen.

#### 5.5.3 Parannusehdotuksia

Ohjeiden päivitys siten, että niissä kiinnitetään huomiota ohjelmoijan tarvitsemaan tietoturvaosaamiseen. Tyylioppaassa on huomioitava myös tietoturvallisuuden kannalta oikeaoppiset ohjelmointitavat ja ei-toivotut ohjelmointikielen piirteet.

Toteutusvaiheessa on opastettava tietoturvakriittisten osioiden eristämiseen ja testattavuuden varmistamiseen.

### 5.6 Testaus

#### 5.6.1 Kuvaus

Testauksessa varmistetaan ohjelmiston toiminnallisuus ja virheettömyys.

Testauksen vaiheet:

1. Integrointitestaus: varmistetaan järjestelmän osien yhteistoiminta
2. Systeemitestaus: varmistetaan toiminnallisten ja ei-toiminnallisten vaatimusten sekä järjestelmän määrittelyn toteutuminen testiympäristössä
3. Hyväksymistestaus: varmistetaan toimivuus ja vaatimustenmukaisuus tuotantoympäristöä vastaavassa ympäristössä

Tietoturvakannanotot prosessikuvauksessa:

- Tietoturvatestaus on vastuutettu omana tehtävänään tietoturvatestaajalle. Tässä viitataan vaatimuksiin ja turvallisuuden testaussuunnitelmaan.
- Toipumistestaus on omana tehtävänään.

Tietoturvakannanotot ohjeissa:

- Huomioitu, ettei tuotantotietoja saa käyttää sellaisenaan testauksessa.

- Vaatimus ”negatiivisista testitapauksista” eli testataan järjestelmän tahallista rikkomista.

Tietoturvakannanotot dokumenttipohjissa:

- Tietoturvaotsikko

Tietoturvakannanotot tarkistuslistoissa:

- Ei suoria mainintoja.

#### 5.6.2 Arviointi

Testaukseen kiinnitetään hyvin huomiota ja myös tietoturvatestaus vaaditaan. Tietoturvallisuuden testaamisen ei kuitenkaan ole ohjeita eikä apuvälineitä.

#### 5.6.3 Parannusehdotuksia

Tietoturvatestaamiseen tarvita erityisiä keinoja ja apuvälineitä, sekä testitapaukset, joilla pyritään sovelluksen väärinkäyttöön. Dokumentaatiossa tulisi mainita tietoturvatestauksen erityistarpeet. Tietoturvatestaus tulisi myös olla dokumenttipohjissa omana kohtanaan.

Systeemityömallin täytyy ohjata testaamaan tietoturvariskianalyysia ja tietoturvavaatimuksia vasten. Tietoturvakriittisten osioiden toteutuksen hyvyys on testattava, pelkkä tietoturvatoiminnallisuuksien testaus ei riitä.

Testauksessa on huomioitava toimivuus annetussa tietoturva-arkkitehtuurissa ja tuotannon tietoturvaympäristössä: keskitetty käyttäjien hallinta, kertakirjautuminen, tietoliikennerajoitukset, työasemarajoitukset, jne.

Hyväksymistestauksen osana on tehtävä tietojärjestelmän tietoturva-auditointi projektin ulkopuolisen tahon toimesta.

### 5.7 Käyttöönotto

#### 5.7.1 Kuvaus

Käyttöönoton vaiheet:

1. Suunnittelu: käyttöönotto, tuotantoon siirto ja kapasiteetti

2. Valmistelu: tuotantoon siirron katselmointi, tuotantoympäristön perustaminen, sovelluksen jakelu ja asennus, toipumismekanismin ja varajärjestelmän pystytys, koulutus ja ohjeistus, peruutussuunnitelma
3. Toteutus: konversiot, käyttöoikeudet, tuotantoon vienti, tuotantotarkastus, tuotantokäytön aloitus, luovutus ja seuranta
4. Katselmointi: päättäminen ja siirto ylläpitoon

Tietoturvakannanotot prosessikuvauksessa:

- Ei prosessikuvausta.

Tietoturvakannanotot ohjeissa:

- Asennusohjeessa muistutetaan palomuurien konfiguroinnista
- Käyttöönottosuunnitelmassa vaaditaan tietoturvallisuuteen liittyvien tehtävien kuvaus: käyttöoikeudet, tuotantovastuut, dokumenttien käsittely sekä fyysiset ja ohjelmalliset suojaukset
- Valvonta- ja operointiohjeessa huomioidaan varmistukset ja varajärjestelyt.

Tietoturvakannanotot dokumenttipohjissa:

- Tietoturvaotsikko

Tietoturvakannanotot tarkistuslistoissa:

- Käyttöoikeuksien tarkistus, muistutus salassapitosopimuksesta, muistutus jatkuvuussuunnitelmasta

### 5.7.2 Arviointi

Dokumenttipohjissa on varattu paikka tietoturvallisuuteen liittyville toimenpiteille. Tietoturvatoimenpiteitä ei kuitenkaan esitetä riittävästi. Varmistusten ja varajärjestelyjen huomiointi on hyvä asia.

#### 5.7.3 Parannusehdotuksia

Ohjeissa tulisi selvemmin ohjata tietoturvallisiin asennuksiin: tietoturvakovennukset, tuotteiden tietoturvaparametrien asetukset, tietoturvakorjaukset käyttöjärjestelmille ja tuotteille, asennuksen tietoturvatarkastus.

Tuotettavassa dokumentaatiossa täytyy mainita käyttöönoton ja ylläpidon suunnittelun kannalta tietoturvakriittiset asiat.

Ohjeistus voisi muistuttaa integroinnista muuhun tietoturvaympäristöön ja – prosesseihin sekä siihen liittyvästä dokumentoinnista.

### 5.8 Ylläpito

#### 5.8.1 Kuvaus

Ylläpidossa kuvataan tuotantokäytössä olevaan sovellukseen kohdistettavia toimenpiteitä.

Ylläpidon vaiheet:

1. Ylläpitoon siirto: inventointi, arviointi, ylläpitosuunnitelman laadinta ja sovelluksen vastaanotto
2. Ylläpito työmääräyksellä: arviointi, arvioinnin tuki, jatkokehitysprojektin käynnistäminen, muutos ylläpitoon, vian havaitseminen.

Tietoturvakannanotot prosessikuvauksessa:

- Ei mainintoja

Tietoturvakannanotot ohjeissa:

- Ylläpitosuunnitelmassa edellytetään tietoturvajärjestelyjen kuvausta: suojaukset, käyttöoikeudet, salassapitosopimukset, kulunvalvonta, etäyhteydet

Tietoturvakannanotot dokumenttipohjissa:

- Tietoturvaotsikko

Tietoturvakannanotot tarkistuslistoissa:

- Ei mainintoja

#### 5.8.2 Arviointi

Ylläpitosuunnitelmassa edellytetään tietoturvajärjestelyjen dokumentointia, mutta ohjeistus ei ole riittävä.

#### 5.8.3 Parannusehdotuksia

Ylläpitosuunnitelmaan täytyy dokumentoida, kuinka seurataan tietoturvallisuuden toteutumista, tietoturvatilanteen muuttumista ja julkaistuja tietoturvakorjauksia. Lisäksi täytyy dokumentoida tietoturvakorjausten asentaminen sekä tietoturvapoikkeamiin reagointi ja niiden raportointi.

Ylläpitosuunnitelmassa täytyy huomioida tietoturvariskianalyysin ja tietoturvavaatimusten päivittäminen mahdollisten muutosten yhteydessä.

Järjestelmän ja tietojen omistajuus sekä jatkuvuussuunnitteluvelvoite on dokumentoitava.

### 5.9 Erillinen tietoturvaohje

#### 5.9.1 Kuvaus

Tietoturvaohje on laadittu auttamaan tietoturvallisuuden huomioimisessa sovellustyössä huomioiden yleisimmät tietoturvavaatimukset, -tekniikat ja -ongelmat. Lisäksi opastetaan Java-ohjelmoinnin tietoturvapiirteissä.

Tietoturvaopastusta annetaan yleisellä tasolla seuraavista aiheista: tietoturvariskianalyysi, verkkotopologiat, web-sovellustyypit, käyttäjien todennus ja valtuutus, tietojen salaus ja allekirjoitus, istunnon hallinta, lokit ja jäljitysketjut, syötteen tarkistaminen.

Systeemityömallissa on myös erillinen ohjeistus arkkitehtuurien kehittämisen avuksi sekä web-sovellukseen liittyviä arkkitehtuuriohjeita. Tietoturvallisuuteen liittyvät kannanotot liittyvät lähinnä palveluiden sijoitteluun ja tietoliikenteen salaamiseen.

#### 5.9.2 Arviointi

Ohje on johdanto tietoturvallisuuteen sisältäen yleiskuvauksen tunnetuimmista tietoturvavaatimuksista ja – ongelmista.

Tämä on ainoa dokumentti, jossa puhutaan tietoturvariskianalyysistä, vaikkakin hyvin yleisellä tasolla. Hyvää on myös tunnettujen ongelmien listaaminen ja selvittäminen.

Ohje sopii oheislukemistoksi ja johdannoksi tietoturvallisuuteen, mutta se ei huomioi tietoturvakriittisten sovellusten erityistarpeita.

#### 5.9.3 Parannusehdotuksia

Ohjeessa olevia asioita täytyy saada yhdistettyä systeemityömallin eri vaiheisiin. Lisäksi systeemityömallissa täytyy huomioida laajemmin ja yksityiskohtaisemmin tietoturvakriittiset sovellukset.

Ohjetta voidaan haluttaessa laajentaa ja parantaa, mutta silti tietoturvatehtävät on integroitava systeemityön eri vaiheisiin, eikä pidä jättää sitä yksittäisen, irrallisen ohjeen varaan.

### 5.10 Katselmointi

#### 5.10.1 Kuvaus

Katselmointiohjeessa on kuvattu koko projektin kannalta oleellisia katselmointikohteita ja – toimenpiteitä. Katselmointi on jaettu projekti- ja tekniseen katselmointiin. Projektikatselmoinnin painopiste on projektin onnistumisessa ja teknisen katselmoinnin arkkitehtuurissa ja toteutuksessa.

Tietoturvallisuuden kannalta on mainittu:

- laitetason tietoturvaratkaisut
- varusohjelmiston tietoturvaratkaisut
- sovellustason tietoturvaratkaisut
- toipumismekanismit
- staattinen analyysi
- tuotantoon oton tietoturvasuunnitelmat

#### 5.10.2 Arviointi

Yleisen käsityksen mukaan katselmointi on yksi parhaista tavoista edistää sovelluksen laatua ja tietoturvallisuutta. Katselmointiohjeet olivat hyviä ja kattavia, mutta tietoturvakatselmoinnin ohjeistusta on syytä tarkentaa.

### 5.10.3 Parannusehdotuksia

Ohjeisiin lisätään erillinen tietoturvakatselmointiosio, joka kuvaa tietoturvallisuuden katselmointitarpeet ja – tavat. Sovelluskoodin tietoturvakatselmointiin on syytä saada apuvälineitä.

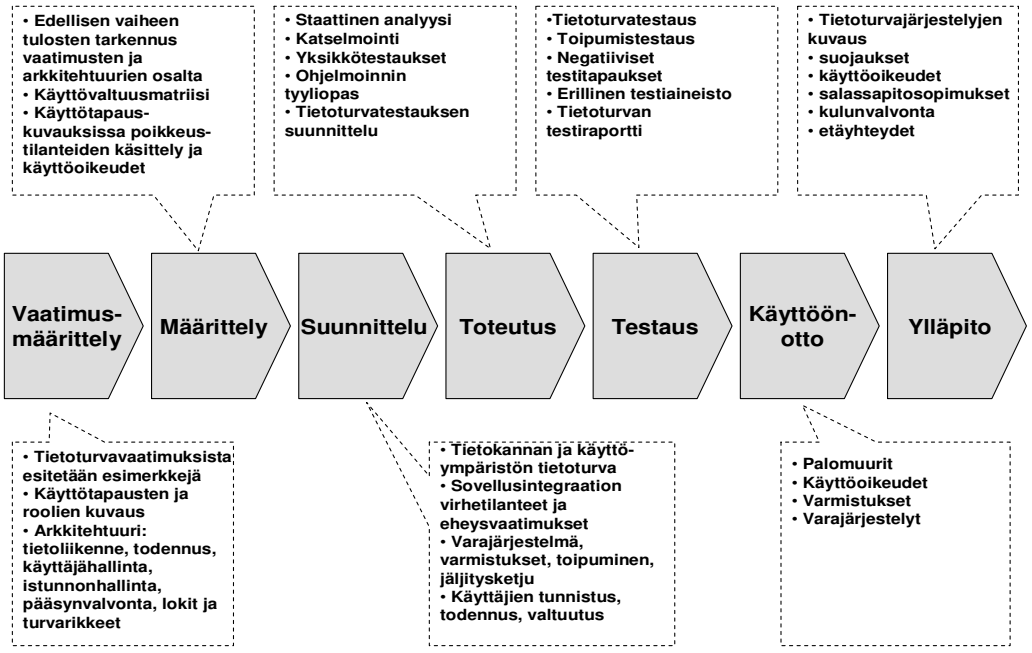
Tietoturvakatselmointi vaatii erityisosaamista. Näkökulma sovelluskoodiin on hyvin erilainen sen mukaan, katselmoidaanko sitä organisaation vaatimustenmukaisuuden, toiminnallisuuden, virhekäsittelyn vai tietoturvallisuuden näkökulmasta. Harva henkilö pystyy arvioimaan sovelluskoodia kaikista näkökulmista yhtä aikaa.

# 6 Yhteenveto ja johtopäätökset

*"Real programmers do not write secure programs. The problem is, at least in part, that real programmers think that programming is about coding."*
*--William Hugh Murray*

Arvioitava systeemityömalli vaikuttaa kattavalta ja laadukkaalta perinteisen systeemityön näkökulmasta. Valitettavasti, myös perinteiseen tyyliin, tietoturvallisuutta ei rakenneta sovellukseen tarpeeksi johdonmukaisesti. Systeemityömalli sisältää hyviä tietoturvallisuuteen liittyviä huomioita ja erityisesti nykyaikaisiin Java- ja Internet-sovelluksiin liittyviä ohjeita. Oleellisin kuitenkin puuttuu eli vaatimus ja ohjeistus tietoturvamääritysten selkeästä kartoittamisesta ja toteuttamisesta tietoturvariskianalyysiin perustuen.

Seuraavassa kuvassa on esitetty systeemityömallin dokumentaatiossa esitettyjä tietoturvakannanottoja ja toimenpiteitä.



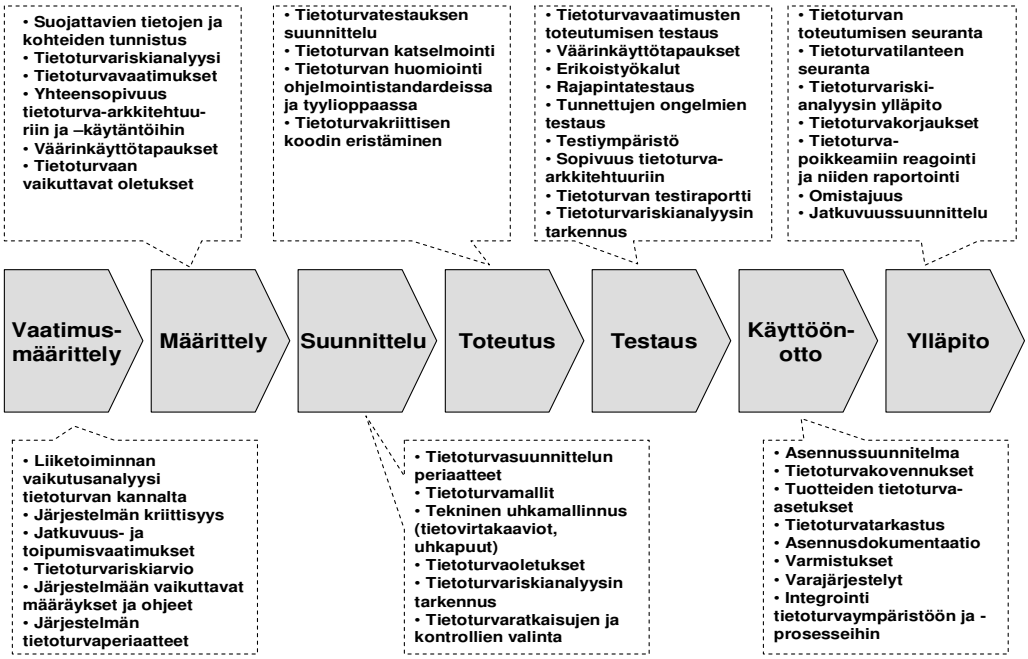

Systeemityömallissa tuodaan tietoturvallisuutta esille kaikissa vaiheissa, mutta usein vain esimerkein ja muistutuksin. Dokumentaatiossa huomioidaan hyvin yleiset tietotur-

vatarpeet, kuten käyttäjien todennus ja valtuutus, tietoliikenteen suojaus, palvelujen sijoittelu, varmistukset ja varajärjestelyt. Tietoturvallisuutta käsitellään vaiheesta riippuen eritasoisesti ja dokumentaatiossa on sekaisin esitystapavaatimuksia, vinkkejä ja oheislukemistoa. Systeemityössä kaivataan kuitenkin systemaattista mallia, ohjeita ja työkaluja riittävän tietoturvatason varmistamiseksi.

Mielenkiintoista on, että esittelymateriaalissa tietoturvallisuutta painotetaan, mutta painotus ei toteudu itse systeemityömallissa. Arvioin, että tämän systeemityömallin avulla sovellusten tietoturvallisuus jää liikaa sovellusprojektiin osallistuvien osaamisen ja kokemuksen varaan.

Systeemityömallia täytyy kehittää siten, että se sisältää ohjeet tietoturvariskianalyysin tekemiseen ja tarkentamiseen projektin aikana sekä ohjeet selkeiden tietoturvavaatimusten kartoittamiselle ja dokumentoinnille. Tietoturvallisuus täytyy vaatimusten pohjalta systemaattisesti suunnitella, toteuttaa ja testata. Ohjeissa ja dokumenttipohjissa täytyy kussakin vaiheessa olla oma tietoturvatehtävänsä ja niihin liittyvät ohjeet. Samoin tietoturvaosaajien tarvetta ja roolia on syytä laajentaa. Systeemityön tietoturvaosaajien täytyy mielestäni pääsääntöisesti olla systeemityöammattilaisia, joilla on riittävä tietoturvaosaaminen ja – näkemys omalta systeemityön osa-alueeltaan. Tietoturvaammattilaisia kannattaa kuitenkin hyödyntää riskianalyysien arvioinnissa, uusien tietoturvahaasteiden arvioinnissa ja ratkaisussa sekä erityiskysymyksissä.

Seuraavissa kuvassa on ehdotukseni systeemityön eri vaiheiden tietoturvatehtävistä:




Dokumentaatioon kaivataan työkaluja ja ohjeita eri vaiheisiin. Esim. tietoturvariskianalyysin esitystapa, tietoturvariskien arviointikriteerit, yrityksen tietoturvamallit, uhkapuumallinnus, testaustyökalut, jne. Jokaisessa vaiheessa on hyvä olla myös oma tietoturvallisuuden tarkistuslista, kunhan muistetaan, että mikään tarkistuslista ei voi olla kattava. Eri vaiheiden tietoturvatehtäviä on kuvattu lisää luvussa *4.1 ”Tietoturvallisuus systeemityön eri vaiheissa”* enkä toista niitä enää tässä.

Edellä kuvatut systeemityömallin tietoturvatehtävät ovat yleispäteviä, eivätkä pelkästään arvioituun systeemityömalliin sopivia. Mielestäni nämä tehtävät kuuluvat kaikkiin kehittyneisiin systeemityömalleihin, joiden avulla kehitetään tietoturvakriittisiä sovelluksia.

Systeemityömallin kehittämisen tueksi suosittelen Information Security Forumin dokumenttia *Standard of Good Practise for Information Security* ja erityisesti sen *Systems Development* osiota [13]. ISF:n standardi kuvaa kattavasti tietojärjestelmäkehityksen tietoturvatarpeet.

Tietoturvatehtävien yksityiskohtien hiomiseen suosittelen dokumentteja *Comprehensive Lightweight Application Security Process (CLASP)* [41] ja *OWASP Guide to Building Secure Web Applications* [29].

Edellä mainittu ISF:n standardi on kattava, mutta koska *VISA PCI DSS* [53] ja *Rahoitustarkastuksen operatiivisten riskien hallinta* [34] vaatimusten täyttäminen ovat keskeisiä organisaatiomme toiminnalle, on niiden täyttyminen varmistettava. Lisäksi, koska organisaatiomme toimintaa tarkastetaan säännöllisesti ulkopuolisten tahojen toimesta, suosittelen koko systeemityömallin evaluointia IT-tarkastuksen *COBIT-mallia* [14] *ja ISO/IEC 17799* standardia [16] vasten.

Systeemityömallin kaupallisuus asettaa haasteita mallin kehittämiselle. Organisaatiomme on sovittava toimittajan kanssa rooleista ja tehtävistä tietoturvallisuuden paremmaksi integroimiseksi systeemityömalliin. Mielestäni on järkevää, että toimittaja sisällyttää dokumentaatioon systemaattisen mallin toteuttaa tietoturvallisia sovelluksia, mutta organisaatiomme yhdistää malliin itse valitsemamme työkalut ja yksityiskohtaiset ohjeet. Roolit ja tehtävät on sovittava selkeästi, jotta päivityksiä voidaan tehdä molempien osapuolten toimesta.

Kunnollinen ja kattava tietoturvallisuuden integrointi systeemityömalliin vaatii pitkäjänteisyyttä ja suunnitelmallisuutta. Tietoturvariskianalyysi ja siitä johdettujen tietoturvavaatimusten tuottaminen ovat ensimmäiset lisättävät tehtävät, koska ne vaikuttavat koko sovellustyöhön ja toimivat myös testauksen perusteena. Seuraavana on syytä kehittää tietoturvatestausta. Näin toimien saamme sovellukselle tietoisesti määritellyt tietoturvatavoitteet ja testausvaiheessa voimme palauttaa epäonnistuneet tuotokset takaisin uudelleensuunniteltavaksi tai -toteutettavaksi. Kokonaisuuden kuntoon saattamiseksi on tehtävä kehityssuunnitelma.

## Lähdeluettelo

1. @stake: *The Security of Applications: Not All Are Created Equal*, 2002, www.netsourceasia.net/resources/atstake_app_unequal.pdf
2. Anderson, Ross: *Security Engineering*, 2001
3. Ferguson, Niels & Schneier, Bruce: *Practical Cryptography*, 2003
4. Gartner: *Press release*, 2004, www.gartner.com/press_releases/asset_104887_11.html
5. Gasser, Morrie: *Building a Secure Computer System*, 1998
6. Graff, Mark G. & van Wyk, Kenneth R.: *Secure Coding - Principles & Practices*, 2003
7. Grand Le, Charles H.: *Software Security Assurance: A Framework for Software Vulnerability Management and Audit*, 2005, www.ouncelabs.com/audit/SoftwareSecurityAssuranceFramework.pdf
8. *Henkilötietolaki* (523/1999), www.finlex.fi
9. Hoglund, Greg & McGraw, Gary: Exploiting *Software - How to Break Code*, 2004
10. Howard, LeBlanc: *Writing Secure Code*, 2002
11. Huseby, Sverre H.: *Innocent Code*, 2004
12. IEEE Security & Privacy May/June 2005 s. 88-91: *Adopting a Software Security Improvement Program*
13. Information Security Forum (ISF): *The Standard of Good Practise for Information Security*, www.isfsecuritystandard.com
14. Information Systems Audit and Control Association (ISACA): *Control Objectives for Information and related Technology (COBIT)*, 2000, www.isaca.org/cobit
15. ISO/IEC 15408: *Evaluation Criteria for Information Technology Security (Common Criteria)*, 2005, www.commoncriteriaportal.org
16. ISO/IEC 17799:2005: *Information Technology - Security Techniques - Code of Practice for Information Security Management*, 2005, 17799.standardsdirect.org

17. ISO 21827: *Systems Security Engineering Capability Maturity Model (SSE-CMM)*, 2003, www.sse-cmm.org

18. ISO/IEC 27001:2005: *Information Technology - Security Techniques - Information Security Management Systems - Requirements*, 2005, 17799.standardsdirect.org

19. Keskuskauppakamari: *Yritysten rikosturvallisuus*, 2005, www.kauppakamari.fi/kauppakamari/julkaisut/Lehdistotiedotteet/fi_FI/turvallisuusselvitys7112005/

20. *Laki luottolaitostoiminnasta* (1607/1993), www.finlex.fi

21. Liikenne- ja viestintäministeriö: *Kansallinen tietoturvallisuusstrategia*, 2004, www.mintc.fi/oliver/upl163-Tietoturvastrategia%2014.pdf

22. McGraw, Gary: *Software Security – Building Security In*, 2006

23. Microsoft: *Security Engineering Explained*, 2005, msdn.microsoft.com/library/en-us/dnpag2/html/secengexplained.asp

24. Microsoft: *The Trustworthy Computing Security Development Lifecycle*, 2005, msdn.microsoft.com/security/default.aspx?pull=/library/en-us/dnsecure/html/sdl.asp

25. Miettinen, Juha E.: *Tietoturvallisuuden johtaminen*, 1999

26. National Cyber Security Partnership Task Force: *Improving the Security Across the Software Development Life Cycle*, 2004, www.cyberpartnership.org/init-soft.html

27. National Institute of Standards and Technology (NIST): *Security Considerations in the Information System Development Life Cycle* (800-64), 2004, csrc.nist.gov/publications/nistpubs/800-64/NIST-SP800-64.pdf

28. O'Neill, Mark et al.: *Web Services Security*, 2003

29. Open Web Application Security Project (OWASP): *Guide to Building Secure Web Applications*, 2005, www.owasp.org/documentation/guide/guide_downloads.html

30. Open Web Application Security Project (OWASP): *Top Ten Project* 2005, www.owasp.org/documentation/topten.html

31. Pedersen, Peter: *Operational readiness to deal with security threats* (seminaarie-sitys), 2005

32. PITAC: *Cyber Security - A Crisis of Prioritization*, 2005, www.nitrd.gov/pitac/reports/20050301_cybersecurity/cybersecurity.pdf

33. PTS Rahoitushuoltotoimikunta: *Rahoitusmarkkinoiden varautumisohje*, 2005

34. Rahoitustarkastus: *Rahoitustarkastuksen standardi 4.4b, Operatiivisten tietoturva-riskien hallinta*, 2004, www.rahoitustarkastus.fi/Fin/Saantely/Maarayskokoelma/Voimassa_olevat_standardit_maaraykset_ja_ohjeet/

35. Ramachandran, Jay: *Designing Security Architecture Solutions*, 2002

36. SANS, *The SANS Top 20 Internet Security Vulnerabilities*, 2005, www.sans.org/top20/

37. Schumacher, Markus: *Security Engineering with Patterns*, 2003

38. Schneier, Bruce: *Beyond Fear*, 2003

39. Schneier, Bruce: *Secrets & Lies*, 2000

40. Secure Business Quarterly: *Tangible ROI through Secure Software Engineering*, 2001, www.sbq.com/sbq/rosi/sbq_rosi_software_engineering.pdf

41. Secure Software Inc.: *CLASP: Comprehensive, Lightweight Application Security Process*, 2005, www.securesoftware.com/solutions/clasp.html

42. Suomen Pankkiyhdistys: *Pankkisalaisuusohjeet*, 2003, www.pankkiyhdistys.fi

43. Swiderski, Frank & Snyder, Window: *Threat Modeling*, 2004

44. *Sähköisen viestinnän tietosuojalaki* (516/2004), www.finlex.fi

45. The Royal Academy of Engineering and The British Computer Society: *The Challenges of Complex IT Projects*, 2004, www.bcs.org/BCS/News/PositionsAndResponses/Positions/complexity.htm

46. Tomhave, Benjamin: *Alphabet Soup: Making Sense of Models, Frameworks, and Methodologies*, 2005, falcon.secureconsulting.net/professional/papers/Alphabet_Soup.pdf

47. U.S. Department of Homeland Security: *Security in the Software Lifecycle,* https://buildsecurityin.us-cert.gov/bsi/docs/Security in the Software Lifecycle DRAFT v08 01092006.pdf

48. *Valmiuslaki* (1080/1991), www.finlex.fi

49. Valtionvarainministeriö: *Valtionhallinnon keskeisten tietojärjestelmien turvaaminen*, 2004, www.vm.fi/vahti

50. Valtionvarainministeriö: *Valtionhallinnon tietojärjestelmäkehityksen tietoturvallisuussuositus*, 2000, www.vm.fi/vahti

51. Valtionvarainministeriö: *Valtionhallinnon tietotekniikkahankintojen tietoturvallisuuden tarkistuslista*, 2001, www.vm.fi/vahti

52. Viega, John & McGraw, Gary: *Building Secure Software*, 2001

53. VISA: *Payment Card Industry Data Security Standard*, 2004, www.visaeurope.com/acceptingvisa/downloads.html