TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

TERO LAITINEN

MOBIILILAITTEIDEN VAHVA AUTENTIKOINTI WINDOWS PHONE ALUSTALLA

Diplomityö

Tarkastaja: Hannu Koivisto
Tarkastaja ja aihe hyväksytty teknisten tieteiden tiedekuntaneuvoston kokouksessa 4. toukokuuta 2016

TIIVISTELMÄ

Tero Laitinen: Mobiililaitteiden vahva autentikointi Windows Phone -alustalla
Tampereen teknillinen yliopisto
Diplomityö, 59 sivua, 0 liitesivua
Kesäkuu 2016
Tietotekniikan diplomi-insinöörin tutkinto-ohjelma
Pääaine: Automaatiotekniikan ohjelmistotekniikka
Tarkastaja: professori Hannu Koivisto

Avainsanat: mobiililaite, vahva autentikointi, Windows Phone

Tässä työssä tutkittiin yleisiä tunnistamistapoja sekä tekniikoita ja arvioitiin niiden soveltuvuutta Windows Phone -alustalla. Tutkimustulosten pohjalta kehitettiin asiakkaiden osoitetietoja käsittelevä Windows Phone 8.1 -sovellus, joka käyttää vahvaa tunnistusta. Lisäksi pohdittiin tutkittujen tekniikoiden heikkouksia ja vahvuuksia, arvioitiin niiden tulevaisuuden näkymiä sekä soveltuvuuksia muihin käyttötarkoituksiin.

Työ jakaantui kahteen osaan: Kirjallisuustutkimukseen ja empiiriseen tutkimukseen. Kirjallisuustutkimusosassa perehdyttiin nykyaikaisiin mobiililaitteillakin hyödynnettäviin tunnistustapoihin ja tietoturvatekniikoihin esittelemällä niiden toimintaperiaatteet ja ominaisuudet. Empiirinen tutkimus suoritettiin toteuttamalla yksinkertainen tunnistus-ja tiedonhakupalvelu ja sitä käyttävä Windows Phone 8.1-sovellus. Tunnistustavan valinta tehtiin kirjallisuustutkimusosuudessa tehtyjen havaintojen pohjalta ja käytettävyyteen, turvallisuuteen ja kustannuksiin liittyviin vaatimuksiin perustuen.

Tutkimuksessa havaittiin, että tarjolla olevista tunnistustavoista vain harvat täyttivät järjestelmälle asettamamme vaatimukset. Osassa tekniikoista edellytetään tunnistamisvaiheessa moniajoa, mikä heikensi sovelluksen käytettävyyttä ja hidasti tunnistamista. Osa tutkimuskohteena olevista tunnistustavoista olisi vaatinut liian korkeat kehitys- tai ylläpitopitokustannukset. Tunnistustekniikoista aikaan perustuvat kertakäyttösalasanat (engl. Time-based One-Time Password, TOTP) osoittautuivat parhaaksi hyvällä turvallisuustasollaan, integroitavuudellaan, käytettävyydellään ja soveltuvuudellaan myös muille mobiilialustoille. Laitekohtainen salausavain ja yhteysosoite tallennettiin laitteelle QR-koodista (Quick Response), josta tiedot tallennettiin sovelluskohtaiseen salattuun tietovarastoon ja sovelluksen käynnistys suojattiin PIN-koodilla. Tiedonsiirto tiedonhakupalvelun ja mobiilisovelluksen välillä tehtiin REST-rajapinnan (Representational State Transfer) ja salatun HTTPS-liikenteen avulla. Pyyntöjen mukana lähetettiin käyttäjätunnuksen ja kertakäyttösalasanan lisäksi laitetunniste ja aikaleima. Laitetunnisteen avulla varmistettiin, että tunnistuspyyntö tulee siltä laitteelta, jolle käyttäjätunnus ja salausavain sidottiin rekisteröinnin yhteydessä. Aikaleimaa käytettiin käyttäjien aikaerojen kompensoimiseen TOTP:ia generoitaessa.

Tutkimuksessa tehtiin johtopäätös, että varsinkin Windows Phone 8.1 -alusta ei vielä tukenut kaikki suosittuja tunnistustapoja kuten fyysisiä biometriikoita. Mutta vuoden 2015 lopulla julkaistut Windows 10 Mobile -laitteet on varusteltu kasvojen ja silmän iiriksen tunnistukseen soveltuvilla kameroilla ja vuoden 2016 kesän jälkeen tämä mobiilialusta alkaa myös tukemaan sormenjälkitunnistusta. Windows 10 Mobile-alustan lupaavimpana uutuutena on Microsoft Passport, joka hyödyntää PIN-koodin tai biometristen tunnisteiden lisäksi laitteelle turvallisesti tallennettuja yksityisiä avaimia käyttäjän tunnistamisessa. Microsoft Passport on hyödynnettävissä omissa sovelluksissa rajapinnan kautta. Microsoftin tarjoamien kattavien ohjeiden avulla kehittäjän on helppo käyttää rajapintaa, mikä tekisi Microsoft Passportista hyvän jatkotutkimuskohteen Windows Phone -alustan tietoturvaan liittyen.

ABSTRACT

Tero Laitinen: Authentication of mobile devices on Windows Phone platform
Tampere University of Technology
Master of Science Thesis, 59 pages, 0 Appendix pages
June 2016
Master's Degree Programme in Information Technology
Major: Software Engineering in Automation Engineering
Examiner: Professor Hannu Koivisto

Keywords: mobile device, strong authentication, Windows Phone

This project studies commonly used authentication methods and technologies and evaluates their suitability for Windows Phone platform. Based on the findings we develop a Windows Phone 8.1 application, which utilizes strong authentication and uses customer information. In addition we discuss the strengths and weaknesses of researched techniques and evaluate their suitability for other purposes and future use.

The research is done in two parts: Literary research and empirical research. In literary research we present modern mobile device authentication methods and techniques by explaining their principles and features. In the empirical study we create a simple web service utilizing strong user authentication and a prototype Windows Phone 8.1 application using that service. The authentication method was selected based on the findings during the literary research, usability, cost, security and some other personal requirements.

The research showed that only few of the authentication methods were suitable for our purposes. Some of the authentication methods required multitasking which weakened the usability of the authentication process. Some of the authentication methods were too expensive to use or develop. Time-based One-time Passwords (TOTP) were found most suitable for us because of their security, integration capability, usability and compatibility for other mobile platforms. A device specific secret key and the destination address of our web service was transferred inside a QR-code, and the content was stored inside the encrypted application storage. The use of our application was secured by using a PIN-code. Our web service was based on REST (Representational State Transfer) and we used HTTPS to send the data securely to our application. Username, TOTP, device ID and timestamp were included with every service request. Device ID was used to make sure that the username and the secret key are used from the device to which they were bound during the registration. Timestamp was used to compensate the user time offset when generating TOTP.

The conclusion of the research was that Windows Phone 8.1 didn't support especially some of the popular authentication methods like biometric authentication. The new Windows 10 Mobile devices, which were published in 2015, support facial and iris

scanning as biometric authentication methods. In summer 2016 Windows 10 Mobile will also start supporting fingerprint scanning. The most promising feature in Windows 10 Mobile will be Microsoft Passport, which uses PIN-code or biometrics in addition to private keys, which are stored securely on the device. Microsoft Passport API can be used in personal applications and Microsoft has presented some helpful instructions on how to use them. Microsoft Passport will surely be something to take a closer look at related to Windows Phone platform security.

ALKUSANAT

Ensimmäisenä haluaisin kiittää työnantajaani Innofactoria mahdollisuudesta tehdä diplomityöni heille töiden ohella. Suuri kiitos kuuluu ohjaajalleni Topi Ahavalle, joka avusti niin projektin käytännön järjestelyjen kuin teknisen ohjeistamisenkin kanssa. Kiitos kuuluu myös kollegoilleni Rami Laiholle ja Henri Hietalalle, joilta sain teknistä tukea yleiseen Windows Phone -sovelluskehitykseen, kehitystyökalujen käyttöönottoon ja sovelluksen sisäiseen jakeluun liittyen.

Diplomityön kirjallisen osuuden ohjaamisesta ja kurssin käytännön asioihin liittyvistä neuvoista haluaisin kiittää professori Hannu Koivistoa. Häneltä sain korvaamatonta opastusta työn rakenteeseen ja sisältöön sekä kurssin muiden osasuoritusten aikataulutuksiin ja suorittamisiin liittyen. Lisäksi haluan kiittää David Hästbackaa osallistumisesta työn tarkastamiseen ja hyvien parannusehdotuksien esittämisestä.

Lopuksi tietysti kiitos kuuluu myös äidilleni Eijalle, joka jaksoi kannustaa ja patistaa minua viikosta toiseen. Kiitos myös kaikille kavereille, jotka jaksoivat kysellä projektini etenemisestä aina tavatessamme.

Espoossa, 23.5.2016

Tero Laitinen

SISÄLLYSLUETTELO

KUVALUETTELO

LYHENTEET JA MERKINNÄT

AES	Advanced Encryption Standard
API	Application Programming Interface
HMAC	Hash Message Authentication Code
HOTP	HMAC-based One Time Password
HTTPS	Hypertext Transfer Protocol Secure
JSON	JavaScript Object Notation
MPNS	Microsoft Push Notification Service
OEM	Original Equipment Manufacturer
OOB	Out-Of-Bounds
OTP	One Time Password
PKI	Public Key Infrastructure
REST	Representational State Transfer
QR	Quick Response
SHA	Secure Hash Algorithm
SIM	Subscriber Identity Module
SMS	Short Message Service
SSL	Secure Sockets Layer
SSP	Secure Simple Pairing
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TOTP	Time-based One-Time Password
UDP	User Datagram Protocol
VPN	Virtual Private Network

1. JOHDANTO

Keskisuuren yrityksen asiakasrajapinnassa toimivan työntekijän kalenteri on varattu täyteen ja päivään mahtuu useita asiakastapaamisia. Näitä velvoitteita suorittaessa pitää pystyä selvittämään minne tapaamiseen on riennettävä seuraavaksi, mutta mobiililaitteelta nähtävien kalenterimerkintöjen tiedot paljastuvatkin puutteellisiksi ja kannettavan tietokoneen avaaminen lisätietojen selvitystä varten on aikaa vievää. Tällaista käyttötapausta varten olisi tarvetta mobiilisovellukselle, jolla voitaisiin pikaisesti tarkistaa tulevat tapaamiset ja niihin liittyvät muut oleelliset tiedot, kuten osoitetiedot, tapaamispaikan sijainti kartalla ja ajo-ohjeet. Mutta näiden arkaluontoisten tietojen tarkastaminen toimiston ulkopuolella pitää pystyä tekemään turvallisesti, joten mobiilisovelluksen käyttämät tiedot tulisi suojata vahvalla tunnistamisella.

Vahvan tunnistamisen tarve ei kuitenkaan koske pelkkiä yrityskäyttäjiä. Esim. sosiaalisen median käyttäjät ovat rekisteröityneitä niin moneen eri verkkopalveluun, ettei salasanojen muistaminen kaikkiin ole edes mahdollista. Sosiaalisen median räjähdysmäisen kasvun myötä tavallisten kaduntallaajien virtuaalisen omaisuuden määrä voi olla niin merkittävä, että sitä ei haluta jättää yhden helposti arvattavissa olevan salasanan taakse. Suuret verkkopalvelut kuten Youtube, Gmail, Twitter ja Hotmail ovat ottaneet 2000-luvun alkupuolella käyttöönsä kaksivaiheisen tunnistamisen käyttäjien verkkosisällön suojaamiseen, mikä on ollut hyvä edistysaskel tilien väärinkäytösten estämiseksi. Koska mobiililaitteet ovat kaikkien helposti saatavilla, ja näitä suuria palveluja on tarvetta käyttää myös mobiililaitteilla, on ollut tarvetta kehittää sopivat tunnistamistavat myös mobiilikäyttäjille. Mobiililaitteiden teknologinen kehittyminen on myös mahdollistanut yhä moderniempien tunnistustapojen hyödyntäminen käyttäjien tunnistamisessa.

Tämän työn tarkoituksena on kehittää Windows Phone 8.1:lle vahvaa tunnistamista hyödyntävä mobiilisovellus, jolla on tarkoitus käsitellä asiakastapaamisiin liittyviä tietoja turvallisesti. Tunnistamistavan ja muiden tietoturvaratkaisujen valinnassa otetaan huomioon kustannukset, käytettävyys, turvallisuus sekä muita järjestelmälle asettamiamme vaatimuksia. Tutkimus toteutetaan kahdessa eri vaiheessa: Kirjallisuusselvityksenä ja empiirisenä tutkimuksensa. Tutkimuksen teoriatausta hankittiin kirjallisuusselvitysosiossa, jossa perehdyttiin vahvoihin tunnistamistapoihin, muihin yleisiin tietoturvatekniikkoihin ja Windows Phone 8.1 -alustan tietoturvaratkaisuihin. Lisäksi tutkittiin em. tekniikoita hyödyntäviä aiemmin toteutettuja järjestelmiä ja arvioitiin, miten tekniikat soveltuvat Windows Phone 8.1 -alustalle. Kirjallisuusselvitys toimi siis tukena empiiriselle tutkimukselle, jossa tekninen järjestelmä suunniteltiin ja toteutettiin kirjallisuusosissa tehtyjen havaintojen perusteella. Empiirisessä osuudessa toteutettiin vahvaa tunnistamista hyödyntävä tunnistamispalvelu ja sitä käyttävä Windows Phone 8.1-

sovellus, sekä arvioitiin syntyneen järjestelmän toimivuutta käytännössä. Tutkimuksen alkamisajankohdan takia sovellusalustaksi valittiin Windows Phone 8.1, sillä Windows 10 Mobilea ei vielä ollut silloin julkaistu, eikä siitä ollut julkisesti saatavilla olevia tietoja. Empiirisestä tutkimusta sovellettiin pääosin niihin tekniikoihin, jotka valittiin hyödynnettäviksi toteutettavassa järjestelmässä. Toteutettu sovellus laitettiin yrityksen Microsoft Company Hubiin jakoon ja tunnistamispalveluun luotiin kymmenelle käyttäjälle testitunnukset. Testikäyttäjiltä ei kuitenkaan kerätty palautetta järjestelmän parantamista varten. Empiirinen tutkimus päätettiin valita toiseksi tutkimustyypiksi, jotta saataisiin parempaa ymmärrystä toteutustavoista ja voitaisiin varmistua valittujen tekniikoiden soveltuvuudesta käytännössä.

Luvussa 2 määritellään vahva tunnistaminen ja kerrotaan lyhyesti tunnistukseen kohdistetuista hyökkäyksistä ja niiden torjumiskeinoista. Luvussa 3 esitellään tämän hetken yleisimpiä tunnistamistapoja, verkkoturvallisuuteen liittyviä tekniikoita ja muita tunnistuksen apuna hyödynnettäviä tekniikoita. Windows Phone -alustan tietoturvaratkaisuja ja sen turvallisuuden vertailua muihin mobiilialustoihin on käsitelty luvussa 4. Luvussa 5 esitellään toteutettavan järjestelmän vaatimukset ja kuvaillaan toteutettu järjestelmä. Luvussa 6 suoritetaan tunnistustapojen vertailu ja arvioidaan niiden soveltuvuutta mobiililaitekäytössä huomioiden järjestelmälle asetetut vaatimukset. Luvussa 7 tehdään yhteenveto, annetaan suosituksia eri tunnistamistapojen hyödyntämisestä Windows Phone -alustalla ja esitellään jatkotutkimustarpeet.

2. VAHVA TUNNISTAMINEN

Tässä luvussa perehdytään siihen, mitä vahva tunnistaminen ja mitä eri tunnistustekijöitä siihen voi liittyä. Vahvan tunnistamisen merkittävyyteen pyritään pääsemään tutustumalla tunnistamisprosessiin, tiedonsiirtoon tai tiedon säilytykseen kohdistuviin hyökkäystapoihin. Lisäksi esitellään pari luokittelutapaa, joita voidaan hyödyntää tietoturvallisten järjestelmien suunnittelussa.

2.1 Mitä on vahva tunnistaminen?

Tietoturvaratkaisujen kehittämisessä käyttäjien tunnistaminen on yksi avainasioista ja tunnistaminen voidaan suorittaa monin eri tavoin [10, s.24]. Tietoturvallisuuden yhteydessä puhuttaessa tunnistamisella tarkoitetaan käyttäjän identiteetin varmistamista tiettyjen menetelmien avulla [78]. Tunnistaminen ei ota kantaa siihen, mitä tunnistettu käyttäjä saa tehdä tai mihin hänellä on oikeus päästä, vaan siitä huolehditaan auktorisoinnilla [10, s.26]. Tunnistusmenetelmät voidaan jaotella eri kategorioihin, joita kutsutaan tekijöiksi. Näiksi tekijöiksi lukeutuvat: ”jotain mitä tiedät”, ”jotain mitä olet” tai ”jotain mitä sinulla on” [10, s.28-31]. Suomen laki jakaa vahvan tunnistamisen näihin kolmeen tekijään. Kun tunnistamisessa hyödynnetään vähintään kahteen eri tekijään perustuvia tunnistustapoja, kutsutaan tunnistamista vahvaksi [78]. Jotkin lähteet listaavat myös muihin tekijöihin perustuvia tunnistusmenetelmiä, kuten ”paikka jossa olet”. Näistä esimerkkinä mainittakoon palvelinsalissa oleva terminaali, joka voi olla ainut paikka, josta palvelimille pääsee käsiksi [10, s.28].

”Jotain mitä tiedät” on todella yleinen tunnistustapa, joka perustuu käyttäjän muistettavissa oleviin asioihin, kuten salasanoihin tai PIN-koodeihin [10, s.27]. Salasanat ovat edelleen yleisin tunnistusmuoto ja riittävän monimutkaisten salasanojen avulla pystytään saavuttamaan kohtalaisen korkea turvallisuustaso [10, s.30]. Monimutkaisia salasanoja on toisaalta vaikeampi muistaa, minkä takia käyttäjät voivat kirjoittaa niitä ylös muistilapuille tai käyttää samaa salasanaa monessa järjestelmässä. Salasanojen joutuminen vääriin käsiin kyseenalaistaa tunnistamisen luotettavuuden, minkä takia niiden säilytyksessä tulee huolellisia [10, s.28-31].

”Jotain mitä olet” perustuu käyttäjän yksilöllisiin fyysisiin ominaisuuksiin, kuten sormenjälkeen, silmän iirikseen, silmän verkkokalvoon tai kasvojen muotoihin. Em. tekijöiden tarkkailuun perustuvia tunnistusmenetelmiä kutsutaan usein biometriseksi tunnistamiseksi [10, s.27]. Biometrisiksi tunnisteiksi voidaan myös lukea käyttäjän käytökseen perustuva toiminta, kuten käsiala tai kirjoitustyyli [10, s.28].

”Jotain mitä sinulla on” pohjautuu käyttäjän omistussuhteeseen johonkin tunnistuksessa käytettyyn esineeseen, kuten pankkikorttiin, henkilökorttiin, fyysiseen tokeniin, älypuhelimeen tai usein myös sähköpostitiliin. Tämän tunnistustavan turvallisuuteen vaikuttaa se, kuinka helposti varastettavissa fyysinen laite on. Esimerkiksi sähköpostitili ei ole läheskään yhtä turvallinen kuin fyysiset laitteet, koska se voidaan kaapata muualta käsin. [10, s.28]

Tunnistuksessa hyödynnettävät tekijät vaikuttavat tunnistuksen vahvuuteen, hintaan ja käytännöllisyyteen, minkä takia tekijät tulisi valita suojeltavan datan mukaan [10, s.29]. Fyysisten biometriikoiden tai laitteiden käyttäminen tunnistusvälineinä johtavat materiaalikustannuksiin, kun taas muistettavissa olevat asiat voidaan tarkistaa ohjelmallisesti. Finanssialalla pelkät salasanat eivät enää ole riittäviä ehkäisemään identiteettiin kohdistuvia hyökkäyksiä niiden nopean yleistymisen ja kehittymisen johdosta. Pankkiautomaatit ovatkin vaatineet PIN-koodin (jotain, mitä tiedät) ja pankkikortin (jotain, mitä omistat) käteisnostojen varmentamiseen [67, s.154]. Kun kyse on asiakkaiden rahoista, olisikin syytä varmistaa, ettei kuka tahansa voi käydä nostamassa yleiseltä automaatilta toisten varoja pelkän salasanan ja käyttäjätunnuksen avulla.

2.2 Tunnistustavan valinta

Tunnistustavan valinta riippuu suuresti toteutettavasta järjestelmästä ja suojeltavasta tiedosta. Esittelen tässä luvussa tunnistustavan valinnalle kaksi eriliasta valintaperustetta, joista ensimmäinen hieman monimutkaisempi sopii pankkisovellutuksiin ja toinen on suppeampi, mutta monissa perussovellutuksissa hyvä tapa tunnistustavan valintaan.

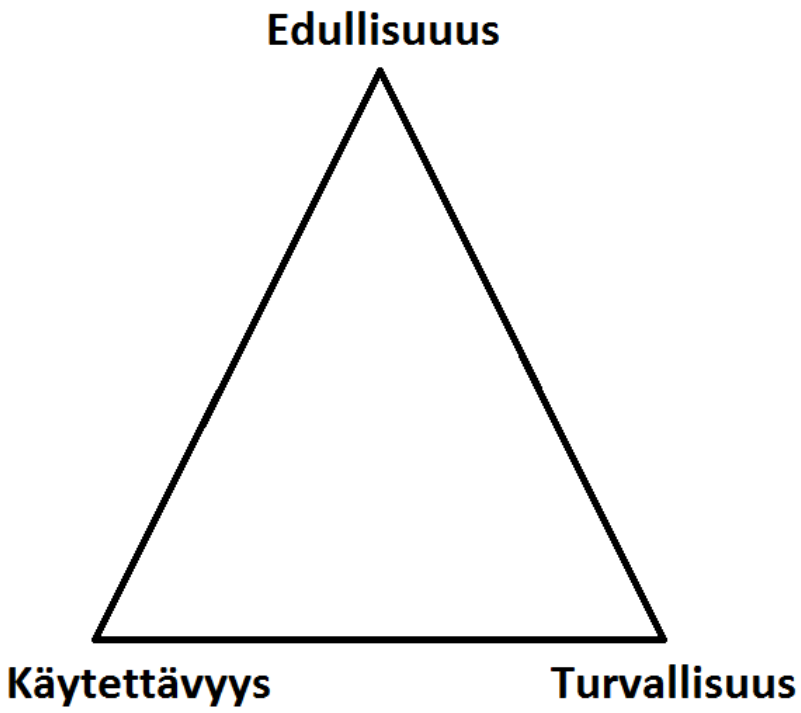
Rao H.R. jakaa tunnistustavan valintaan liittyvät kriteerit kirjassaan Managing Information Assurance in Financial Services viiteen osaan: helppokäyttöisyys, turvallisuus, joustavuus, skaalautuvuus ja kustannustehokkuus sekä saatavuus eri kanavilla [68, s.162]. Valintaperusteet perustuvat pankkisovellusten tietoturvaratkaisujen valintoihin.

- **Helppokäyttöisyys:** Käyttäjätyytyväisyys on kriittinen osa jokaista tietoturvaratkaisua arvioitaessa. Jos tunnistustavan käyttö on liian vaivalloista tai monimutkaista, niin käyttäjät alkavat käyttämään toista tunnistamistapaa tai lopettavat palvelun käytön kokonaan.
- **Turvallisuus:** Tietoturvaratkaisun pitää pystyä torjumaan riittävän hyvin kalastelut, välimieshyökkäykset, murtautumisyritykset ja muut uhat. Turvallisuudesta tulee huolehtia niin normaalikäytön kuin poikkeustilanteenkin aikana, kun esim. käyttäjän tunnistamiseen tarvitsema laite on kadonnut ja se odottaa korvaamista. Tunnistuspalvelujen pitää pystyä todentamaan autenttisuutensa käyttäjälle ja tarjoamaan heille toinenkin tunnistusvaihe salasanan lisäksi.
- **Joustavuus:** Yhä kasvavassa määrin tunnistamista tehdään riskienhallinnan perusteella. Esim. käyttäjän katsoessa pankkitilinsä saldoa riskikerroin on pienempi, kuin jos hän tekisi tilisiirron ulkomailla. Tunnistuspalveluissa pitää pystyä

vaihtamaan tunnistamistapoja sen mukaan, kuinka suuri riski transaktiolla on. Käyttäjien hyväksynnän varmistamiseksi tunnistamisen tulisi olla vain yhtä tunkeileva, kuin mitä riskiltä edellytetään.

- **Skaalautuvuus ja kustannustehokkuus:** Tunnistustavan tulisi pystyä mukautumaan tulevaisuuden muutoksiin, jonka asiakaskunnan monimuotoistuminen tuo mukanaan. Järjestelmän kokonaiskustannuksissa tulisi huomioida lisenssi-, ylläpito- ja laitteiston korvaamisesta aiheutuvat kustannukset.
- **Saatavuus eri käyttökanavilla:** Viisaat organisaatiot tarjoavat pankkipalvelujaan myös selainpohjaisen verkkopankin ulkopuolella, kuten mobiililaitteilla ja pankkiautomaateilla.

Palvelun tietoturvaratkaisujen valitseminen ei kuitenkaan ole aivan näin yksinkertaista, sillä valinnoilla on toistensa välisiä riippuvaisuuksia. Järjestelmän tietoturvaratkaisuihin liittyvät päätökset vaikuttavat järjestelmän turvallisuustasoon, käytettävyyteen ja kustannuksiin, joiden välinen riippuvuus voidaan mallintaa kuvan 1 mukaisesti.



Kuva 1. *Järjestelmän ominaisuuksien riippuvuus, perustuu lähteeseen [71]*

Jokaisesta järjestelmästä voidaan tehdä turvallinen, mutta käytettävyys voi kärsiä sen kustannuksella. Jos järjestelmästä halutaan tehdä myös käytettävä, kasvavat järjestelmän kustannukset. Kustannukset eivät ole pelkästään rahallisia, vaan ne voivat vaatia aikaa ja työvoimaa. Turvallisuus täytyy huomioida jo suunnitteluvaiheessa, eikä se ole vain lähes valmiiseen tuotteeseen lisättävä ominaisuus. Järjestelmästä ei myöskään ole tarkoitusta tehdä täysin turvallista, vaan turvallisuus voi olla riittävä. [71, s.16-17]

2.3 Salasanoihin kohdistuvat hyökkäykset

Yksi vanhimmista hyökkäyksistä perustuu käyttäjien salasanojen arvaamiseen. Sosiaalisen median käytön yleistyttyä käyttäjien julkisilta tileiltä on löydettävissä myös paljon henkilökohtaista tietoa, jota voidaan hyödyntää salasanojen arvaamishyökkäyksissä. Koneellisesti suoritettavia hyökkäyksiä voidaan suorittaa esim. sanakirja- (engl. dictionary), brute force-, ja sateenkaaritauluhyökkäyksien (engl. rainbow table) avulla. Käyttäjätunnuksia ja salasanoja on myös mahdollista kalastella käyttäjiltä väärennettyjen sivustojen kautta, tallentaa niitä tähän tarkoitukseen tehdyillä haittaohjelmilla, eli keyloggereilla tai kurkkia tunnukset olan yli kirjautumistilanteessa. Jos tietoliikennettä ei ole salattu, tietoa voidaan myös suoraan nähdä verkkoliikennettä nuuskimalla. [11, s.12-16]

Tietokoneen näyttöön kiinnitetyn tai näppäimistön alle piilotetun salasanalapun luvattoman lukemisen lisäksi voi käyttäjän salasana tai PIN-koodi päätyä helposti vääriin käsiin olan yli katsomalla [11, s.17]. Tämä hyökkäystapa on huomioitu järjestelmissä korvaamalla salasanakenttään kirjoitetut merkit esim. asteriskeilla, jolloin selkokielinen salasana ei ole luettavista kentästä. Toinen tapa suojautua on rajoittaa näkyvyyttä laitteen näppäimistölle väliseinillä tai muilla ylimääräisillä rakenteilla, joita pankkiautomaateilla ja pankkikorttien lukijalaitteilla on hyödynnetty. Käyttäjä pystyy tarvittaessa suojaamaan näppäimistöä toisella kädellään, jolla näkyvyyttä voidaan rajoittaa entisestään.

Olan yli katsominen ei ole mahdollista silloin kun käyttäjä kirjautuu järjestelmään esim. kotonaan tai toiselta puolelta maapalloa. Käyttäjien tarkkaileminen on mahdollista laitteelle asennettujen haittaohjelmien tai keyloggerien avulla. Keyloggerit ovat käyttäjän laitteelle asennettuja haittaohjelmia, jotka tarkkailevat käyttäjän syötteitä ja lähettävät tiedot hyökkääjälle. Syötteistä on mahdollista poimia käyttäjätunnuksia sekä salasanoja ja käyttää niitä palveluihin tunkeutumisessa. Keyloggereja voidaan torjua virustentorjuntaohjelmilla ja noudattamalla varovaisuutta verkosta ladattavien tiedostojen lataamisen ja sähköpostiliitteiden avaamisen kanssa. Mobiilisovellusalustoilla haitallisten ohjelmien ujuttaminen toisten henkilöjen laitteelle on vaikeaa, eikä hiekkalaatikoinnin takia sovellukselle saa annettua riittävästi oikeuksia, jotta se pääsisi käsiksi toisten sovellusten tietoihin [57].

Kaikki eivät tule ajatelleeksi, kuinka paljon tietoa he paljastavat itsestään lukuisissa käyttämissään verkkopalveluissa. Lemmikin tai lapsen nimen tai syntymäajan käyttäminen salasanana voi kostautua, jos nuo tiedot luettavissa julkisesti tai taustatietojen pohjalta pääteltävissä. Tästä hyvänä esimerkkinä on vuonna 2008 Yhdysvaltojen varapresidenttiehdokkaaksi pyrkineen Sarah Palinin Yahoo -sähköpostitili, jolle onnistuttiin murtautumaan salasanan palautumistoiminnon tietoturvakysymysten, julkisten tietojen ja pienen päättelyn avulla. Osa tietoturvakysymyksen oikeista vastauksista oli löydettävissä Wikipediasta. [77] Kun tietoturvakysymyksen vastaus on salasanaa heikompi, kan-

nattaa salasanan palauttaminen hoitaa turvallisemmalla tavalla, kuten lähettämällä väliaikainen salasana tai sen uusimislinkki sähköpostilla.

Kun salasanojen manuaalinen arvaaminen käy liian työlääksi voidaan salasanoja alkaa arvailemaan ohjelmallisesti ennalta määritellyn listan, kuten sanakirjan pohjalta. Ohjelman käyttämä sanakirja sisältää tavallisten sanojen lisäksi muita potentiaalisia salasanoja. Sanakirjahyökkäyksien tehokkuus perustuu käyttäjien taipumukseen valita helposti muistettavia tavallisia sanoja tai ihmisten ja paikkojen nimiä sisältäviä salasanoja. Hyökkääjä käyttää arvauksissaan merkityksellisiä sanakirjasanoja tai nimiä, joita on huomattavasti vähemmän kuin mielivaltaisesti generoituja satunnaisia salasanoja. Sanakirjahyökkäyksiä voidaan torjua vaatimalla käyttäjiltä monimutkaisempia salasanoja. [11, s.13]

Toinen ohjelmallisesti suoritettava hyökkäystyyppi on brute force, jossa pyritään generoimaan kaikki mahdolliset salasanavariaatiot ja kokeilemaan niitä kohdekäyttäjän salasanana tunnistuksessa. Brute force hyökkäyksen onnistuminen riippuu salasanan monimutkaisuudesta, joten pitkiä ja useista merkeistä koostuvia salasanoja käytettäessä käyttäjän salasana on uusittava salasanapolitiikan takia ennen kuin kaikki vaihtoehdot on ehditty käydä läpi. Tehokas suojautumiskeino brute force hyökkäyksiä vastaan onkin vaatia käyttäjiltä riittävän pitkiä ja monimutkaisia salasanoja, [11, s.12] rajoittamalla salasanojen arvauskertojen määrää tai lisätä tahallisesti viivettä tunnistusprosessiin [10, s.30]. Brute force hyökkäysten tehokkuuteen vaikuttaa myös se, miten järjestelmä ilmoittaa käyttäjälle virheellisestä tunnistusyrityksestä. Jos hyökkääjälle ilmoitetaan, että annettu käyttäjätunnus on virheellinen, ei hänen kannata yrittää tunnistusta sillä tunnuksella käyttäen eri salasanoja. [21, s.146] Tämän takia tunnistuksessa näytettävät virheilmoitukset eivät saisi paljastaa liikaa tietoa, jota hyökkääjä pääsee hyödyntämään.

Koska brute force hyökkäyksessä salasanojen generointi on aikaa vievää, voidaan arvauksien tehokkuutta parantaa ottamalla kombinaatiot sateenkaaritaulusta, johon on valmiiksi generoitu mahdolliset arvot. Sateenkaaritaulujen käyttöä voidaan torjua satunnaistamalla tunnistamisprosessia, kuten käyttämällä haaste-vastetta tai käyttämällä salasanan tarkistussumman (engl. hash) laskentaan satunnaista tietoa (engl. salt). [11, s.13-14]

2.4 Uudelleenlähetyshyökkäys

Uudelleenlähetyshyökkäys (engl. replay attack) on suosittu tapa yrittää tunkeutua järjestelmään toisena käyttäjänä, mutta se vaatii hyökkääjältä pääsyn verkkoon kohdekäyttäjän ja palvelimen välille. Hyökkääjän ei tarvitse saada selville selkokielistä salasanaa, jos salasana aina kryptataan samalla tavalla ja salattu tunnistetieto pysyy samana. Kun käyttäjä lähettää palvelimelle tunnistuspyynnössä tunnistetietonsa salattuna, niin hyökkääjä voi kaapattuaan tiedon lähettää palvelulle saman salatun tunnistustiedon. [11, s.14]

Uudelleenlähetyshyökkäyksiltä voidaan suojautua sisällyttämällä jotain vaihtuvaa tietoa, kuten palvelimen lähettämä satunnainen merkkijono tai aikaleima kryptattuna käyttäjän salasanalla. Tällä tavoin tunnistetieto on aina erilainen ja sen muodostamiseen vaaditaan käyttäjän salasanaa. [11, s.14] Uudelleenlähetyshyökkäyksiltä pystytään suojautumaan myös salausta käyttämällä, jolloin tunnistetiedot eivät ole hyökkääjälle nähtävissä [18, s.128-130].

2.5 Välimieshyökkäys

Jos hyökkääjä onnistuu asettumaan käyttäjän ja palvelimen, tai kahden toistensa kanssa keskenään kommunikoivan päätteen väliin sekä lukemaan ja muokkaamaan heidän välistä viestittelyään, kutsutaan tätä hyökkäystä välimieshyökkäykseksi (engl. man-in-the-middle attack). Tällä hyökkäyksellä voidaan kirjautumistietojen lisäksi varastaa myös käyttäjän ja palvelimen välillä liikkuvaa dataa. [11, s.16] Välimieshyökkäyksen yksi variaatio on session kaappaus (engl. session hijacking), jossa hyökkääjä pystyy hallitsemaan liikennettä käyttäjän ja palvelimen välillä [11, s.16]. Hyökkääjä voi odottaa käyttäjän kirjautuvan palveluun ja kaapata session saadessaan selville käyttäjän session tokenin [22, s.106]. Tämän jälkeen hyökkääjä voi valikoivasti säädellä palvelun ja käyttäjän välistä liikennöintiä [11, s.16].

Välimieshyökkäykseltä voidaan suojautua esimerkiksi käyttämällä liikenteen salausta, molemminpuolista tunnistusta tai tiedon eheyden tarkistusta [11, s.16]. Välimieshyökkäykset pitää toteuttaa uhrin käyttäessä palvelua ja hyökkääjältä vaaditaan taustatietoa järjestelmästä, mitkä tekee hyökkäyksistä vaikeita toteuttaa. Toisaalta pelkkä käyttäjän vahva tunnistaminen ei riitä välimieshyökkäysten torjumiseen. [66, s.159]

2.6 Kalastelu

Kalastelussa (engl. phishing) käyttäjiä yritetään houkutella vieraille väärennetyille sivustoille, joissa heitä yritetään saada syöttämään jotain arkaluontoista tietoa kuten käyttäjätunnuksia, salasanoja tai sosiaaliturvatunnuksia. Näitä tietoja voidaan sitten käyttää kirjautumisyrityksissä oikeisiin palveluihin tai kauppatavarana. Kalastelua pystytään suorittamaan automaattisesti ja vähäistä ihmistyövoimaa hyödyntäen [34, s.433].

Kalasteluviestejä lähetetään pääosin sähköpostien avulla roskapostina [34, s.435], joten käyttäjänä kalasteluilta pystyy suojautumaan noudattamalla varovuutta ja tarkkaivaasuutta arkaluontoista tietoa luovuttaessa. Palveluntarjoajan tulisi tiedottaa käyttäjiään käytännöistään ja mahdollisista heidän nimissään liikkuvista kalastelu- tai huijausviesteistä. Palveluntarjoaja voi myös vähentää hyökkäyspinta-alaa varaamalla itsellensä ne verkkotunnukset (engl. domain), joiden nimet muistuttavat todella paljon varsinaisen sivuston osoitetta. Tällä tavoin voidaan pienentää todennäköisyyttä, että käyttäjä päätyy vieraalle sivustolle osoitekenttään pienen kirjoitusvirheen takia.

2.7 Tiedon varastaminen laitteelta

Sen lisäksi, että palvelun ja käyttäjän välinen liikenne on lähetettävä salattuna, tulee myös mobiililaitteelle säilötty ja sovelluksen käyttämä data olla myös suojattu. Laitteelle säilötty data voidaan suojata kryptaamalla tiedot laitteelle ja purkamalla salaus salasanalla. Yrityskäytössä voidaan myös salata koko puhelimen muisti esim. Exchange ActiveSyncin avulla. Sovelluksen käyttämä data voidaan suojata pitämällä tietoa sovelluskohtaisessa muistissa, johon muut sovellukset ei pääse niihin käsiksi. Sovelluksen käytön aikana esiintyvä data voidaan suojata vaatimalla PIN-koodia käyttöä sovelluksen aloitettaessa. PIN-koodin arvausyrityksien määrä tulisi rajoittaa kolmen ja viiden välillä ja PIN-koodin tulisi olla riittävän pitkä, eikä liian helposti arvattavissa oleva. [7, s.246] Ainakin Windows Phone -alustalla pystytään myös ohjelmallisesti selvittämään, onko käyttäjällä PIN-koodin kysely päällä lukitusnäytön poistossa. Jos näin ei ole, niin käyttäjältä voidaan estää arkaluotoista tietoa sisältävän sovelluksen käyttäminen.

3. VAHVAT TUNNISTUSTAVAT JA VERKKO-TURVALLISUUS

Tässä luvussa esitellään verkkoturvallisuuteen ja käyttäjien tunnistamiseen käytettyjä yleisiä tekniikoita sekä esitellään niiden vahvuuksia ja heikkouksia.

3.1 Verkkoturvallisuus

3.1.1 HTTPS

HTTP-liikenne lähettää tiedon salaamatta, jolloin liikennettä nuuskimalla voi saada selville käyttäjätunnuksia, salasanoja [11, s.14] ja myös mitä tahansa muuta arkaluontoista tietoa. HTTPS (Hypertext Transfer Protocol Secure) on salattu versio HTTP-protokollasta ja sitä käytetään käyttäjän selaimen ja palvelimen välisen luottamuksellisen tiedon suojaamiseen. HTTPS-sivut käyttävät tiedon salaamiseen nykyään TLS-protokollaa (Transport Layer Secure) [85][73], joka turvaa yksityisyyden ja datan eheyden kommunikoivien osapuolien välillä [28]. TLS:n edeltäjä SSL (Secure Sockets Layer) ei ole enää turvallinen ja TLS:n edellinen versio TLS 1.1 korvattiin vuonna 2008 julkaistulla TLS 1.2-versiolla, joka on tällä hetkellä voimassaoleva standardi. TLS 1.2 toi parannuksia erityisesti salausalgoritmien neuvottelun joustavoittamiseen. Lisäksi heikompia salakirjoitusjärjestelmien, kuten MD5:n ja SHA-1:n, käyttö pseudosatunnaisten lukujen generointifunktiossa ja digitaalisissa allekirjoituksissa korvattiin vahvemmalla SHA256-algoritmilla. [28] TLS 1.3 on luonnosteluvaiheessa tällä hetkellä. [87]

TLS koostuu kahdesta protokollasta, TLS Record Protokollasta ja TLS Handshake Protokollasta. TLS Handshake takaa, että kommunikoivat osapuolet voivat tunnistaa toisensa, neuvotella käytetystä salausalgoritmista ja -avaimesta ennen kuin sovelluksella lähetetään yhtään dataa. Tunnistus tapahtuu asymmetriseen tai julkiseen avaimeen pohjautuvaan kryptografiaan kuten RSA:n avulla. TLS Record Protokolla takaa, että yhteys on yksityinen ja luotettava. Yksityisyys taataan salaamalla tieto symmetrisen kryptografian avulla, käyttäen esim. AES:ää (Advanced Encryption Standard). Symmetrisen algoritmin ainutlaatuinen salausavain luodaan sessiokohtaisesti esim. TLS Handshaken avulla. Tiedon luotettavuus taataan eheystarkistelulla käyttäen turvallisia tiivistefunktioita (engl. hash function), kuten SHA-1:stä. [28]

Kun käyttäjä ottaa HTTPS-yhteyden verkkosivulle, lähettää verkkopalvelu SSL sertifikaattinsa selaimelle. Sertifikaatti sisältää julkisen avaimen, jonka avulla suojattu sessio voidaan aloittaa. Tämän jälkeen selain ja verkkopalvelu voivat aloittaa SSL kättelyn

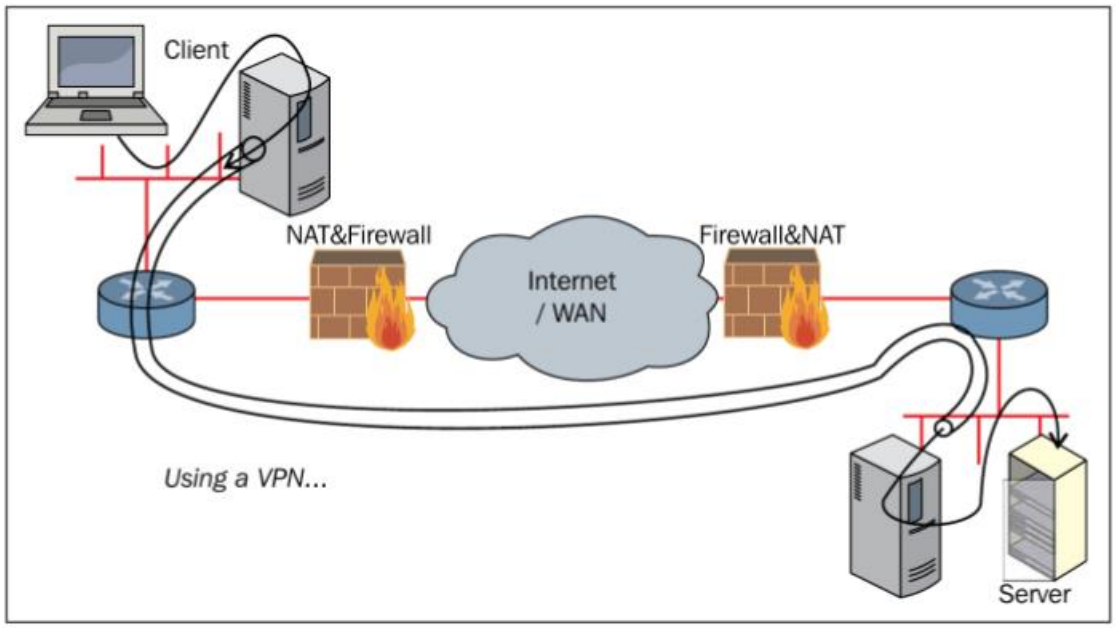
(engl. handshake) [85], jolla saavutetaan kolme asiaa: Ensin osapuolet sopivat käytettävistä salakirjoitusjärjestelmistä kuten kryptografisista algoritmeista, jota he käyttävät datan suojaamiseen. Toiseksi he muodostavat yhteisen jaetun salaisuuden sessioavainmateriaalin muodostamista varten. Lopuksi suoritetaan tunnistaminen, johon SSL/TLS määrittely sallii kolme eri tapaa: anonyymi (engl. anynymous), pelkän palvelimen- (engl. server only) ja molemminpuolinen tunnistaminen (engl. mutual), joista tavanomaisin on pelkän palvelimen tunnistus. [73] Jos avaintenvaihtoon käytetään turvallista salakirjoitusjärjestelmää (engl. cipher suite), kuten väliaikaista Diffie-Hellmania (Ephemeral Diffie-Hellman), niin sessioavaimet johdetaan osapuolten käyttämistä pitkäaikaisista avaimista. Vaikka hyökkääjä saisikin joskus selville osapuolten käyttämät pitkäaikaiset avaimet, ei olisi hänen kannaltaan kannattavaa yrittää päätellä aiemmin käytettyjä sessioavaimia. [72][74, s.84]

TLS-protokolla sallii käyttäjän ja palvelimen välisen kommunikoinnin tavalla, joka on suunniteltu estämään salakuuntelua, peukalointia ja datan väärentämistä. TLS ei itse pysty estämään välimieshyökkäyksiä, sillä hyökkääjällä on olemassa eri tapoja, jolla hän voi pakottaa viestittelyn osapuolet neuvottelemaan käyttöönsä heikommat salausalgoritmit. [28]

TLS:llä on kolme muuta merkittävää rajoitusta. Se ei takaa turvallisuutta päästä päähän, vaan se turvaa kommunikointikanavan ainoastaan selaimen ja palvelimen välillä. TLS ei siis anna mitään suojaa, kun pyyntöjä käsitellään palvelimen taustakoodissa. Toiseksi, palvelin tunnistetaan vain, jos käyttäjä tarkistaa palvelimen TLS sertifikaatin. Selaimet voivat tarkistaa, onko sertifikaatti luotettu, mutta ne jättävät lopulta käyttäjän vastuulle jatkaako tämä palveluun yhdistämistä. Kolmanneksi, hyökkääjä voi hyödyntää verkko-osoitetta, joka on hyvin samankaltainen kuin hyökkäyksen kohteena olevalla sivustolla. Jos hyökkääjä onnistuu hankkimaan palvelimelleen hyväksytyn TLS sertifikaatin ja houkuttelemaan käyttäjän sivustolleen esim. kalastelun avulla, niin TLS ei pysty suojelemaan käyttäjää. Selain ei varoita TLS sertifikaatista, jos sen on luotetun tahon myöntämä, eikä käyttäjä välttämättä huomaa pientä eroa URL:ssa. [73] Jos käyttäjä yhdistää väärennettyyn palveluun, saa hyökkääjä haltuunsa kirjautumistiedot selkokielisenä ja hän voi käyttää niitä kirjautumisessa vastaavaan oikeaan palveluun. [16, s.313]

3.1.2 VPN

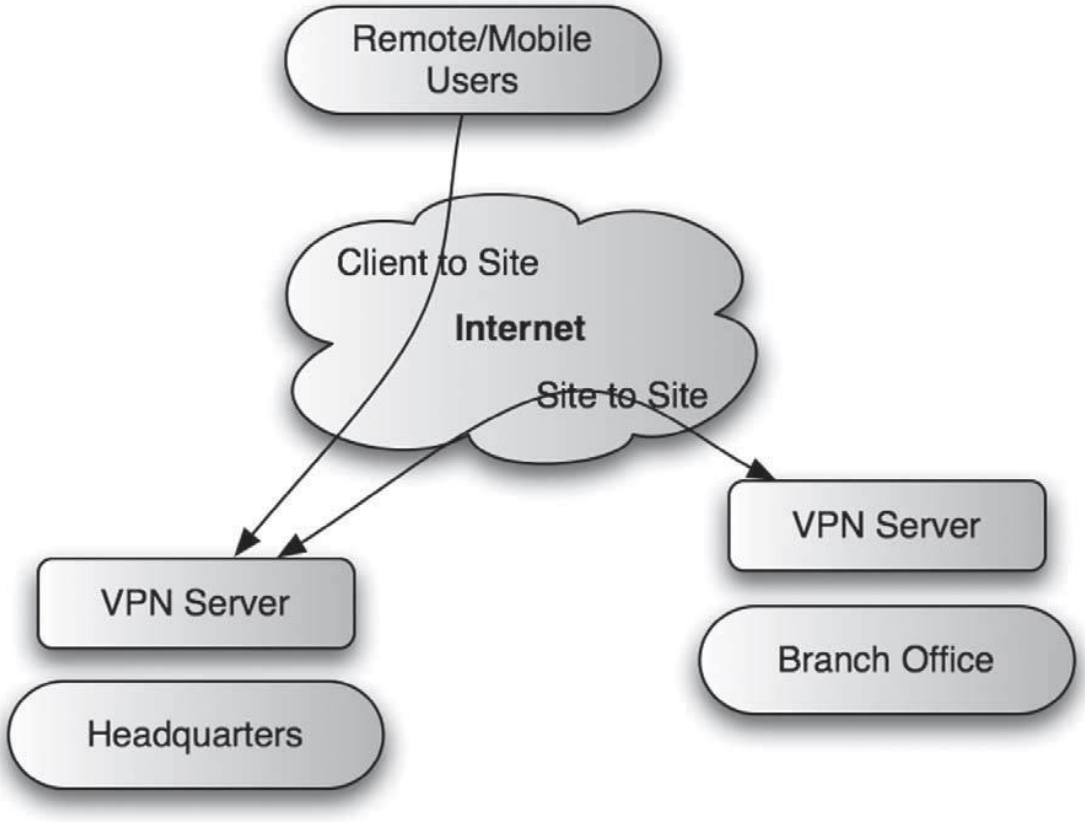
Verkkoliikenne on melko avointa tarkkailulle ja analysoinnille. Vaikka salatut protokollat kuten HTTPS ja SSH (Secure Shell) ovat turvallisempia, ne pystytään silti tunnistamaan verkkoliikennettä tutkivan hyökkääjän toimesta. Hyökkääjä näkee minkä tyyppistä liikennettä minkäkin palvelimen ja tietokoneen välillä liikkuu. [23, s.4] VPN (Virtual Private Network) on yksityinen tietoverkko, joka käyttää julkista telekommunikaatioinfrastruktuuria säilyttämällä yksityisyyden tunnelointiprotokollien ja muiden tietoturvatekniikoiden avulla [9, s.223]. Kun käytetään VPN:ää, niin verkkoliikenne VPN:n sisällä ei ole tunnistettavissa. Käytettäessä VPN:ää toimii liikenne kuvan 2 mukaisesti.



Kuva 2. *VPN:n toiminta[23, s.4]*

Kun VPN-liikenne reititetään verkossa, näkevät muut verkon laitteet vain VPN-liikenteen, muttei mitä VPN-tunnelin sisällä liikkuu. Tunnelin sisällä HTTPS- ja SSH-liikenne on suojattu muilta VPN-tunnelin käyttäjiltä, mutta ei tunnistettavissa tunnelin ulkopuolelta. VPN ei ainoastaan salaa liikennettä tunnelin sisällä, se myös piilottaa ja suojaa yksittäisiä yhteyksiä tunnelin ulkopuolella olijoilta. Koska VPN-liikenne päästetään reitittimien ja palomuurien läpi, pitää turvallisuudesta huolehtia muilla tavoin. [23, s.4]

VPN:ää käytetään erottamaan yrityksen sisäiset verkot avoimesta Internetistä ja sillä voidaan tehokkaasti turvata yritysten väliset palvelut tai etätyöntekijöiden pääsy yrityksen sisäverkon palveluihin [9, s.223]. Kuvassa 3 on havainnollistettu VPN:n käyttötarkoitukset.



Kuva 3. *VPN:n käyttötarkoitukset [9, s.223]*

VPN:ää käytetään kahdella tavalla: yhdistämällä kaksi LAN-verkkoa toisiinsa tai muodostamalla etäyhteys yksittäisen päätteen ja verkon välillä. VPN mahdollistaa kotona olevien tai matkustavien etäkäyttäjien yhteydet yritysverkkoon. [7, s.855][54]

VPN-yhteys muodostus tapahtuu tunneloinnin avulla. Tunnelointi sisältää pakettien kapseloinnin, reitityksen ja kapseloinnin purkamiseen. Kapseloinnissa paketti muodostetaan uudelleen lisäämällä alkuperäisen paketin otsikkotietoihin lisätietoa, joka voi sisältää uuden osoitteen ja reititystietoa. Kapselointivaiheessa myös salataan alkuperäisen paketin sisältö. Reititystiedon avulla paketti ohjataan perille. Kun kapseloitu paketti saapuu määränpäähän, kapseloinnin lisäämä lisätieto voidaan poistaa ja alkuperäisen paketin kohteen tietojen avulla voidaan alkuperäinen paketti toimittaa kohteeseen. [26, s.29]

Useimmat VPN-toteutukset käyttävät liikenteen salausta ja tunnistamista. Salauksella varmistutaan, etteivät muut liikennettä tarkkailevat osapuolet voi purkaa ja analysoida dataa. Tunnistamista on hyödynnetty kahdessa vaiheessa: Käyttäjän tunnistamisessa ja yhteyden suojaamisessa. Ensin varmistutaan, että järjestelmään yhteyden ottaneet tunnistetaan. Tämä tunnistus voi esim. perustua käyttäjätunnuksiin ja salasanoihin tai sertifikaatteihin. [23, s.5] Yhteyden suojaamisessa lähetetyt paketit allekirjoitetaan. Ennen pakettien kuorman purkamista tarkistaa järjestelmä, että sen vastaanottamat paketit on allekirjoitettu asianmukaisesti. Tunnistamalla paketit, järjestelmä säästää suoritusaikaa

kun tunnistamisehtoja täyttämättömiä paketteja ei tarvitse purkaa. Näillä pystytään estämään palvelunestohyökkäykset ja välimieshyökkäykset olettaen, että allekirjoitukseen käytetyt avaimet on pidetty salassa. [23, s.5]

Erilaisia VPN-ratkaisuja on olemassa useita, joista osa on kaupallisia ja osa avoimeen lähdekoodin perustuvia. VPN-tuotteet voidaan jaotella seuraaviin kategorioihin: [23, s.5-7]

- PPTP-protokollaan perustuvat VPN:t
- IPSec/L2TP-protokolliin perustuvat VPN:t
- SSL:ään perustuvat VPN:t
- OpenVPN

Koska PPTP ei heikomman turvallisuutensa puolesta ole enää niin yleisesti käytössä [23, s.6][34] ja OpenVPN usein luokitellaan SSL/TLS:ään perustuvaksi [23, s.9] niin esittelen tässä vain IPSec/L2TP ja SSL:ään pohjautuvien VPN:ien toimintaa.

IPSec perustuu standardiin RFC2411[38] ja se toimii OSI-mallin kerroksilla 2 ja 4. IPSec voidaan konfiguroida käyttämään ennalta jaettuja avaimia tai X.509-sertifikaatteja VPN-yhteyden turvaamiseen. VPN-yhteyden tunnistamiseen se käyttää X.509-sertifikaatteja, salasanoja tai kertakäyttösalasanoja. IPSec voi toimia kahdessa eri moodissa: Tunnelointimoodissa (engl. tunneling mode) ja kuljetusmoodissa (engl. transport mode). Kuljetusmoodissa ainoastaan IP-paketin kuorma salataan ja tunnistetaan. Kuljetusmoodia käytetään useimmiten L2TP:n (Level 2 Tunneling Protocol) kanssa, jolloin L2TP huolehtii käyttäjän tunnistamisesta. Pelkkä IPSec:in käyttö on myös mahdollista. Tunnelointimoodissa koko IP-paketti salataan ja tunnistetaan, ja siitä muodostetaan uusi IP-paketti uusilla otsikkotiedoilla. IPSecin etuna on, että se tulee melkein kaikkien käyttöjärjestelmien, palomuurien, reitittimien ja kytkinten ajurien mukana. IPSec on erittäin turvallinen, se on tuettuna hyvin eri käyttöjärjestelmillä ja tietoturvapolitiikkojen määrittäminen on monipuolista. IPSecin haittana on, että monet palveluntarjoajat ovat tehneet käyttämäänsä IPSec:iin laajennuksia, mikä tekee palveluntarjoajien välisten yhteyksien muodostamisesta lähes mahdotonta. IPSec:ssä konfigurointi ja vikojen selvitys on toisaalta vaikeaa ja lisäksi IPSec ei intregroidu hyvin NAT:tattujen verkkojen kanssa. [23, s.6-9]

SSL:ään perustuvat VPN:t, joihin myös OpenVPN voidaan luokitella, ovat nykyään yleisimpiä VPN-tyyppejä. SSL VPN:T perustuvat SSL/TLS-protokolliin eivätkä ne yleensä käytä asiakaspuolen sovellusta vaikka joitain sovelluksia kuten Cisco AnyConnect ja Microsoft SSTP löytyy. Monet SSL VPN:t käyttävät HTTPS:n kanssa samaa verkkoprotokollaa ja SSL/TLS:ää salatun yhteyden muodostamiseen. Yhteys suojataan yleensä X.509-sertifikaateilla ja yhteyden tunnistus tehdään kertakäyttösalasanoilla tai salasanoilla. SSL VPN:t eivät tarvitse juuri ollenkaan asiakaspuolen sovelluksia toimiakseen. Täten asiakaspuolen asennukset ja alustukset ovat nopeita. SSL VPN:ien huono

puoli on, etteivät ne ole täysverisiä VPN:iä vaan tarjoavat pääsyn vain yhdelle tai joukolle palvelimia. Paikallisen datan jakaminen on myös vaikeampaa SSL VPN:llä. [23, s.7-9]

VPN:ää käytetään esim. pankkiautomaattien ja pankkien väliseen turvalliseen tiedonsiirtoon tai organisaatioiden sisällä toimistojen välisen tiedon suojaamiseen. Kuluttaja voi käyttää VPN:ää avoimessa WLAN-verkossa liikenteen salaamiseen tai sensuurin ja maarajoitusten kiertämisessä päästäkseen ulkomaisiin verkkopalveluihin. [23, s.3]

VPN:n käytön etuina ovat edulliset käyttökustannukset, mutta koska yhteys muodostetaan julkisen verkon yli, ei sen suorituskyky ole yrityksen hallittavissa. VPN:n käyttöönotto yrityksessä ei ole yksinkertaista ja se tulee yleensä hankkia palveluntarjoajalta. Tunnistuksen ja salauksen ansiosta VPN takaa korkean turvallisuustason. [27] Koska VPN:n kautta on pääsy yrityksen sisäverkkoon, tulisi tunnistuksen olla vahva. Vaarana on myös yhteyden muodostamisen jälkeen, että laite joutuu vääriin käsiin, jolloin laitteen haltijalla olisi pääsy sisäverkkoon.

Kohdeyritys, jolle mobiilisovellus kehitetään, käyttää Cisco AnyConnect VPN:ää, johon tarvitaan Ciscon tarjoama sovellus. OpenVPN-tekniikkaan perustuvan Cisco AnyConnect Secure Mobility Client for Windows Phone -sovelluksen käyttö ei ollut mahdollista vielä Windows Phone 8:lla mutta 8.1-versiossa se löytyy jo tuettuna. Kyseinen sovellus on tällä hetkellä saatavilla sovelluskaupasta, joten nykyään VPN:n käyttö tällä sovelluksella olisi mahdollista. [33][39]

3.2 Käyttäjien tunnistaminen

Tässä luvussa perehdytään tällä hetkellä käytössä oleviin käyttäjien tunnistamiseen liittyviin tekniikohin. Kukin tunnistamistapa selitetään pääpiirteittäin, minkä lisäksi pohditaan tekniikoiden heikkouksia ja vahvuuksia.

3.2.1 Out Of Band

OOB eli Out Of Band -tunnistus tehdään käyttämällä eri kommunikointikanavaa kuin millä palvelun sisältöä lähetetään. Jos palvelua käytetään selaimen kautta, niin tunnistamisessa voidaan selaimeen syötettävien käyttäjätunnuksen ja salasanan lisäksi vaatia puhelimeen toimitettua kertakäyttösalasanaa. Kertakäyttösalasana voidaan esim. lähettää tunnistuspalvelimelta puhelimeen SMS-viestinä (Short Message Service) tai generoida suoraan mobiililaitteella. SMS-viestejä tukevien puhelinten yleistyessä palvelimelta generoidut ja puhelimeen lähetettävät kertakäyttösalasanat ovat muodostuneet edulliseksi tavaksi parantaa turvallisuutta. [15, s.67] SMS-viestien käyttö soveltuu pitkän kantaman viestinvälitykseen, jonka luotettavuus voidaan taata palveluntarjoajien toimesta. SMS-viestejä vastaan on olemassa hyökkäyksiä, mutta niiden hyödyntämiseen vaaditaan paljon tietotaitoa ja resursseja. [43] Hyökkääjän pitää pystyä käyttämään vä-

limieshyökkäystä, jotta hän pystyy pääsemään OOB-tunnistusta hyödyntävään järjestelmään [16, s.306].

Jos sovelluksen käytön kannalta on käytännöllistä, voidaan myös käyttää tunnistusviestien lähettämiseen lyhyen kantaman langattomia kanavia kuten NFC:tä (Near Field Communication) tai Bluetoothia. NFC toimii radiotaajuudella 13,6MHz ja se kantama on alle 10cm. Tunnistamisen turvallisuus perustuu siihen luottamukseen, että laitteen välitön lähiympäristö on turvallinen viestin lähetykseen. NFC on käytössä jo mobiililaitteiden lähimaksupäätteissä, joissa käyttäjä voi kuitata ostoksensa käyttämällä puhelintaan maksupäätteen päällä maksun hyväksymiseksi. NFC:tä vastaan pystytään käyttämään välimieshyökkäyksiä ja viestejä voidaan tarkkailla, ellei yhteystason tietoturvasta ole huolehdittu eikä sitä ole vahvistettu välimieshyökkäyksiä torjuvalla protokollalla, kuten kertakäyttöisellä sessioavaimella. [43, s.24] Lyhyen kantamansa takia NFC:n käyttö vaihtoehtoisena kanavana on hyvin rajallista. Sitä voitaisiin hyödyntää esim. rekisteröintiprosessissa, joka suoritetaan kontrolloidussa ympäristössä. Bluetooth on suositu lyhyen kantaman kommunikointitekniikka, joka on tuettuna suurimmassa osassa mobiililaitteista. Bluetoothin hyödyntämisestä mobiilimaksujärjestelmistä löytyy myös toteutuksia. Bluetooth-laitteita pystyy etsimään lähiympäristöstä, mutta laiteparin muodostaminen on hidasta. Lisäksi vanhojen laitemallien käyttö esim. maksupäätteenä ei ole turvallista, koska ne eivät tue Bluetooth versiota v2.1, johon laitteiden turvallinen paritus (engl. Secure Simple Pairing, SSP) on lisätty. [43, s.25]

OOB kanavia voidaan käyttää tunnistamisen lisäksi myös turvallisessa tiedonsiirrossa viivakoodinlukijalla varustetulle laitteelle [67, s.8]. Tätä keinoa hyödyntävät esim. Microsoft Authenticator ja Google Authenticator- sovellukset, joiden käyttäjäkohtaiset avaimet näytetään käyttäjälle selaimen ruudulla ja ne ovat skannattavissa QR-koodin lukijasovelluksella puhelimelle.[55] Myös WhatsApp Web, Wikipedia ja LinkedIn hyödyntävät QR-koodeja [62]

3.2.2 Kertakäyttösalasanat

Kertakäyttösalasanat (engl. One-Time Password, OTP) ovat salaisuuksia, joita voidaan käyttää tunnistamisessa vain kerran, joten käyttäjä tarvitsee uuden salasanan jokaista tunnistusta varten. Kertakäyttösalasanat voidaan toimittaa käyttäjälle kirjautumiskertojen yhteydessä jonkin toisen kanavan kautta. [11, s.22] Kertakäyttösalasanoja voidaan myös toteuttaa token- tai USB-laitteilla, älykorteilla, salasanalistoilla tai ohjelmallisesti. [16, s.304] Token-pohjaisista käyttäjän toimesta generoitavista kertakäyttösaloista kerron tarkemmin luvuissa 3.2.4 - 3.2.6.

SMS-viestit lähetetään salaamattomana GSM-verkossa, mutta niiden salakuuntelu on silti hankalaa. Hyökkääjä ei kuitenkaan pysty kirjautumaan mihinkään pelkällä kertakäyttösalasanalla. SMS-viestit ovat tuttuja useimmille käyttäjistä, mikä helpottaa tunnistamistavan omaksumista. SMS-viestit eivät vaadi muutoksia SIM-korttiin (Subscriber

Identity Module) tai puhelimen sovelluksiin. Kertakäyttösalasanoja voidaan generoida myös erityisvalmistetulta SIM-kortilta, mikä tekee järjestelmästä entistä turvallisemman kun salasanoja ei tarvitse lähettää salaamattoman väylä yli. [24]

Yksi monien tuntema kertakäyttösalasanoihin pohjautuva tunnistustekniikka on käytössä verkkopankkien tunnuslukulistoissa. Erityisesti SMS-viesteillä ja sähköpostilla lähetettävät kertakäyttösalasanat ovat käytössä monissa kotimaisissa palveluissa [70].

3.2.3 Mobiilivarmenne

Mobiilivarmenne on mobiilipäätelaitteen liittymäkorteilla sijaitseviin yksityisiin avaimiin perustuva asiointivarmenne, jota voidaan käyttää henkilön sähköiseen tunnistamiseen, viestinnän salaamiseen ja sähköiseen allekirjoitukseen [1, s.7]. Yksityiset avaimet luodaan varmenteelle turvallisesti, eivätkä ne ole kopiotavissa tai siirrettävissä liittymäkortilta. Vuodesta 2012 lähtien avaimet ovat olleet vähintään 2048-bittisiä RSA-avaimia. [1, s.24] Sähköistä allekirjoitusta ja tunnistamista varten on omat avaimensa ja niiden käyttö on suojattu tunnusluvuilla [1, s.25].

Mobiilivarmennetunnistus toimii samalla tavalla eri operaattorien tarjoamilla mobiilivarmenteilla. Mobiilivarmennetunnistuksessa käyttäjä syöttää aluksi verkkopalveluun käyttäjätunnuksensa tai kännykkänumeronsa, jonka jälkeen palvelusta lähetään käyttäjän numeroon tieto kirjautumisyrityksestä. Käyttäjä kuittaa pyynnön syöttämällä 4-8-numeroisen tunnuslukunsa, jonka jälkeen tieto tunnistamisesta lähtee verkkosivulle ja palvelu aukeaa. [2] Käyttäjän tunnuslukua ei lähetetä eteenpäin, vaan sen oikeellisuus tarkistetaan SIM-kortilla [3].

Mobiilivarmenteeseen on saatavilla myös häirinnänestopalvelu, joka estää tunnistuspyynnön lähettämisen käyttäjän matkapuhelimeen, ellei tunnistuspyynnön aloittajan syöttämä häirinnänestokoodi ole oikea. [4] Häirinnänestopalvelulla pystytään torjumaan kalasteluyrityksiä, jossa hyökkääjä yrittää kirjautua järjestelmään toisena henkilönä ja saada hänet kuittaamaan tunnistamispyyntönsä.

Mobiilivarmenne on saatavilla eri operaattoreilta, joten käyttäjän ei tarvitse vaihtaa operaattoria halutessaan hyödyntää tätä tunnistuspalvelua. Rekisteröinti tosin vaatii henkilökohtaisen käynnin liikkeessä ja mobiilivarmenteen käyttöönotto voi edellyttää SIM-kortin vaihtamista, sillä se ei toimi kaikilla liittymätyypeillä. [2] Esim. uudemmat dataliittymät eivät olleet saatavilla mobiilivarmennekorteilla, mikä kävi ilmi vuonna 2015 tehdyn kyselyn perusteella. Mobiilivarmenne soveltuisi paremmin niille puhelinmalleille, jotka tukevat kahta SIM-korttia. Tällöin mobiilivarmenne voitaisiin ottaa aiemman SIM-kortin rinnalle. Mobiilivarmenteet ovat operaattorista riippuen käyttäjille maksuttomia tai hyvin edullisia parin euron kuukausihinnalla [60][66][69]. Ne ovat erittäin turvallisia ja kulkevat kätevästi puhelimen mukana, joten ylimääräisten salasanalistojen tai muiden lisälaitteiden mukana kuljettamista ei vaadita [3]. Vaikka mobiilivarmenteen

käyttö onkin selkeää verkkopalveluissa, pyyntöjen erillinen vahvistusnäkymä voi haitata käyttöä mobiilisovellusta tunnistettaessa.

Vaikka mobiilivarmenne on lupaavalta kuulostava mobiilitunnistukseen käytettävä tekniikka, ei se kuitenkaan ole noussut todella suosituksi vuoden 2010 käyttöönottonsa jälkeen. Vasta vuoden 2015 lopulla ensimmäinen suomalainen pankki otti sen käyttöönsä, vaikka se onkin ollut jo laajemmin käytössä julkishallinnolla, kuten vakuutusyhtiöillä. [2][65]

3.2.4 Tokenit

Token on salausavainta käyttävä laite, jonka salausalgoritmi on tunnistuspalvelimelle tiedossa. Tokenit generoivat kertakäyttösalasanoja, jotka toimivat vain yhdellä tunnistuskerralla. [7, s.75] Tokenit voidaan jakaa fyysisiin- (engl. hard) ja sovelluspohjaisiin (engl. soft) tokeneihin. Fyysiset tokenit ovat pieniä laitteita, jotka generoivat kertakäyttösalasanat pienelle ruudulle sisäisen salausavaimensa perusteella joko napin painalluksesta tai lyhyin väliajoin. Fyysiset tokenit ovat kestäviä ja helppoja käyttää, eikä niiden käyttöön vaadita muuta lisälaitteistoa. [67, s.155] Fyysisistä tokeneista yhtenä esimerkkinä on RSA SecurID [8, s.114], RSA SecurID käyttää jaettua salaisuutta, jolloin käyttäjien tokenien generoimaa aikaan perustuvaa kertakäyttösalasanaa voidaan verrata palvelinpuolella samalla ajanhetkellä generoituun salasanaan. Sovelluspohjaiset tokenit ovat käyttäjän koneelle tai mobiililaitteelle asennettavia sovelluksia, joihin salainen avain on turvallisesti säilötty. [67, s.155] Sovelluspohjaiset tokenit eivät ole yhtä turvallisia kuin fyysiset tokenit, sillä käyttäjän koneelle voidaan tunkeutua ja sovellusta on fyysistä laitetta helpompi manipuloida. [8, s.114] Tokeneja käytetään useimmiten kaksivaiheisessa tunnistuksessa, joten niiden pelkkä hallussapito ei takaa pääsyä järjestelmään. Käyttäjän pitää lisäksi tietää esim. tokenin omistajan käyttäjätunnus ja salasana tai PIN-koodi.

Tokenit voivat toimia aikasynkronisesti, tapahtumasynkronisesti tai haasteeseen perustuen. Aikasynkroninen token generoi tunnuslukuja palvelun kanssa synkronoidun kellonsa perusteella. Tapahtumasynkronisessa mallissa palvelin ylläpitää salasanalistaa, josta palvelin näkee järjestykseen perustuen, mikä salasana kulloinkin vaaditaan kirjautumiseen. Haasteeseen perustuvassa mallissa käyttäjälle lähetetään palvelimelta joku vaihtuva tieto, jonka käyttäjä syöttää tokeniinsa. Token generoi haasteesta vastauksen, jonka käyttäjä lähettää palveluun. Palvelu vahvistaa vastauksen perusteella, että käyttäjä on oikea. [7, s.76] Tapahtumasynkronisten ja aikasynkronisten tokenien käyttämien algoritmien toimintaperiaatteesta on kerrottu tarkemmin luvuissa 3.2.5 ja 3.2.6.

Tokeneihin perustuvissa järjestelmissä on etuna se, että käyttäjät usein ilmoittavat kadonneista tokeneista tai niihin sidotuista laitteista pikaisesti, jolloin token voidaan deaktivoida [18, s.127-128]. Toisaalta fyysisten tokenien uusiminen ja uudelleenrekisteröinti on hidasta. Fyysiset kertakäyttösalasanatokenit ovat kalliita ja työläitä ottaa käyttöön

suurissa organisaatioissa [7, s.461]. Jos fyysisiä tokeneja hallinnoidaan eri hallinnointipalvelimilta, voi käyttäjä tarvita eri tokenit eri palveluille. Jos käyttäjällä on useita fyysisiä tokeneja käytössä, niin niiden mukana kuljettaminen on työlästä, ja on vaikeampaa muistaa mikä token kuuluu mihinkin palveluun. Sovelluspohjaisessa tokenissa käyttäjä sidotaan laitteeseen, jolta hänen tulee kirjautua. [8, s.114] Mikä tahansa token-sovellus, jolle on lisätty sama salausaivan, pystyy generoimaan samoja kertakäyttösalasanoja. Käytettävän palvelun tietoturvakäytännöistä riippuen käyttäjä voisi käyttää samaa salausavainta sekä PC- että mobiilisovelluksessaan. Käyttäjällä voi olla laitteellaan asennettuna useiden eri valmistajien token-sovelluksia, mikä tekee niiden hallinnoinnista helppoa.

Fyysisiin tokeneihin kohdistuvista hyökkäyksistä on esimerkkinä RSA:n palvelimille vuonna 2011 tehty murto, jossa käyttäjäkohtaisia jaettuja salaisuuksia päätyi hyökkääjien haltuun. Hyökkääjillä oli ollut tiedossa myös kertakäyttösalasanojen generointiin käytetyn algoritmin tietoja, joten he onnistuivat generoimaan haltuunsa saamia avaimia vastaavien käyttäjien kertakäyttösalasanoja. Hyökkääjät onnistuivat tunkeutumaan tokeniensa avulla sotakalustotoimittaja Lockheed Martinin VPN:ään, mutta murto ehdittiin havaitsemaan ajoissa ja suuremmilta vahingoilta vältyttiin. Murto tuli RSA:lle erittäin kalliiksi, sillä käyttäjien tokenit piti uusia. Jos tokenit olisivat käyttäneet asymmetrisia avaimia, olisi hyökkääjien haltuun joutunut ainoastaan palvelimelta saadut julkiset avaimet, joita ei olisi pystytty hyödyntämään hyökkäyksessä. [47][51] Symmetrisiä salausavaimia hyödyntävien tokenien suuri ongelma onkin avaimien joutuminen vääriin käsiin. Käyttäjän kopio avaimesta tulisi säilyttää salattuna fyysisessä laitteessa tai token-sovelluksessa.

3.2.5 HMAC:iin perustuva kertakäyttösalasana

Kertakäyttösalasanat ovat ehdottomasti yksi yksinkertaisimmista ja suosituimmista kaksivaiheisen tunnistamisen muodoista verkkosovelluksissa. Palvelujen käytäntöjen yhtenäistämiseksi ja yhteensopivuuden parantamiseksi on kehitetty HMAC:iin (Hash Message Authentication Code) perustuva kertakäyttösalasana (engl. HMAC-based One-Time Password, HOPT), jota kuka tahansa laite- tai sovellusvalmistajan voi hyödyntää. Se voidaan myös sulauttaa esim. äly-korteille, USB-tikkuihin, SIM-korteille [6] tai sovelluksiin. HOTP toimii tapahtumasynkronisesti ja se lasketaan HMAC-SHA-1 algoritmilla, joka saa parametreiksi käyttäjäkohtaisen jaetun salaisuuden ja kasvavaan laskuriarvon. Tulos lisäksi typistetään ja muutetaan numeeriseen muotoon, jotta lopputuloksena saatu kertakäyttöinen salasana olisi käyttäjän helposti syötettävissä. HOTP-salasanat generoidaan kaavalla

$$HOTP(K, C) = Truncate(HMAC - SHA - 1(K, C)), \quad (1)$$

jossa K on käyttäjän ja palvelun välinen jaettu salaisuus, C on 8-tavuinen laskurin arvo ja Truncate on metodi, joka typistää HMAC-SHA-1 algoritmilla lasketun 160-bittisen

tiiviste käyttäjäystävällisempään muotoon. kuten 6-8 merkkiseksi numeroksi. HOTP pituuden tulisi olla vähintään 6- mutta mielellään 7- tai 8-merkkinen. [6]

HOTP-algoritmin käytössä on se selkeä etu, että sitä käyttävää järjestelmään voidaan hyödyntää monilta eri laitteelta tai sovelluksesta. Salasana generoituu samalla tavalla, kunhan yksityinen salaisuus on sama. Hyökkäykset HOTP:ta kohtaan ovat hankalia. Vaikka hyökkääjä saisi käsiinsä joukon jo käytettyjä kertakäyttösalasanoja, olisi vaikeaa määrittää kertakäyttösalasanojen generoimiseen tarvittu salainen avain. Hyökkääjän paras mahdollisuus olisi vaan yrittää arvata vaadittu kertakäyttösalasana brute forcella. [6] Esim. 6-merkkinen salasana voi saada vain miljoona erilaista arvoa, joten sen arvaaminen on hyvinkin mahdollista. Tästä syystä palvelun turvallisuuden varmistamiseksi tulee myös muistaa toteuttaa palvelinpuolen logiikka rajoittamaan käyttäjän tekemien arvausten määrä lyhyellä aikavälillä. Lisäksi järjestelmässä tulee varmistaa, etteivät jaetut salaisuudet joudu vääriin käsiin missään käytön elinkaaren vaiheessa [6]. HOTP ei sovellu kovin hyvin tilanteisiin, jossa käyttäjä käyttää sovellusta useammalta eri laitteelta. Koska salasanan generointi pohjautuu kasvavaan laskuriarvoon eivätkä sovellukset synkronoi laskurin arvoa keskenään, generoi toinen sovelluksista aina jo käytetyn salasanan.

HOTP:iin perustuvat kertakäyttösalasanat ovat käytössä esim. seuraavissa palveluissa. Tietoturvayhtiö Yubicon tarjoama Yubikey on USB-liitännällä varustettu fyysinen laite, joka generoi HOTP-salasanoja napin painalluksesta. Sitä voidaan käyttää kaksivaiheisessa tunnistamisessa esim. kirjauduttaessa Googlen tileille tai Windows-palvelimille ja -työasemille. Yubikeystä on myös versioita, jotka toimivat NFC-yhteydellä, joten ne eivät tarvitse suoraa kontaktia kohdelaitteen kanssa. Sama Yubikey voidaan rekisteröidä moneen eri palveluun ja monet palvelut hyväksyvät myös useamman Yubikeyn rekisteröinnin. Yubikeylle voi myös tallentaa monimutkaisia salasanoja, jolloin sitä voidaan käyttää perinteisessä käyttäjätunnus-salasana-tunnistamisessa. [40] Pelivalmistaja Blizzard on myös ottanut käyttöönsä käyttäjien pelitilien suojaamisessa HOTP-salasanoja generoivat tokenit sekä mobiilisovellukset, joiden avulla pelaajat voidaan tunnistaa kaksivaiheisesti. Fyysinen token generoi salasanoja napin painalluksesta ja mobiilisovelluksessa vastaava toiminto on toteutettu ohjelmallisesti. [41]

3.2.6 Aikaan perustuva kertakäyttösalasana

Aikaan perustuva kertakäyttösalasana (engl. Time-based One-Time Password, TOTP) on laajennus HOTP:stä ja se toimii nimensä mukaisesti aikasynkronisesti. Siinä satunnaisen salasanan generoinnissa käytettävä laskuriarvo on korvattu ajanhetken avulla. Tätä algoritmia voidaan käyttää monien eri verkkosovellusten, kuten VPN-yhteyksien, langattomien verkkojen ja verkkoasiointipalvelujen, tunnistamisessa. Algoritmin käytön vaatimuksen on, että asiakaspään sovellus ja palvelu pystyvät määrittämään Unix-ajan joka on ajanhetken tarkasteluun sovittu formaatti. [5]

TOTP lasketaan kaavalla

$$TOTP = HOTP(K, T), \quad (2)$$

jossa K on käyttäjän yksilöllinen jaettu salaisuus ja T on aikaleimaan perustuva ”järjestysnumero” tunnistuspyynnölle. T määritellään kuluneiden aikaintervallien lukumääräksi Unix-alkuajasta lähtien. Aikaintervalin sisällä generoidut kertakäyttösalasanat tuottavat saman tuloksen. Hyvänä aikaintervallina pidetään 30s aikaa, joka mahdollistaa hyvän käytettävyyden turvallisuutta turhaan rajoittamatta. Pyynnön lähetyksen ja käsittelyn viiveen takia palvelinpuolella voidaan joutua hyväksymään myös edellinen generoitu salasana. Hyökkäysaikaikkunan pienentämiseksi on suotavaa, ettei tämän vanhempia salasanoja hyväksytä. TOTP-salasanan generoimiseen tarvittava salainen avain voidaan säilöä sovellukseen tai laitteelle, jolle ei ole mahdollista tunkeutua. [5] Kunkin käyttäjän yksityinen salaisuus on myös palvelimen tiedossa, joten palvelinpuolella voidaan tarkistaa käyttäjän lähettämä TOTP [29]. TOTP soveltuu HOTP:ia paremmin useamman laitteen käytettäväksi, sillä aika kuluu tasaisesti eri laitteilla ja ne pysyvät synkronoituna.

TOTP tarjoaa suojaa kalasteluyrityksiä vastaan, sillä sen lyhyen voimassaoloajan takia hyökkääjän on toimittava nopeasti saatuaan haltuunsa yhden salasanan. Kertakäyttösalasanoja ei voi tämän takia myöskään säilöä ja myydä myöhemmin eteenpäin. [16, s.306] Kuten HOTP-salasanojen kohdallakin, brute force hyökkäykset ovat paras vaihtoehto TOTP-kertakäyttösalasanoja vastaan. [5] Koska TOTP:iin perustuva salasana kuitenkin vaihtuu väliajoin, on se HOTP:tä turvallisempi brute forcea vastaan. Järjestelmän tulisi kuitenkin rajoittaa arvauskertojen määrää tai tarkoituksenmukaisesti hidastaa niiden käsittelyä siten, että kaikkien vaihtoehtojen iteroiminen olisi tehotonta.

Markkinoilla on eri toimittajien, kuten Googlen ja Microsoftin tarjoamia, TOTP-kertakäyttösanojen generointiin soveltuvia sovelluksia. Microsoft Authenticator on Windows Phone -sovellus, joka hyödyntää TOTP-algoritmia kertakäyttösalasanojen generoimiseen puolen minuutin välein. Sovellusta voidaan käyttää kaksivaiheiseen tunnistukseen useilla eri Microsoftin tuotteilla. Kaksivaiheisen tunnistuksen käyttöönotto esim. sähköpostitilille voidaan tehdä sähköpostisovelluksen tietoturva-asetusten kautta. Käyttäjäkohtainen jaettu salaisuus näytetään QR-koodin muodosta, josta se on skannattavissa puhelimen kameralla Authenticator sovellukseen. QR-koodin mukana siirretään tarvittavat tilitiedot ja jaettu salaisuus, jota mobiilisovellus käyttää käyttäjäkohtaisten kertakäyttösalasanojen luomisessa. Jaettu salausavain voidaan liittää mobiilisovellukseen myös manuaalisesti tekstimuodossa. Kun kaksivaiheinen tunnistus on otettu käyttöön, vaatii sähköpostisovellus jatkossa käyttäjätunnuksen ja salasanan jälkeen vielä syöttämään kertakäyttösalasanan, jonka käyttäjä voi pikaisesti katsoa avaamalla Authenticator-sovelluksensa. Kirjautumispyyntö tarkistetaan palvelinpuolella generoimalla kirjatutuneen käyttäjän jaettua salaisuutta käyttäen kertakäyttösalasana ja vertailemalla tätä käyttäjän lähetettyyn salasanaan. Jos salasanat täsmäävät niin käyttäjä pääsee käyttämään palvelua. Microsoftin Authenticator-sovelluksessa on mukana aikakorjaus-

asetus, joka ottaa huomioon palvelimen ja käyttäjän puhelimen kellonaika-asetuksen eron. Tällöin mobiilisovellus toimii, vaikka käyttäjän puhelimen kello olisikin useamman aikaintervallin verran edellä tai jäljessä palvelimeen verrattuna. [49][53]

3.2.7 Push viestit

Push-viestit ovat mobiililaitteella näytettäviä lyhyitä viestejä, kuten ilmoituksia tai muistutuksia. Niitä voidaan lähettää sovelluksien tai palvelujen kautta ja ne voivat näkyä puhelimessa, vaikkei käyttäjällä olisi tietty sovellus edes auki. [32] Microsoftin push- viestit lähetetään Microsoftin Push Notification Servicen (MPNS) kautta, joka toimii Microsoft Azure pilvessä. Se muodostaa yhteyden Windows Phone -sovelluksen ja push-viestien data-sisällön tarjoajan välillä, joka on yleensä jokin web service. [30, s.405] Azure Notification Hubin avulla Push-viestejä voidaan lähettää myös iOS- ja Android-laitteille [56].

Push-viestejä voidaan esittää kolmella tavalla: Tile-, toast- tai raw-ilmoituksina. Tile- ja toast-tyyppisten viestien vastaanottaminen onnistuu vaan, jos mikään sovellus ei ole käynnissä. Toast-viestit päivittyvät puhelimen näytön yläreunaan tekstimuotoisina ja ne ovat yleensä kiireellisiä. Kun toastia klikataan, avautuu tietty sovellus. Toast-viesti voi välittää sovellukselle myös tietoa. Tile-viesti päivittyy etusivun sovellusikonien kuvia tai tekstiä ja sillä voitaisiin esim. päivittää säätietoja. Raw-viestejä voidaan lähettää ajettavalle sovellukselle ja ne ovat näkyvissä vain, jos sovellus on auki. [30, s.405] Raw-viestejä voisi hyödyntää tunnistuksessa myös kertakäyttösalasanan lähettämiseen.

Push-viestien etuna esim. tekstiviesteihin verrattuna on, että niiden lukeminen ei vaadi erillisiä toimia [63]. Tekstiviestit voivat toki näkyä osittain puhelimen yläpalkissa kuten Windows Phone 8.1:lla, jolloin käyttäjä ehtii lukea lyhyet viestit avaamatta tekstiviesti-sovellusta. Push-viesteistä ei aiheudu käyttäjille myöskään kustannuksia. Push-viestit käyttävät tunnistettavasta palvelusta erillään olevaa tiedonsiirtokanavaa, joten hyökkääjä ei pysty kaappaamaan niitä etukäteen, vaikka hän onnistuisi salakuuntelemaan käyttäjän ja palvelun välistä liikennettä. Azure Notification Hubin kautta lähetetyt push-viestit ovat melko edullisia varsinkin, kun viestien lähetysmäärät ovat pienet. [44] Kehitys- ja käyttöönottokustannukset olisivat voineet nousta suuriksi push-viestejä hyödyntäessä.

Push-viestejä hyödynnetään tunnistamisessa esim. Duo Mobilen ja Authyn tarjoamilla palveluilla. Duo Mobile käyttää kaksivaiheisessa tunnistuksessa käyttäjätunnuksen ja salasanan lisäksi yhtenä kirjautumisvaihtoehtona push-ilmoituksia. Palveluun syötetään ensin käyttäjätunnus ja salasana, jonka jälkeen käyttäjälle voidaan lähettää push-viesti mobiililaitteeseen. Tässä käyttötapauksessa käyttäjä kuittaa push-viestin, jonka jälkeen hänet on vahvasti tunnistettu. Duo Mobile on käytössä esim. Facebookin työntekijöillä [31]. Authy tarjoaa kaksivaiheiseen tunnistamisen push-viesteihin perustuvaa palvelua, joka on käytössä esim. pelien suoratoistopalvelussa Twitchissä. Authyn mobiilisovellus on tällä hetkellä saatavilla vain iOS- ja Android-laitteille [64].

3.2.8 Riskiin perustuva tunnistaminen

Riskiin perustuva tunnistaminen käyttää riskiprofilointia päätelläkseen onko käyttäjän tunnistuspyyntö epäilyttävä. Kullekin tunnistuspyynnölle annetaan riskiluokitus, ja jos se ylittää määrätyn raja-arvon niin käyttäjälle voidaan esittää turvallisuuskysymys tai heiltä voidaan vaatia ylimääräistä tunnistustekijää. Käyttäjät profiloidaan esimerkiksi sen perusteella mihin aikaan he yleensä kirjautuvat, mitä laitetta he käyttävät tai mitä tunnistustapaa he käyttävät. Jos jollain tunnistuskerralla olosuhteet eivät täsmää käyttäjälle tehdyn profiilin kanssa, kasvattaa se kyseisen tunnistuskerran riskiluokitusta. [12, s.21] Riskiprofilointia voidaan tarkentaa myös kirjautumistapahtuman jälkeen palvelun käytön yhteydessä esim. käyttäjän yrittäessä muuttaa tietoturva-asetuksia. Käyttäytymistietoa voidaan tallentaa esim. evästeiden avulla. [13, s.413]

Riskiin perustuva tunnistaminen on hyödyllinen tapa reagoida epäilyttäviin tilanteisiin, jotka johtuvat käyttäjän poikkeamista toimista. Riskiin perustuva tunnistaminen ei ole itsessään vahva tunnistusmuoto, mutta sitä voidaan hyödyntää tunnistamaan tilanteet, joissa kaksivaiheinen tunnistus olisi tarpeen. Siksi sitä usein hyödynnetäänkin muiden tunnistustekniikoiden ohella.

Riskiin perustuva tunnistus on käytössä esim. verkkopankkisovelluksissa ja sitä voidaan käyttää, kun asiakas yrittää esim. suorittaa normaalista poikkeavia tilisiirtoja ulkomaille. Ainakin Nordean verkkomaksutapahtuman voi joutua vahvistamaan puhelimitse, jos tilaa esim. vähemmän tunnetusta ulkomaisesta verkkokaupasta tavaraa. Vastaisuuden varalta Nordea antaa mahdollisuuden kuitata vastaavat ostotapahtumat tekstiviestillä. Riskiin perustuva tunnistaminen sopii erityisesti palveluihin, joilta vaaditaan korkeaa turvallisuustasoa myös harvoin esiintyvissä epäilyttävissä tilanteissa. Näitä voi olla verkkopankkien ostotapahtumisen lisäksi tilanteet, joissa hyökkääjä alkaa poistelemaan tunkeutumaltaan sosiaalisen median tililtään sisältöä tai yrittää sulkea koko tilin.

3.2.9 Biometrinen tunnistaminen

Biometrisessa tunnistamisessa käyttäjän identiteetti pyritään varmistamaan ottamalla hänestä jokin mittaustulos ja vertaamalla sitä aiemmin otettuun vastaavaan mittaustulokseen [18, s.128-130]. Biometriset tunnisteet voidaan jakaa fyysisiin ja käytökseen perustuviin tunnisteisiin. Fyysisiä tunnisteita ovat sormenjäljet, ääni, silmän iiris tai verkkokalvo, suonikuviot, kämmenen tai sormen geometria tai vaikka korvan muoto. Käytökseen perustuvia tunnisteita voivat olla hiiren käyttötekniikka, näppäinpainallusten viiveet, allekirjoitus tai mobiililaitteen asento laitetta käytettäessä. [8, s.109][52]

Biometristä tunnistusta varten tarvitaan aina näytteiden keräämisen takia joku siihen erityisesti soveltuva laite, sensori [13, s.598] tai käytökseen perustuvassa tunnistamisessa erityistä sovelluslogiikkaa. Käyttäjältä otetut näytteet eivät aina ole samanlaisia, vaan ne voivat muuttua olosuhteiden mukaan. Tämän takia tunnistamisessa riittää, että näyte

on riittävän lähellä tiedossa olevaa näytettä. Rekisteröintivaiheessa käyttäjältä voidaan ottaa useampi näyte, joiden keskiarvo tallennetaan tietokantaan. Tunnistamisessa käyttäjältä otetaan uusi näyte ja tunnistuspyyntö hyväksytään, jos näyte on riittävän lähellä vertailunäytettä. Biometrisessä tunnistuksessa suorituskykyä arvioidaan sen perusteella, kuinka usein oikeutettu käyttäjä tulkitaan tunkeutujaksi (engl. False Reject Rate, FRR) ja kuinka usein tunkeutujaa erehdytään luulemaan oikeutetuksi (engl. False Accept Rate, FAR). [19, s.373] Jos liian useaa tunkeutujaa luullaan lailliseksi, ei järjestelmä ole kovin turvallinen. Jos monet laillisista käyttäjistä tulkitaan tunkeutujiksi, voivat he kokea järjestelmän käytön rasittavaksi. [75, s.156] Biometrisen tunnistamisen turvallisuutta säädellään tunnistuskriteerejä tiukentamalla, mikä valitettavasti vaikuttaa käytettävyyden heikentymiseen. FAR:n tulisi olla pieni, jotta järjestelmän tulisi olla turvallinen. Toisaalta FRR:n tulisi olla pieni, jotta järjestelmä olisi käytettävä. [75, s.156]

Biometristen tunnisteiden etuna on, ettei niitä voi hävittää tai unohtaa [18, s.128-130]. Toisaalta, jos biometrinen data joutuu vääriin käsiin, niin käyttäjä ei pysty vaihtamaan tunnistettaan toisin kuin esim. salasanojen kohdalla. [13, s.598] Kaikki biometriset järjestelmät kärsivät ainakin jonkin verran virhetulkinnoista, eli FAR tai FRR on liian suuri. [18, s.128-130] Biometristen tunnistustapojen käyttöönotto on melko kallista, minkä ne takia niitä ei käytetä laajalti [11, s.36]. Koska fyysiset biometriset tunnisteet eivät muutu ajan myötä, voidaan niitä käyttää käyttäjän tunnistamisessa [11, s.34]. Fyysisten biometriikoiden käyttö vaatii sopivaa laitteistoa ja sensoreja tunnistustavan käyttöön. Jos laitteessa ei ole riittävän tarkkoja sensoreja, ei tunnistamisesta saada riittävän tarkkaa, jolloin turvallisuus tai käytettävyys voi kärsiä. Tunnistamiseen vaikuttavat myös olosuhteet. Esim. laitteen ja sormien likaisuus ja kulumat vaikeuttavat sormenjälkitunnistusta [8, s.179], kuvatunnistuksessa valaistus vaikuttaa kuviin [8, s.182] ja ääneen perustuvassa tunnistuksessa taustamelu aiheuttaa virheitä. Jos laitetta on mahdollista käyttää valvomattomassa ympäristössä, on tunnistus altis tunnisteiden väärentämiselle. Kameralle voidaan esim. näyttää kuvaa käyttäjän kasvoista [8, s.109] tai käyttäjän äänestä nauhoitettu ääninäyte voidaan soittaa laitteelle [18, s.128-130]. Käytökseen perustuvassa tunnistamisessa on etuna, että tunnistamista voidaan tehdä jatkuvasti käytön yhteydessä, eikä se vaadi lisälaitteistoa. [75, s.156] Omien tunnistusalgoritminen toteutus on tosin työlästä ja varsinkin käyttäjäkohtaisten erojen ja vaihtelevan käyttötilanteiden takia tunnistuksen tarkkuus vaihtelee.

Mobiililaitesovelluksia ajatellessa sopivia fyysisiä biometriikoita olisivat vain kasvon tai silmän kuvat, sormenjäljet tai puheääni. Sopivia käytökseen perustuvia biometriikoita olisivat kirjoitusnopeus, näppäinkomentojen viiveet, laitteen asento tai tyypillisiin käyttötapauksiin tai sovelluksessa navigointiin perustuva tarkkailu. Fyysisten biometriikoiden ongelmaksi voi muodostua, että käyttäjän tunnistetieto on kaikille nähtävissä. Näin ollen hyökkääjä voi hämätä kasvotunnistusta näyttämällä kameraan kuvaa hyökkäyksen kohteena olevasta käyttäjästä. Biometrisessa tunnistamisessa tulisi myös huomioida tilanteet, joissa palvelua käytetään eri laitteilta. Esimerkiksi älypuhelimen ja

tabletin sensorien antamat mittaustulokset voivat poiketa ratkaisevasti toisistaan johtuen laitteiden ja näyttöjen erilaisesta koosta ja otteesta, jolla laitteista pidetään kiinni. Lisäksi käytökseen perustuvan tunnistamisen pitäisi pystyä päivittämään käyttäjän profiilia käytön myötä, sillä esimerkiksi navigoiminen ja toimintojen suorittaminen nopeutuu käytön myötä.

Deutsche Bank on testaamassa mobiililaitteilla fyysisiä ja käytökseen perustuvia biometriikoita hyödyntäviä tunnistamistapoja, jotta mobiilimaksamista saataisiin turvallisemmaksi ja suurempia maksuja voitaisiin sallia. Tekniikalla seurataan n. 50 eri tekijää, jotka liittyvät esim. numeronäppäimistön painamisvoimakkuuteen, puhelimen asentoon, käyttäjän sijaintiin, kasvon ilmeisiin ja sormenjälkeen. Järjestelmä pystyy tunnistamaan myös käyttäjän asennon ja se pystyy ottamaan huomioon tilanteet, joissa käyttäjä ei pysty käyttämään laitettaan normaalisti – kuten käyttäessään laitetta vasemmalla kädellä oikean käden ollessa murtuneena. Järjestelmä on ollut kokeilukäytössä ja sen suunnitellaan tulevan pian käyttöön pankin 10 000 työntekijälle laajempaa pilotointia varten. Pankin johto tulee päättämään myöhemmin järjestelmän laajemmasta käyttöönotosta. [52]

Mastercardilta oli viime vuonna Yhdysvalloissa ja Hollannissa kokeilukäytössä ”selfieihin” perustuva tunnistamistapoja, jota voidaan käyttää salasanojen korvaamiseen verkkomaksuissa. Käyttäjän pitää edelleen syöttää myös aiemmin vaaditut luottokorttitiedot maksun yhteydessä, ja kuvaan perustuvaa tunnistamista käytetään vain epäilyttävissä tapauksissa. Kuvaan perustuva tunnistaminen tapahtuu PC:lle tai mobiililaitteelle ladattavan sovelluksen avulla, jonka kautta käyttäjän itsestään ottama kuva lähetetään palveluun. Vaikka tietoturva-asiantuntijat ovat huomauttaneet, että myös kuvaan perustuvaa tunnistamista voidaan huijata, on Mastercard ollut vakuuttunut, että sen toteuttamat tietoturvamekanismit havaitsevat epäilyttävät käytön yritykset. Kuvaa ottaessa käyttäjän tulee räpäyttää silmiään, millä voidaan erottaa tilanteet, joissa kameralle yritetään näyttää käyttäjän kuvaa. Tämä toteutustapa ei kuitenkaan ole täysin aukoton, sillä kuvia voidaan manipuloida tekemällä kuvasta animaatio kuvankäsittelyn avulla. Mastercardilta saamien tietojen mukaan 92% kokeilukäyttäjistä piti uutta järjestelmää salasanoja parempana. Mastercardin on tarkoitus laajentaa uuden tunnistustavan käyttöönottoa kesällä 2016 Kanadaan ja useampaan Euroopan maihin, joihin mukaan lukeutuu myös Suomi. [58][59]

3.2.10 Sertifikaatit

PKI (Public Key Infrastructure) tukee digitaalisten sertifikaattien käyttöä hallitsemalla avaimia ja tarjoamalla julkisen avaimen sertifikaatteja [76, s.54]. Sertifikaatti on joukko julkisesti saatavilla olevia käyttäjään liittyviä attribuutteja ja julkinen avain, jonka luotettava taho, sertifikaattivarmentaja (engl. Certification Authority), on todennut oikeaksi [11, s.26]. Sertifikaatit sitovat allekirjoittajan identiteetin hänen julkiseen avaimeensa. Varmentajat usein muodostavat hierarkian, jossa ylemmän tason varmentaja voi jakaa

sertifikaatteja alemmalle tasolle [76, s.54]. Varmentaja luo käyttäjäkohtaiset sertifikaatit kullekin käyttäjälle, toimittaa ne käyttäjille ja säilyttää palvelimella kopiot kunkin käyttäjän sertifikaatista [20, s.162]. Sertifikaatit ovat voimassa, kunnes niiden voimassaoloaika umpeutuu tai ne perutaan (engl. revoke) voimassaoloaikanaan. Sertifikaatti kumotaan, kun käyttäjän identiteetti vaihtuu tai käyttäjän yksityinen avain vaarantuu. [76, s.54] Sertifikaatin allekirjoituksen validoinnissa tarkistetaan myös varmentajasertifikaattien luotettavuus, voimassaoloajat ja perumistiedot [76, s.54]. Sertifikaatteihin perustuva tunnistaminen on vahvempi kuin salasanoihin perustuva, sillä siinä käyttäjä omistaa jotain tietämisen sijaan. [14][19, s.366]

Sertifikaatti sisältää julkisen avaimen, sertifikaatin omistajan, voimassaoloajan, sertifikaatin myöntäjän, sarjanumeron, sertifikaatin yksityisellä avaimella tehdyn digitaalisen allekirjoituksen ja allekirjoitukseen käytetyn algoritmin tiedot. [20, s.171] Koska sertifikaatit sisältävät tiedon käyttäjän identiteetistä, voidaan niitä käyttää käyttäjän tunnistamisessa. Jokaisella sertifikaatilla on sen mukana tulevaan julkiseen avaimeen matemaattisesti sidottu yksityinen avain, jota ei lähetetä sertifikaatin mukana. [11, s.26]

Sertifikaatteja on tarjolla palvelimien ja asiakkaan tunnistamista varten. X.509-sertifikaattia, jota palvelin käyttää todistaakseen identiteettinsä asiakkaalle tai asiakkaan puolesta toimivalle käyttäjäagentille (engl. user agent) kutsutaan palvelinsertifikaatiksi. X.509-sertifikaattia, jota asiakas käyttää todistaakseen identiteettinsä palvelimelle kutsutaan asiakassertifikaatiksi. [20, s.171] TLS:n kättelyprosessissa osapuolien tunnistamisessa käytetään X.509-sertifikaatteja. Palvelin lähettää julkisella avaimellaan varustellun X.509-sertifikaattinsa asiakkaan selaimelle. Selain validoi sertifikaatin luottamaansa sertifikaattivarmentajalistaa vasten. Jos sertifikaatti on luotettujen varmentajien listalla ja sertifikaatti kuuluu kohteena olevalle verkko-osoitteelle, niin selain on vakuuttunut palvelimen identiteetistä. TLS sallii myös molemmin puoleisen tunnistamisen: Selain voidaan asettaa lähettämään X.509-asiakassertifikaatti palvelimelle TLS-kättelyn yhteydessä, jolloin palvelin voi varmistua käyttäjän identiteetistä. [20, s.171] Tällä hetkellä X.509-standardin uusin versio on X.509 v3. [14][19, s.258]

3.2.11 Nordea Tunnuslukusovellus

Mobiilisovellusten yleistymisestä on esimerkkinä Nordean julkaisema tunnuslukusovellus, joka on ollut saatavilla kesästä 2015 lähtien Applen, Androidin ja Windows Phonen sovelluskaupoissa. Sovelluksen päämääränä on ollut korvata vanhat tunnuslukukortit ja sitä voidaan käyttää sekä kirjautumiseen että maksutapahtumien vahvistamiseen. Nordean verkkopankin lisäksi sovellus käy myös verkko-ostosten maksamisessa ja tunnistuksessa muiden palveluntarjoajien palveluihin. [50]

Esimerkiksi Nordean verkkopankissa Windows Phone 8.1 -tunnuslukusovelluksen käyttö toimii seuraavasti. Käyttäjä avaan puhelimellaan tunnuslukusovelluksen, joka jää odottamaan tunnustustapahtuman vahvistuspyyntöä. Käyttäjä siirtyy selaimella verkko-

pankkiin ja valitsee tunnistustavaksi tunnuslukusovelluksen ja syöttää kenttään asiakasnumeronsa. Tämän jälkeen mobiilisovellus saa tiedon vahvistettavasta tunnistuspyynnöstä, jonka käyttäjä kuittaa aktivoinnin yhteydessä määrittämällään nelinumeroisella PIN-koodilla. Jos PIN-koodi on oikein, se hyväksytään ja sovellus jää odottamaan maksujen vahvistuspyyntöjä. Käyttäjä suorittaa maksun verkkopankissa, jonka jälkeen maksun tiedot (maksun summa ja tilinumero) tulevat näkyviin mobiilisovellukseen ja käyttäjä kuittaa maksun syömällä PIN-koodinsa uudelleen. [48][50]

Nordean asiakkaana otin tämän sovelluksen kokeilukäyttööni selvittääkseni, onko siitä tunnuslukulistan korvaajaksi. Sovelluksen kokeilukäytössä Windows Phone 8.1 -laitteella havaitsin muutamia ilmeisiä käytettävyysongelmia, joista muiltakin käyttäjiltä oli tullut negatiivista palautetta. Esimerkiksi tekstiviestillä vastaanotettavan aktivointikoodin syöttäminen sovellukseen oli toteutettu hankalasti. Vaikka koodin pystyikin kopioimaan tekstiviestistä leikepöydälle, ei sitä voinut liittää sovellukseen, koska liitäpainike puuttui sovelluksesta. Ilman toista laitetta tai kynää ja paperia koodin lisääminen sovellukseen siis olisi ollut hyvin tuskallista. Lisäksi tilin luonnin yhteydessä luotiin parikin eri laite- tai käyttäjätunnusta, mutta verkkopankkiin kirjautuessa vaadittiinkin edelleen vanhaa Nordean asiakasnumeroa. Kolmanneksi ongelmaksi koin sovelluksen aloitus- ja vahvistusnäkymät, jotka olivat täysin samanlaiset. Sovelluksen sulkemisen ja uudelleenavaamisen yhteydessä minulle jäi epäselväksi vaaditaanko minulta vielä kirjautumista vai onko sovellus valmis vastaanottamaan maksujen vahvistuspyyntöjä. Jos mobiilimaksamiselle tulee yllättävää tarvetta esim. junalippuja ostettaessa, on sovelluksen käyttäminen kätevää. Tällöin tunnuslukulistaa ei tarvitse kantaa mukanaan, vaan yllättävän maksun vahvistus onnistuu mobiilisovelluksen avulla. Verkkopankkiin kirjautuminen mobiilisovelluksen avulla vaati tunnuslukukortin tavoin käyttäjältä edelleen asiakasnumeron muistamista, mutta tunnuslukukortin vaihtuvan koodin sijaan kiinteää PIN-koodia. Haittapuolena onkin se, että huonomuistisien käyttäjien voi olla vaikea muistaa tunnistamiseen vaadittuja 8-numeroista käyttäjätunnusta ja 4-numeroista PIN-koodia.

4. WINDOWS PHONE ALUSTAN TURVALLISUUSRATKAISUT

Tässä luvussa käydään läpi hyvän kokonaiskuvan saamiseksi eri mobiilialustojen turvallisuustoimintojen vertailu, jonka jälkeen käsitellään tarkemmin Windows Phone -alustan tietoturvaratkaisuja

Windows Phone -alustaa kuulee välillä sanottavan muita alustoja turvallisemmiksi. Galen Gruman vertailee vuonna 2015 InfoWorldiin kirjoittamassaan artikkelissaan Mobile security: iOS vs. Android vs. BlackBerry vs. Windows Phone mobiilialustoilla käytössä olevia turvallisuusominaisuuksia ja tietoturvakäytäntöjä [57]. Kokosin hänen artikkelistaan taulukkoihin 1 ja 2 mielestäni oleellisimpia turvallisuuteen liittyviä ominaisuuksia, jotka on jaoteltu kahteen eri kategoriaan: Toiminnot, jotka vaatii EAS käytön (Exchange Active Sync) ja toiminnot, jotka ovat käytettävissä alustoilla natiivisti. Erikoistapauksien merkitykset on esitetty taulukon jälkeen ja selitetty lyhyesti myös vertailun yhteydessä.

Taulukko 1. *Mobiilialustojen turvallisuusvertailu EAS, perustuu lähteeseen [57]*

	Android 4,5,6	**iOS 7,8,9**	**Blackberry**	**Windows Phone 8-8.1**
EAS (Exchange ActiveSync)				
Laitteen salaus	+	+	+	+
Salauksen pakotus	+ (*)	+	+	+
Salasanapolitiikka	+	+	+	+
Wi-Fi käytön esto	-	MDM	MDM	+ (8.1)
Bluetooth käytön esto	-	MDM	MDM	-

MDM = Jos Mobile Device Management on käytössä
8.1 = Käytössä vain Windows Phone 8.1:llä
* = Vain, jos asennettu yrityssovellusosioon

EAS: ollessa käytettävissä Android, iOS, Blackberry ja Windows Phone -laitteet tukevat laitteen salausta, ja laitteen salaus voidaan pakottaa käyttöön kaikilla muilla laitteilla paitsi Androidin yrityssovellusosion ulkopuolelle. Salasanan pituuteen ja monimutkaisuuteen liittyvät politiikat voidaan ottaa käyttöön kaikilla alustoilla. Wi-Fi käytön esto ei ole mahdollista Androideilla eikä Windows Phone 8:lla, ja Bluetooth käytön esto ei ole mahdollista Androideilla eikä Windows Phone:lla. [57]

Taulukko 2. *Mobiilialustojen turvallisuusvertailu, perustuu lähteeseen [57]*

	Android 4,5,6	**iOS 7,8,9**	**Blackberry**	**Windows Phone 8-8.1**
Natiivit ominaisuudet				
Laitteen salaus	AES 128 (*)	AES 256	AES 256	AES 256
VPN	+	+	+	+ (8.1**)
Rajoita sovelluskauppasovelluksia	+	+	+	+
Allekirjoitetut sovellukset vaaditaan	-	+	+	+
Yrityssovellusten ja datan pyyhkiminen	+ (***)	+	+	+ (8.1)
Yrityssovellusten pakotettu päivitys	+ (***)	+	+	+
Hiekkalaatikointi	+	+	+	+
Kopioi ja liitä -esto	+	+	+	+ (8.1)
Estä pilvitallennus	-	+	+	+ (8.1)
Data roaming esto	-	+ (9)	-	-

8.1 = Käytössä vain Windows Phone 8.1:llä
9 = Valvotuilla iOS 9 laitteilla
* = Käyttäjä voi poistaa käytöstä, ei tuettu kaikilla malleilla
** = VPN käyttö on rajoitettu
*** = Jos tallennettuna turvattuun säilöön

Natiiveista toiminnoista laitteen salausta tukevat kaikki muut laitteet, paitsi Androideilla salaus toimii vain osalla malleista, se on toteutettu heikommalla EAS 128-salauksella ja se on otettavissa pois käytöstä käyttäjän toimesta. VPN toimii kaikilla alustoilla, mutta vain osittain Windows Phone 8.1:llä. Kaikilla alustoilla voidaan rajoittaa sovelluskauppasovellusten lataamista, mutta allekirjoitettuja sovelluksia ei vaadita Androidilla. Yrityssovellukset ja niiden käyttämä data voidaan pyyhkiä laitteelta muilta paitsi muille kuin työympäristöön asennetuilta Androideilta ja Windows Phone 8:lta. Yrityssovellusten pakotettu päivitys on mahdollista kaikilla muilla paitsi työympäristön ulkopuolelle asennetuille Android-sovelluksille. Kaikki alustat käyttävät hiekkalaatikointia. Kopioinnin ja liittämisen esto on mahdollista muilla paitsi Windows Phone 8:lla. Pilvitallennusta voidaan rajoittaa iOS:llä, Blackberryllä ja Windows Phone 8.1:llä. Data roaming-esto on mahdollista vain valvotuilla iOS 9 -laitteilla. [57]

Edellä käsiteltyjen tietoturvaominaisuuksiin liittyvien soveltuvuuksien perusteella ei voida sanoa, että Windows Phone -alusta olisi joukon turvallisin. Suuri osa tietoturvatoiminnosta puuttui vielä varsinkin Windows Phone 8 -alustalta. iOS ja Blackberry tukivat useampia tietoturvaominaisuuksia kuin Windows Phone, kun taas Android tuki tietoturvaominaisuuksia huonoiten. Pahiten Androidin tietoturvaan ovat vaikuttaneet sovelluskauppaan päätyneet haitalliset sovellukset sekä toimintamalli, jossa laitevalmistajat vastaavat itse käyttöjärjestelmäpäivityksien julkaisuista [57]. Windows Phone 8 ja 8.1 -alustan käyttäjäkunta on pienempi, mistä voi osaksi johtua haittaohjelmien kehittäminen suositummille alustoille. Käyttäjäkunnan koko vaikuttaa myös haluun toteuttaa

sovelluksia kyseiselle alustalle, mikä on huono varsinkin, jos hyödylliset tietoturvaa edistävät sovellukset jää tämän takia kehittämättä.

Käsitellään seuraavaksi tarkemmin Windows Phone -alustan tietoturvaratkaisuja. Windows Phonen mukana tulevat sovellukset on kehitetty luotettavien OEM-valmistajien (Original Equipment Manufacturer) toimesta ja sovelluksissa on voitu käyttää sellaisia rajapintoja ja resursseja, jotka eivät ole perinteisen sovelluskehittäjän käytettävissä. Sovelluskehittäjällä ei esimerkiksi ole pääsyä SIM-kortin tietoihin, jota voisi käyttää vaikka laitteen tunnistamisessa [25]. Kolmannen osapuolen Windows Phone -sovelluksilla ei ole pääsyä puhelimen tiedostojärjestelmään, missä ovat myös turvallisuusnäkökohdat taustalla [61, s.6-7,16]. Rajapinta tarjoaa kuitenkin mahdollisuuden pyytää puhelimen laitenumeron, jota voidaan käyttää samaan käyttötarkoitukseen. Tässä ratkaisussa on se etu, että vihamielinen sovelluskehittäjä ei voi kerätä liian yksityistä laitetietoa käyttäjiltä sovelluksessaan turhaan. Huonona puolena on vastaavasti se, että tunnistussovellusta kehittävä henkilö ei pysty hyödyntämään vastaavia tunnistetietoja sovelluksessaan.

Windows Phone -alusta sisältää useita yksityisyyttä ja laitekohtaista tietoturvaa edistäviä ominaisuuksia, joihin kuuluvat hiekkalaatikointi, SafeBoot-toiminto, sovellusten hyväksyttämisprosessi ja BitLocker. Sovellukset on eristetty toisistaan hiekkalaatikoinnin avulla, eikä niillä siis ole pääsyä yleisiin tai toisten sovellusten resursseihin. Sovellukset pystyvät kommunikoimaan keskenään ainoastaan ennalta määrättyjen kommunikaatiokanavien ja tietotyyppien avulla. [37] Kaikki sovellukset on myös pakotettu toimimaan matalammalla hiekkalaatikkokerroksella, jolloin haitallisilla sovelluksilla ei ole pääsyä esim. rekistereihin tai lukittuihin rajapintoihin. Sovelluksien lataaminen puhelimen muistiin on vaikeaa SafeBoot -toiminnon takia. Se varmistaa puhelimen käynnistysvaiheessa, että vain laillisesti Microsoftin kauppapaikasta ladatut ja allekirjoitetut sovellukset ladataan. [35] Windows Phonen 8.1:n koko tallennustila ja sovelluksien omat tietovarastot on mahdollista salata BitLockerin avulla. BitLockerin käyttö on mahdollista vain yrityskäyttäjille ja sen pystyy ottamaan käyttöön ActiveSyncin tai laitehallintakäytäntöjen avulla. [36]

Hiekkalaatikointi ja sovelluskohtainen salattu tietovarasto lisäävät merkittävästi sovellusten tietoturvaa, sillä se vaikeuttaa toisen sovelluksen tietojen varastamista laitteelta käsin. SafeBoot -toiminnon ansiosta haitallisten sovellusten ajaminen laitteella vaikeutuu, mikä myös vaikeuttaa laite- tai sovelluskohtaisten tietojen varastamista. BitLockerin edut tulevat siinä vaiheessa esiin, kun laite on joutunut vääriin käsiin ja hyökkääjä pystyy vapaasti tekemään sille mitä vain. Jos laitteelle on tallennettuna salaista tietoa, on sitä entistä vaikeampaa saada selville kryptauksen takia.

Windows 10 ja sen mobiiliversion Windows 10 Mobile ovat tuoneet mukanaan uudet tunnistamistekniikat, Windows Hellon ja Microsoft Passportin. Windows Hello on Windows 10 alustojen sisäänrakennettu biometrinen tunnistustapa, johon kuuluu sormenjälki- ja kasvojentunnistus. Näiden avulla käyttäjä voi avata Windows laitteidensa

lukituksen. Microsoft Passport on Windows 10:n kaksivaiheinen tunnistamistapa, joka käyttää laitteeseen sidottua salattua avainta sekä Windows Helloa. Microsoft Passport muistuttaa toimintatavaltaan toimikortteja, mutta sen käyttöönotto ei vaadi kortinlukijoita tai PKI:ta sertifikaattien hallintaa varten, vaan avaimet säilötään turvallisesti laitteella tai sovelluksen tietovarastossa. Microsoft Passport voidaan ottaa käyttöön käynnistysasetuksista PIN-koodin määrityksen jälkeen ja tunnistustapa on käytettävissä käyttäjän Microsoft tai Azure Active Directory-tileillä. PIN-koodin määrittämisen jälkeen Windows Hello voidaan ottaa käyttöön vaihtoehtoisena tunnistustapana. Microsoft Passportia voidaan käyttää omaan sovellukseen kirjautuessa tai myös kirjauduttaessa mobiilisovelluksen käyttämään palveluun. [83] Microsoft Passportin toimintaperiaate mobiilisovelluksella kirjauduttaessa palveluun on esitetty kuvassa 4.

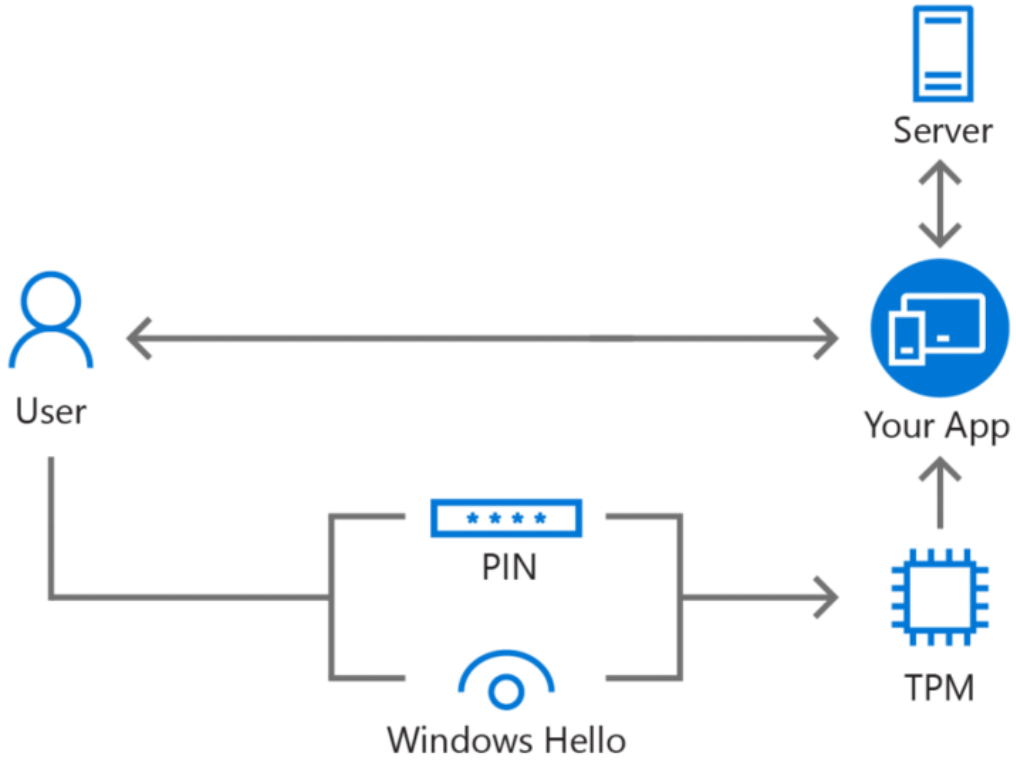
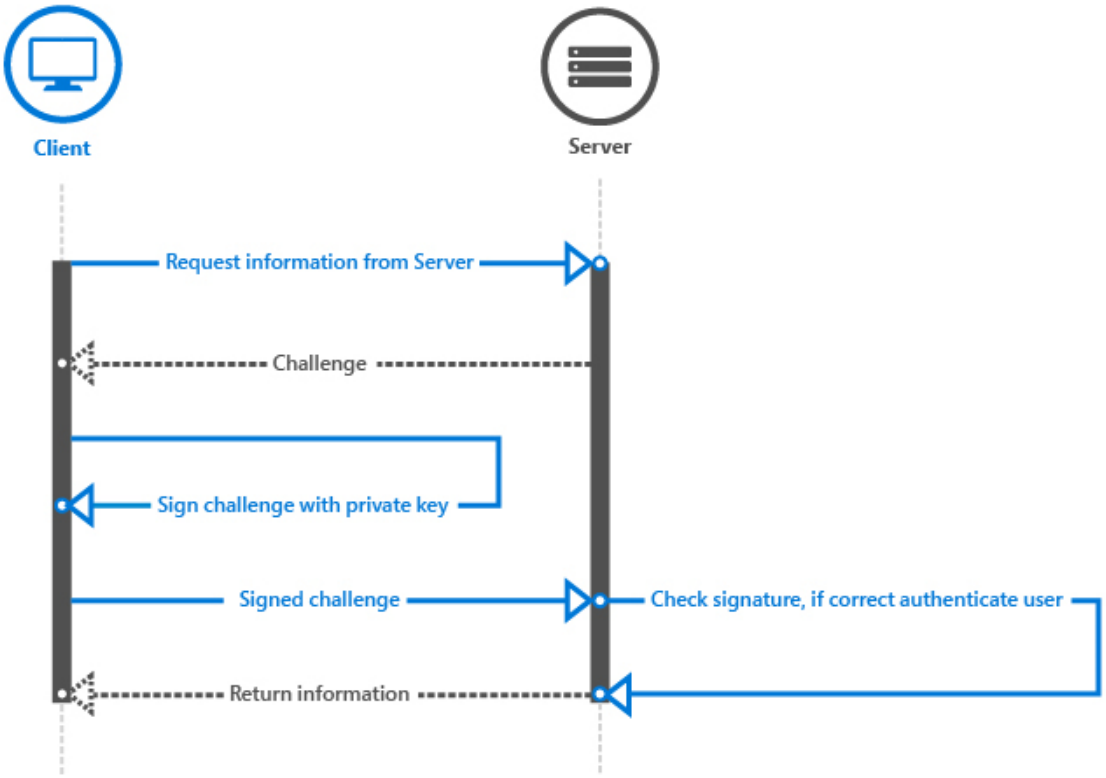


Figure 1 How Microsoft Passport works with the other components

Kuva 4. *Microsoft Passportin käyttö omassa sovelluksessa [83]*

Microsoft Passportissa ratkaisevana osana turvallisuudesta vastaa Trusted Platform Module (TPM), joka avustaa avainten säilyttämisessä laitepohjaisessa tunnistamisessa. Microsoft Passport tunnistaa käyttäjän PIN-koodin tai Windows Hellon avulla, jonka jälkeen se käskee TPM-sirua generoimaan käyttäjäkohtaiset julkiset ja yksityiset avaimet. Yksityinen avain säilötään turvallisesti TPM:ään, eikä sitä siihen ole suoraa pääsyä. Yksityistä avainta voidaan kuitenkin hyödyntää tunnistamisessa Microsoft Passport

API:n (Application Programming Interface) kautta. Julkista avainta käytetään tunnistamispyynnöissä käyttäjien tunnistamisessa ja viestien alkuperän varmentamisessa. Palvelinpuolen käyttäjätietorakenteen tulee sallia saman käyttäjän useammat eri avaimilla toimivat päätelaitteet. Kun käyttäjä rekisteröityy sovelluksessa, lähettää se palvelinpuolelle käyttäjän tietojen mukana käyttäjälle luodun julkisen avaimen. [83] Session ensikirjautumisen etenemin on esitetty kuvassa 5.



Kuva 5. *Windows 10 sovelluksen tunnistamin Microsoft Passportilla [83]*

Kun käyttäjän sovellus halutaan tunnistaa tämän jälkeen, lähetetään käyttäjälle haaste, jonka sovellus allekirjoittaa yksityisellä avaimellaan. Kun allekirjoitettu haaste lähetetään takaisin palvelimelle, voidaan sen aitous tarkistaa palvelimen tiedossa olevan julkisen avaimen avulla. Tämän jälkeen palvelin voi generoida käyttäjää varten sessiotokenin, joka välitetään palvelupyyntöjen mukana käyttäjän tunnistamisessa loppusession ajan. [83] Tällä hetkellä suosituista sovelluksissa esim. salasanojen säilyttämiseen tarkoitetussa LastPassissa ja tiedostojen säilytykseen tarkoitetussa DropBoxissa on Windows Hello käytössä monivaiheisessa tunnistamisessa. [80][81]

Windows Hello on hyödynnettävissä ainoastaan, jos käyttäjän laitteessa on sormenjäljen lukijalaite tai silmän iiriksen kuvaamiseen soveltuva kamera. Microsoft on julkaissut kaksi uutta lippulaivamallia - Lumia 950 ja Lumia 950 XL, joissa on iiriksen kuvaamisen soveltuvat infrapunakamerat. [84] Windows Hello on näin ollen tuettu ainakin näillä

malleilla. [46] Nykyisistä Windows 8.1 puhelimista tarvittava laitteisto puuttuu, joten Windows Hello ei ole niillä käytettävissä. [42] Windows 10 Mobile biometriset tunnistustavat eivät käyttäjäarvostelujen mukaan ole olleet tarpeeksi sujuvia, vaan puhelinta joutuu pitämään paikallaan muutaman sekunnin verran lähellä kasvoja. Tähän voi tulla jatkossa parannuksia, jos Microsoft onnistuu optimoimaan tunnistusalgoritmeja ja täten nopeuttamaan tunnistamisprosessia. Microsoft myi hiljattain puhelinliiketoimintansa, joten se ei enää julkaise uusia Lumia-puhelimien Windows 10 Mobile -alustalle. Microsoft jatkaa kuitenkin Windows 10 Mobile -alustan sekä -puhelinmallien kehitystä ja OEM-laitevalmistajien tukemista. [17]

5. TOTEUTETTU JÄRJESTELMÄ

Tässä luvussa on kuvattu järjestelmään liittyvä vaatimusmäärittely sekä siihen pohjautuvat suunnitteluratkaisut. Lisäksi kerromme tarkemmin mobiilisovelluksen ja tiedonhakupalvelun toimintaperiaatteista ja muista niihin liittyvistä toteutusratkaisuista.

5.1 Vaatimusmäärittely

Vaatimusmäärittely tapahtui pääosin projektin alkuvaiheessa asiakkaan käytössä olevan järjestelmään perehtymisen ja työn tehtävänannon yhteydessä. Yleiset vaatimukset sovittiin työtä rahoittaneen yksikön yhteyshenkilön ja parin muun teknisen asiantuntijan avustuksella. Vaatimukset muodostettiin sovellusalustan sekä alkuperäisen asiakkaan käytössä olevien järjestelmien ja tarpeiden perusteella. Asiakkaan käytössä olevat järjestelmät vaikuttivat vahvasti järjestelmän tietoturvavaatimuksiin. Käytettävyyteen ja hallinnointiin liittyvät vaatimukset muodostuivat asiakkaan tarpeiden pohjalta. Kustannusvaatimukset tarkennettiin kehitystyön aikana, vaikkei absoluuttista hintatasoa määriteltykään. Vaatimuksiin tuli pieniä tarkennuksia ja muutoksia teknisen toteutuksen aikana, kun järjestelmästä päätettiin tehdä prototyyppi asiakkaan luovuttua hankkeesta.

Alustan valintaan vaikutti suurelta osin se, että yrityksemme kehittää järjestelmiä Microsoftin tuotteilla ja tekniikoilla. Koska työn aloitusvaiheessa Windows Phone 8 oli uusin Microsoftin puhelinmalli, valittiin se sovellusalustaksi. Windows Phone 8.1 julkaisun jälkeen sovellus päivitettiin Windows Phone 8.1-yhteensopivaksi versioksi.

Kehitettävälle sovellukselle asetettiin seuraavat vaatimukset, joihin liittyvät kuvaukset ja ratkaisut on kuvattu omissa kappaleissaan. Käyttämämme vaatimusmäärittely perustuu pääosin luvun 2.2 turvallisuuteen, käytettävyyteen ja kustannuksiin perustuva luokitukseen, mutta sitä on hieman tarkennettu omia tarpeita silmälläpitäen.

1. Järjestelmän tulee hyödyntää vahvaa tunnistamista
2. Järjestelmän kehitys- ja ylläpitokustannuksien tulee olla kohtuulliset
3. Tunnistamistavan sovelluttava muillekin mobiilialustoille
4. Järjestelmän uusien käyttäjien rekisteröinnin tulee olla helppoa.
5. Järjestelmän on pystyttävä suojautumaan erilaisia hyökkäyksiä vastaan
6. Laitteiden väärinkäytösriskit tulee huomioida
7. Tunnistustavalla oltava hyvä soveltuvuus mobiililaitekäytössä

Järjestelmän tulee hyödyntää vahvaa tunnistamista. Järjestelmän tunnistustekijät valittiin liittymään käyttäjän omistussuhteeseen ja tietämykseen, sillä Windows Phonessa 8.1:ssä ei vielä ollut vakiovarusteena laitteistoa kehittyneitä biometrisia tunnisteita

varten. Sovelluksen avaaminen vaatii sen, että käyttäjä tietää PIN-koodin. Tunnistuspalvelu taas suostuu palauttamaan tiedot sovellukselle ainoastaan, jos pyyntö on lähetetty siltä laitteelta, jolle henkilökohtaisesti toimitettu salainen avain on ensimmäisenä rekisteröity. Tämä tieto varmistetaan tarkastamalla pyynnön mukana toimitettava käyttäjätunnus, käyttäjän laitteen ID sekä TOTP-salasana. Rekisteröintivaihe on tarkoitus suorittaa vahvaa luottamussuhdetta noudattaen, eli käyttäjä on tunnettu ja hänelle toimitetaan sovelluksen tarvitsema käyttäjäkohtainen avain henkilökohtaisesti. Tarkempi kuvaus rekisteröintiprosessista on kuvattu kohdassa 5.2.1.

Järjestelmän kehitys- ja ylläpitokustannuksien tulee olla kohtuulliset. Käyttäjien hallinnointi ja rekisteröinti toteutettiin asiakkaan hallittavissa olevaksi, jottei kasvava käyttäjämäärä aiheuta kasvavia ylläpitokustannuksia toimittajalle. Koska järjestelmässä ei hyödynnetty muiden palveluntarjoajien tuotteita, ei järjestelmästä muodostu käyttömaksukustannuksia. Pienellä jatkokehitystyöllä ja rekisteröintiprosessin automatisoinnilla ylläpitokustannuksienkin määrä olisi vähennettävissä.

Tunnistamistavan sovelluttava muillekin mobiilialustoille. Koska tiedonhakupalvelu on tehty REST -palveluna, on yleiskäyttöinen ja helposti hyödynnettävä eri palveluissa ja laitteilla. Tunnistamistavan hyödyntäminen on mahdollista myös iPhone- tai Android -laitteilla, sillä molemmissa on kamerat ja QR-koodinlukijasovelluksia saatavilla.

Järjestelmän uusien käyttäjien rekisteröinnin tulee olla helppoa. Rekisteröinti pitää olla myös mahdollista tehdä palvelua käyttävän tahon puolesta, joten sitä ei tulisi hoidattaa jollain kolmannella osapuolella, kuten operaattorilla. Käyttäjäkohtaisten avaimien generointi on helppo integroida käytössä olevaan järjestelmään ja henkilökohtaiset avaimet on helppo lisätä olemassa oleviin järjestelmiin. Hallintakäyttöliittymään oikeudet saanut henkilö voi tehdä uusien käyttäjien rekisteröintejä, joten käyttäjien hallinnasta voidaan huolehtia itse.

Järjestelmän on pystyttävä suojautumaan erilaisia hyökkäyksiä vastaan. Tiedonsiirtoon kohdistuvat hyökkäykset on torjuttu käyttämällä kertakäyttöistä TOTP-salasanaa sekä salattua HTTPS-liikennettä. TOTP-salasana generoidaan käyttäjän henkilökohtaisen avaimen perusteella ja koska salasana vaihtuu puolen minuutin välein, on uudelleenlähetyshyökkäysten toteutus vaikeaa. Salasanoista olisi voitu tehdä myös täysin kertakäyttöisiä, mutta testikäytössä havaittiin myös laillisia käyttötapauksia, joissa käyttäjä ehtii lähettämään useamman kuin yhden pyynnön salasanan voimassaoloaikana. Koska käytettävän palvelun yhteysosoite toimitetaan QR-koodin mukana ja sitä käytetään suoraan sovelluksen muistista, ei hyökkääjä pysty huijaamaan käyttäjää menemään väärään osoitteeseen ja tarkastelemaan tunnistetietoja liikenteestä. Näin ollen välimieshyökkäysten ja sessioiden kaappausten toteutus on vaikeaa. Kalasteluviestit ovat tehottomia järjestelmää vastaan sillä käyttäjä ei itse edes tiedä hänen käyttäjätiliin liittyviä tunnistetietoja tai generoitunutta TOTP-salasanaa, joita vaaditaan tunnistuspyyntöjen mukana. Salasanoihin kohdistuvilta hyökkäyksiltä suojauduttiin rajoittamalla sekä

sovelluksen PIN-koodin sekä tiedonhakupyyntöjen tunnistustietojen arvauskertoja. Tästä syystä esim. salasanoihin kohdistuvat brute force -hyökkäykset eivät ole toimivia. PIN-koodin esittämien estettiin kirjautumisnäkymästä, jotta käyttäjän PIN-koodin katsominen olan yli olisi hankalampaa**.** Windows Phone 8.1 -alustan tietoturvaratkaisuista johtuen käyttäjän laitteelle on vaikea ujuttaa keyloggeria tai suorittaa muuta haitallista ohjelmaa, joka kaappaisi käyttäjän PIN-koodin tämän tietämättä.

Laitteiden väärinkäytönriskit tulee huomioida. Sovelluksen käyttö on suojattu PIN-koodilla, joka myös vaaditaan käyttäjältä aina kun sovellus siirtyy aktiiviseksi taustalta. Näin ollen käyttäjä ei voi unohtaa sovellusta auki taustalle siten, että joku voisi myöhemmin päästä jatkamaan käyttäjän sessiota kirjautumatta sisään. Käytännössä ainut tapa välttää kirjautuminen olisikin kaapata laite käyttäjältä, joka on ehtinyt kirjautua ja parhaillaan käyttää sovellusta. Mobiililaitteelle tallennettu tieto on salattu sovelluskohtaiseen tietovarastoon, joten sitä ei saa otettua laitteelta ulos. Käyttäjien henkilökohtaiset avaimet ovat laitekohtaisia ja avain on käytettävissä vain siltä laitteelta, jolta se ensin rekisteröidään. Tämä huolehditaan käytännössä siten, että tunnistuspyynnön mukana lähetetään myös ainutlaatuinen laitetunnus, joka liitetään käyttäjän avaimeen palvelun päässä. Vaikka joku onnistuisikin saamaan käsiinsä toisen käyttäjän QR-koodin esim. roskakorista, ei hän onnistu kirjautumaan tämän käyttäjän tunnuksilla tietämättä myös tämän laitekoodia.

Tunnistustavalla oltava hyvä soveltuvuus mobiililaitekäytössä. Hyvä soveltuvuus pyrittiin saavuttamaan integroimalla osa tunnistamiseen vaadituista tekijöistä räätälöityyn mobiilisovellukseen. Käyttäjän näkökulmasta hänen tarvitsee käyttää tunnistamisessa vain PIN-koodia. Mobiilisovellusten moniajo ei ole erityisen jouhevaa, joten esimerkiksi erillisen sovelluksen tai tekstiviestillä vastaanotetun salasanan liittäminen mobiilisovellukseen olisi vaikeuttanut kirjautumista. Sovelluksen käyttöön ei vaadita tietokoneella suoritettavaa vaihetta tai mitään ulkoista lisälaitetta. Kaikki muut toiminnot saatiin integroitua sovellukseen paitsi rekisteröintivaiheessa tarvittavaa QR-koodin lukeminen, joka kuitenkin onnistuu puhelimen kamerasovelluksella. QR-koodin sisältö siirretään sovellukseen kopioimalla viestin sisältö leikepöydälle ja liittämällä se rekisteröintinäkymään, mikä kuitenkin tulee suorittaa vain kerran. Tunnistustavaksi valittu TOTP-salasana on generoitavissa vastaavalla tavalla myös muilta alustoilta kuten iOS- ja Android -alustoilla. HTTPS-pyynnöt REST-rajapinnan kautta ovat myös yleiskäyttöisiä, joten ne valittiin tiedonhakupyyntöjen toteutustavaksi.

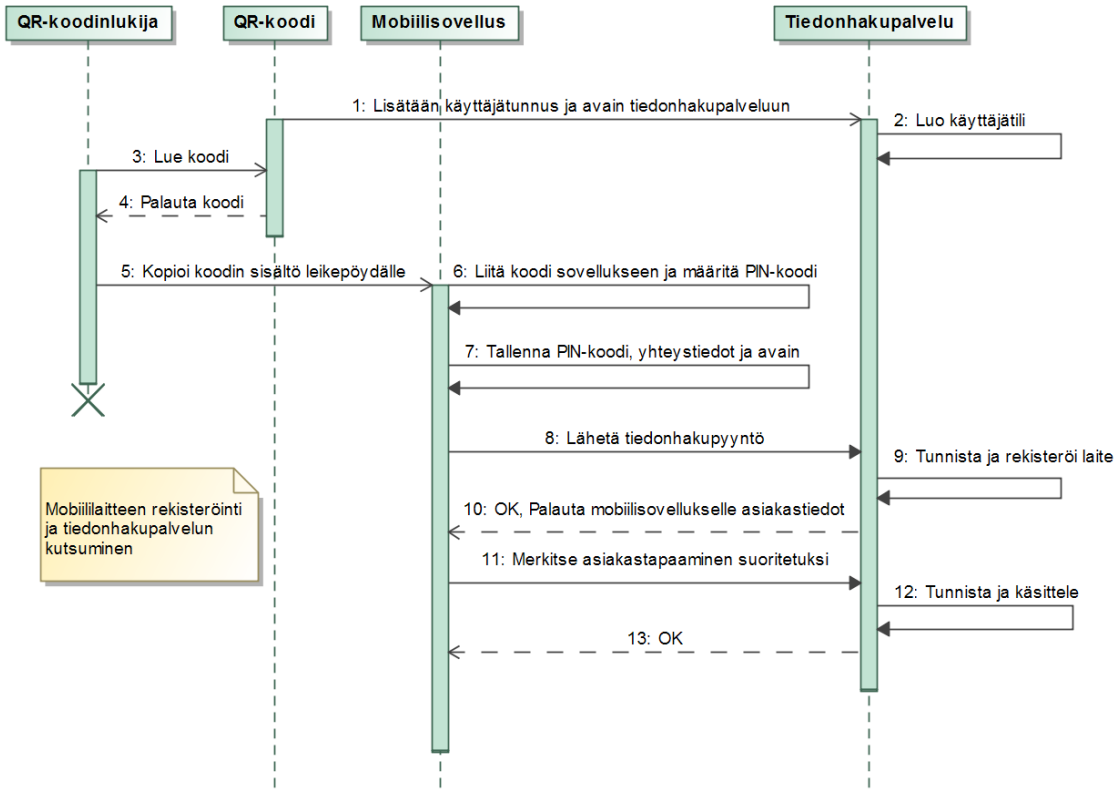
5.2 Mobiilisovellus

Tässä aliluvussa on kuvattu toteutetun mobiilisovelluksen toiminta. Koska mobiilisovellus on vasta prototyyppiasteella, käyttää se vain esimerkkidataa, jonka se hakee rajapinnan kautta tiedonhakupalvelulta.

5.2.1 Perustoiminta

Kehitetty mobiililaitesovellus on Windows Phone 8.1 -alustalle tehty asiakastapaamisiin liittyviä tietoja esittävä prototyyppisovellus. Tulevien asiakastapaamisten tietojen esittämisen lisäksi sovelluksella pystyy myös kuittaamaan asiakastapaamisia suoritetuiksi ja hakemaan reittiohjeita asiakkaan osoitteeseen. Kaikki asiakastiedot tulevat sovellukseen tiedonhakupalvelusta, joka on myös kuvattu tässä luvussa 5.3. Asiakastietoja ei tallennetta laitteelle, vaan ne ovat käytössä sovellusmuistissa sovelluksen elinkaaren ajan.

Mobiililaitesovellus on tällä hetkellä ladattavissa yrityksen henkilöstölle Microsoft Company Hubista. Asennuksen jälkeen ensimmäisen käyttökerran yhteydessä sovellus vaatii käyttäjää rekisteröimään laitteensa, jolloin käyttäjätunnus ja salausavain sidotaan käyttäjän laitteeseen. Rekisteröintiprosessin ja tiedonhakupyyntöjen toiminta on esitetty kuvassa 6.



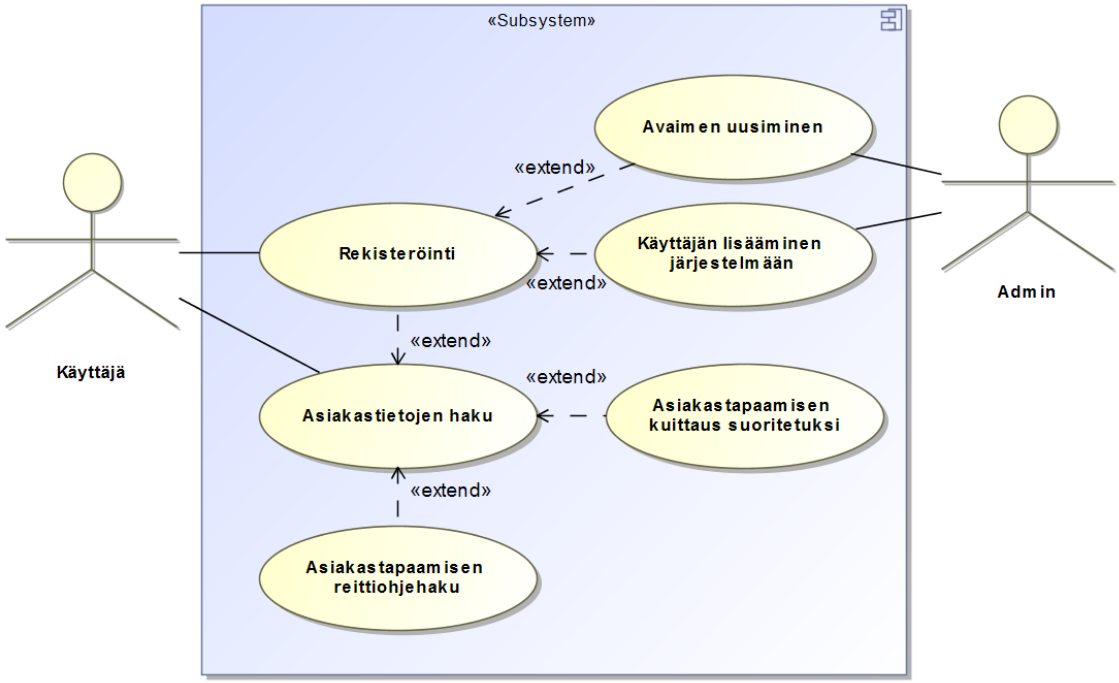
Kuva 6. *Mobiililaitteen rekisteröinti ja tiedonhakupyynnön lähetys*

Rekisteröintiä varten käyttäjän tulee hankkia henkilökohtainen avain, jota vaaditaan tiedonhakupalvelun tunnistamista varten. QR-koodi generoidaan käyttäjälle järjestelmänvalvojan toimesta ja samalla käyttäjälle luodaan käyttäjätili tiedonhakupalveluun (1). Avain toimitetaan käyttäjälle QR-koodin muodossa esim. tulostettuna paperille. Käyttäjä lukee QR-koodin sisällön puhelimen kamerasovelluksella (3-4) ja tallentaa koodin sisällön leikepöydälle (5). Käyttäjä avaa mobiilisovelluksen, joka avautuu rekis-

teröintinäkymään. Käyttäjä liittää leikepöydälle kopioimansa koodin mobiilisovellukseen ja määrittelee PIN-koodin (6), jota vaaditaan aina sovellusta käynnistettäessä. Käyttäjä tallentaa tiedot, jolloin mobiilisovellus tallentaa PIN-koodin sekä QR-koodista saamansa avaimen sekä muut yhteystiedot sovelluksen tietovarastoon (7). Sovellus siirtyy asiakastietonäkymään, joka lähettää tiedonhakupyynnön tiedonhakupalveluun (8). Tiedonhakupalvelu tunnistaa käyttäjän generoimalla kertakäyttösalasanan tiedossaan olevalla käyttäjän avaimella ja vertaamalla sitä pyynnön mukana tulleeseen salasanaan. Salasanat täsmäävät, mutta tiedonhakupalvelu huomaa, ettei käyttäjän laitetta vielä sidottu käyttäjän avaimeen. Tiedonhakupalvelu poimii tiedonhakupyynnöstä käyttäjän laitetunnisteen ja tallentaa sen käyttäjätilin tietoihin (9). Tunnistamisen ja rekisteröinnin jälkeen tiedonhakupalvelu lähettää käyttäjälle pyydetyt tiedot asiakastapaamisista (10). Hetken kuluttua käyttäjä valitsee sovelluksesta yhden asiakastapaamisen ja kuittaa sen suoritetuksi. Tästä lähtee muutospyyntö tiedonhakupalveluun (11). Tiedonhakupalvelu tunnistaa käyttäjän kuten aiemminkin ja varmistaa myös, että pyyntö tulee siltä laitteelta, joka käyttäjätilille on rekisteröity (12). Lopuksi tiedonhakupalvelu kuittaa käyttäjälle, että muutospyyntö käsiteltiin onnistuneesti (13). Käyttäjä sulkee lopuksi sovelluksen.

5.2.2 Toiminnot

Tiedonhakupalvelun erilaiset käyttötapaukset ja niiden väliset riippuvuudet on esitetty kuvassa 7.



Kuva 7. *Järjestelmän toiminnot*

Järjestelmänvalvojalla on tiedonhakupalvelun ylläpitoon liittyen kaksi tehtävää: Käyttäjien lisääminen järjestelmään ja käyttäjäkohtaisten salausavaimien uusimien. Käyttäjän

rekisteröinnin edellytyksenä on, että hänet on ensin lisätty järjestelmään. Järjestelmänvalvoja voi myös uusia käyttäjän salausavaimen tarvittaessa, jonka jälkeen käyttäjän pitää suorittaa rekisteröinti uudelleen. Jotta käyttäjä pystyy hakemaan asiakastietoja tai tekemään muita toimintoja, tulee hänen ensin olla rekisteröitynyt. Rekisteröinnin jälkeen käyttäjä voi hakea asiakastiedot mobiililaitteelleen ja selata tietoja. Käyttäjä voi avata haluamansa asiakastapaamisen tiedot omaan näkymäänsä, jossa näkyy asiakkaan nimen, osoitteen, tapaamisen ajankohdan ja statuksen lisäksi pieni karttanäkymä. Käyttäjä voi tässä näkymässä käsitellä asiakaskohtaisia tietoja, kuten suorittaa reittihaun asiakastapaamiseen tai kuitata tapaamisen suoritetuksi.

5.3 Tiedonhakupalvelu

Tässä aliluvussa on kuvattu tiedonhakupalvelu, joka toimittaa asiakastapaamisiin liittyvää tietoa mobiilisovellukselle. Koska tiedonhakupalvelun yhteyteen toteutettiin myös käyttäjien tunnistaminen, viitataan siihen myös tunnistamispalvelu -nimellä. Toteutetun järjestelmän tiedonhakupalvelu koostuu asiakastietoja palauttavasta REST-rajapinnasta sekä käyttäjä- ja asiakastapaamistiedoista koostuvasta testidatasta.

REST on hypermediajärjestelmien arkkitehtuurimalli, joka koostuu joukoista rajoitteista. REST:in rajoitteita noudattamalla lopputuloksena on arkkitehtuuri, joka toimii hyvin mm. skaalautuvuuden, muunneltavuuden, käytettävyyden ja saatavuuden osalta. [45, s.2] REST:iä käyttävät web servicet eivät yleensä sisällä matalan tason tietoturvatoimintoja, vaan HTTP-pyyntöjen suodatukseen käytetään palomuureja tai muita ratkaisuja. [73][86] Tiedonhakupalvelussamme tunnistaminen suoritettiin pyyntöjen mukana tulleiden tunnistetietojen avulla, ennen kuin tiedonhakupyyntöjä käsiteltiin.

Tiedonhakupalvelu toimii REST-rajapinnan kautta, joka asetettiin toimimaan vain salatun HTTPS-yhteyden yli. Tiedonhakupyyntöjen tulokset palautetaan rajapinnan kautta JSON-muodossa (JavaScript Object Notation). Tiedonhakupalvelua ei integroitu olemassa olevaan asiakas- ja käyttäjätietokantaan vaan asiakastapaamisen tiedot piti sisällään kuvan 8 mukaiset XML-muotoiset tiedot.

```
<?xml version="1.0" encoding="utf-8"?>
<addresses>
  <address id="100">
    <customer>Perttilä, Jorma</customer>
    <meeting>15.05.2015 14:00</meeting>
    <streetaddress>Hämeenkatu 3</streetaddress>
    <city>Tampere</city>
    <visited>true</visited>
  </address>
…
</addresses>
```

Kuva 8. *REST-tiedonhakupalvelun tietosisältö*

REST-rajapinta tarjoaa kolme palvelua: kaikkien asiakastapaamisten tietojen hakemisen, yhden tapaamisen tietojen hakemisen ja yhden tapaamisen tietojen muokkaamisen.

Hakutoiminnot tehdään GET- ja muokkaus POST-pyynnöllä. Koska tämän sovelluksen kannalta ei ollut oleellista pystyä muuttamaan muita tapaamisten tietoja, niin POST-pyynnöillä voidaan ainoastaan merkitä tapaamisia suoritetuksi,

5.3.1 Käyttäjienhallinnan toteutus

REST-mallin mukaisesti käyttäjien sessioita ei tallenneta, vaan jokaisen pyynnön yhteydessä suoritetaan käyttäjän tunnistaminen. Tunnistusta varten tiedonhakupalvelulla on tiedossa oikeutettujen käyttäjien osalta käyttäjätunnus, salainen avain, laitetunnus sekä aikaleimasta päätelty aikaero palvelimeen verrattuna. Käyttäjätunnukset ja salaiset avaimet on luotu QR-koodigeneraattorin avulla ja laitetunnukset lisätään käyttäjille mobiililaitteen rekisteröitymisvaiheessa ensimmäisen tunnistuspyynnön käsittelyn yhteydessä. Kuva 9 esittää järjestelmän käyttäjiin liittyvät tiedot, joita käytetään lähinnä tunnistamisessa.

```
<?xml version="1.0" encoding="utf-8"?>
<users>
  <user id="10">
    <login>user10</login >
    <key>K6JDF09K34MKLDGO9LODN3…212HNSDFKSDJM354MSWFNK2M24</key >
    <deviceid>OP-QR-ST-77-88-99</deviceid >
    <timedifference>15</timedifference >
  </user>
…
</users>
```

Kuva 9. *Tunnistuspalvelun käyttäjätiedot*

Tunnistamisvaiheessa pyynnöistä tarkistetaan seuraavat ehdot:

- Käyttäjätunnus löytyy järjestelmästä
- Käyttäjän salausavaimella generoitu TOTP vastaa käyttäjän toimittamaa salasanaa. Tarkistuksessa huomioidaan käyttäjäkohtainen aikaero.
- Laitetunnus on rekisteröity kyseiselle käyttäjälle. Tai jos käyttäjä ei ole rekisteröinyt vielä laitettaan, niin lisätään käyttäjän tietoihin laitetunnus.

5.4 QR-koodien generaattori

QR-koodien generointi on tehty erikseen toteutetulla yksinkertaisella Windows Forms-sovelluksella, joka luo testikäyttäjien tunnukset tiedonhakupalvelun vaatimaan muotoon. Generaattorille voi määrittää testikäyttäjän nimen ja sen palvelun osoitteen, johon tunnus tulee käytettäväksi. Generaattori luo näiden lisäksi satunnaisen avaimen, liittää nämä kolme tietoa QR-koodiin ja esittää koodin näytöllä. Koska QR-koodien generointi tehdään erillään tiedonhakupalvelusta, tulee käyttäjätunnus ja salausavain lisätä myös tiedonhakupalveluun käyttäjän tietoihin ks. kuva 9. QR-koodi voidaan lukea mobiilisovellukseen suoraan ruudulta tai tallentaa kuvaksi käyttäjälle toimittamista ja myöhempää lukemista varten.

5.5 Hylätyt ratkaisuvaihtoehdot

Jotta laitteen rekisteröinti olisi käyttäjäystävällisempää, voitaisiin myöhemmin mobiilisovellukseen integroida kameratoiminto, jolla QR-koodin tiedot saataisiin tallennettua kätevämmin mobiililaitteelle. Tätä toimintoa ei koettu tarpeelliseksi toteuttaa vielä toistaiseksi, sillä ensikirjautuminen tarvitsee suorittaa vain kerran ja se on loppujen lopuksi nopea tehdä leikepöydän kauttakin.

Tiedonhakupalvelun käyttämät asiakastapaamis- ja käyttäjätiedot on tallennettu prototyyppiversiossa vain XML-tiedostoon. Näitä varten ei luotu omaa tietokantaa, mutta vastaavat tiedot olisi jatkossa helppo hakea tai lisätä olemassa olevien järjestelmien tietokantoihin.

Järjestelmän hyödyntämistä TOTP-salasanoista ei päätetty tehdä aidosti kertakäyttöisiä sellaisen käyttötapauksen takia, jossa käyttäjä ehtii hakemaan asiakastiedot ja merkitsemään tapaamisen suoritetuksi yhden kertakäyttösalasanan voimassaoloaikana.

QR-koodin generointi toteutettiin omana sovelluksenaan, mutta se olisi myös integroitavissa tunnistamispalveluun. Jos tämän kaltainen järjestelmä otettaisiin käyttöön, voitaisiin koodien generointi ja käyttäjien hallinta integroida olemassa olevaan järjestelmään. QR-koodien tietosisältö voitaisiin myös salata esim. palvelukohtaista symmetristä salausavainta käyttäen ja purkamalla QR-koodin sisällön salaus mobiilisovelluksessa samalla avaimella.

6. TUNNISTUSTAPOJEN VERTAILU VAATIMUSTEN POHJALTA

Toteutettavalle järjestelmällä asetetut vaatimukset karsivat pois suurimman osan tunnistustekniikoista tekniikoita valittaessa omaan käyttötarkoitukseemme. Tulosten kokonaiskuvan helpottamiseksi tulokset on koottu taulukkoon 3, johon kunkin vaatimuksen täyttymisehdot ovat kerrottu tunnistustapakohtaisesti periaatteella: kriteeri täyttyy (+) tai kriteeri ei täyty (-). Jos muutoin ei ole mainittu, niin tunnistustavan soveltuvuus on arvioitu Windows Phone 8.1 -alustaa vasten. Valinnat on perusteltu tulosten esittelyjen jälkeen.

1. Järjestelmän tulee hyödyntää vahvaa tunnistamista
2. Järjestelmän kehitys- ja ylläpitokustannuksien tulee olla kohtuulliset
3. Tunnistamistavan sovelluttava muillekin mobiilialustoille
4. Järjestelmän uusien käyttäjien rekisteröinnin tulee olla helppoa.
5. Järjestelmän on pystyttävä suojautumaan erilaisia hyökkäyksiä vastaan
6. Laitteiden väärinkäytösriskit tulee huomioida
7. Tunnistustavalla oltava hyvä soveltuvuus mobiililaitekäytössä

Taulukko 3. *Tunnistustekniikoiden vertailu Windows Phone 8.1 -alustalla*

	1	2	3	4	5	6	7	YHT
Salasana ja OOB+OTP	+	+	+	+	+	-	-	**5**
Salasana ja VPN	+	+	-	+	+	+	-	**5**
Salasana ja Mobiilivarmenne	+	+	+	-	+	+	-	**5**
Salasana ja Push-viestit	+	-	+	+	+	+	+	**6**
Salasana ja Fyysinen token	+	-	+	-	+	+	+	**5**
Salasana ja Sovelluspohjainen token	+	+	+	+	+	+	+	**7**
Salasana ja Biometrinen	+	+	-	-	+	+	-	**4**
Salasana ja Sertifikaatti	+	-	-	+	+	+	+	**5**
Microsoft Passport (Windows 10 Mobile)	+	+	-	+	+	+	+	**6**

Toisella tiedonsiirtokanavalla lähetetyt kertakäyttösalasanat lähetetään useimmiten SMS-viesteillä tai sähköposteilla, joten keskitytään arvioinnissa näihin medioihin. Toisen tiedonsiirtokanavan ja kertakäyttösalasanan käyttö ei tee tunnistustavasta vahvaa kaikkien tulkintojen mukaan, kun toisena tekijänä on salasana. Jos otetaan sähköposti toiseksi tiedonsiirtokanavaksi, voidaan tulkita sähköpostitilin olevan jotain, mitä käyttäjällä on. Sähköpostitiliin kirjautuminen tosin vaatii vain käyttäjätunnuksen ja salasanan, jolloin voitaisiin myös tulkita tunnistamisen koostuvan kahdesta tietämykseen perustuvasta tekijästä. Tekstiviestien vastaanottaminen vaatii, että käyttäjällä on hallussaan SIM-kortti, joten tunnistaminen on luokiteltavissa vahvaksi. Monissa palveluissa on valmiiksi tuki ainakin sähköpostiviesteille, joten sähköpostin käyttäminen toisena tiedonsiirtokanavana on edullista. Jos taas käytetään SMS-viestejä kertakäyttösalasanojen lähetykseen, voivat kustannukset nousta kirjautumismäärien kasvaessa suuriksi. Sähköpostit ja SMS-viestit on vastaanotettavissa eri käyttöjärjestelmällä varustettujen modernien puhelimien toimesta, joten tekniikka soveltuu muillekin alustoille. Käyttäjien rekisteröinti voidaan tehdä organisaation toimesta ja monivaiheisen tunnistaminen käyttöönotto vaatii vain käyttäjän sähköpostiosoitteen tai puhelinnumeron kysymistä, jos sellaista ei rekisteröintivaiheessa tehdä. Kertakäyttösalanoilla voidaan suojautua salasanoihin kohdistuvia hyökkäyksiä, uudelleenlähetyshyökkäyksiä ja kalastelua vastaan, sillä kertakäyttösalasanan syöttäminen tunnistamisvaiheessa on mahdollista vasta kuin käyttäjätunnus ja salasana on syötetty oikein ja kertakäyttösalasana on eri kirjautumiskerroilla. SMS-ja sähköpostiviestien huonona puolena oli se, että niitä pystyy lukemaan lukitsemattomalta mobiililaitteelta, jos sähköpostitili on määritelty laitteelle. Lisäksi kertakäyttösalasanojen lukeminen ja kopioiminen tekstiviesti- tai sähköpostisovelluksesta hidastaa tunnistamisprosessia ja voi turhauttaa käyttäjiä sovelluksien vaihtelusta johtuen.

Sisäänrakennettu VPN-tuki Windows Phone 8.1 -laitteilla oli vielä huonosti tuettua, eikä kaikilta valmistajilta ollut tarjota VPN-mobiilisovelluksia. Esim. yrityksemme käytössä oleva Cisco AnyConnect VPN-sovellus oli työn teknisen toteutusvaiheen aikana vasta kehitysvaiheessa, joten sitä emme voineet hyödyntää. Nykyään se on jo saatavilla Windows Phone 8.1 -laitteille. Koska yrityksellämme on jo käytössä VPN, ei siitä olisi aiheutunut hankintakustannuksia. VPN-yhteys on erittäin turvallinen tapa muodostaa salattu yhteys käyttäjän ja palvelun välille, koska viestit voidaan salata ja niiden eheys voidaan tarkistaa. Väärinkäyttötilanteessa VPN:n huono puoli on, että se antaa pääsyn myös muihin sisäverkon jaettuihin resursseihin. Cisco AnyConnect on saatavilla myös Android ja iOS-käyttäjille, mutta sovellusten saamien arvostelujen perusteella monet käyttäjät eivät ole olleet tyytyväisiä salasanan kyselyyn uudelleenyhdistäminen yhteydessä. Mobiilikäytettävyydeltään sovelluksen käyttäminen VPN-ratkaisuna ei olisi kovin käytettävä mobiililaitteella, ellei joustavampaa kirjautustapa tule tarjolle jatkossa.

Mobiilivarmenne on tunnistustapana vahva, sillä erikoisvalmisteinen SIM-kortti voidaan luokitella joksikin, mitä omistat. Mobiilivarmennesopimukset ovat myös hyvin edullisia parin euron kuukausihinnallaan ja ilmaisilla aloituskausillaan. Koska mobiili-

varmenne on SIM-kortin sisällä, soveltuu se eri puhelinmalleille ja mobiilipuhelinalustoille. Mobiilivarmenteen käyttöönotto on hidasta, sillä rekisteröinti vaatii henkilökohtaisen käynnin operaattorilla. Lisäksi mobiilivarmennetta ei ole ollut saatavilla kaikille uusimmille dataliittymille, minkä johdosta osa käyttäjistä joutuisi hankkimaan uuden liittymän mobiilivarmennetta varten. Turvallisuutensa puolesta mobiilivarmenteella voidaan suojautua yleisimmiltä hyökkäyksiltä. Kalasteluja vastaan on otettavissa käyttöön häirinnänestopalvelu, jolla estetään aiheettomien tunnistuspyyntöjen kuittaamisyritykset toisella. Mobiilivarmenteen käyttö edellyttää tunnistuspyyntöjen kuittaamista PIN-koodilla, joten pelkkä laitteen joutuminen vääriin käsiin ei mahdollista väärinkäyttöä. Mobiilivarmenteelle tulevat kuittauspyynnöt ovat muusta sovelluksesta erillään, mikä voi haitata sen käyttöä mobiililaitesovellusta käytettäessä. Mobiilivarmenteen käyttö olisi sujuvampaa selainpohjaisen palvelun tunnistamisessa.

Push-viestit tunnistamisessa lukeutuvat myös vahvaksi tunnistamistavaksi, sillä ne vaativat puhelimen omistussuhteen. Jos push-viestien vaatimia järjestelmiä ei ole jo käytössä, kasvavat käyttöönottokustannukset merkittävästi. Käyttökustannuksiltaan Microsoftin push-viestien käyttö on kohtalaisen edullista ja pienillä tunnistusmäärillä jopa ilmaista. Kehityskustannukset koiemme kuitenkin liian suuriksi. Microsoft push-viestit ovat lähetettävissä iOS- ja Andoid-alustojen laitteille Azure Notification Hubin kautta, joten, tunnistustapa soveltuisi samalla näille mobiilialustoille. Uuden käyttäjän rekisteröinti push-palveluun on vaivatonta. Push-viesteillä voidaan torjua yleisimmät hyökkäykset, sillä push-viestien väärennys toiselle laitteelle ei ole mahdollista. Väärinkäytösriskien estämiseen vaikuttaa suuresti sovelluksen toimintatapa. Sovelluksen pitäisi pyytää jokin tunnistetieto, jottei sen väärinkäyttö olisi mahdollista laitteen joutuessa vääriin käsiin. Mobiililaitekäytettävyyden arviointi riippuu hyvin pitkälti toteutustavasta.

Fyysiset tokenit luokitellaan tunnistetekijöistä joksikin, mitä omistat. Paritettuna sovelluksessa salasanan kanssa, saadaan tunnistustavasta vahva. Fyysiset tokenit ovat yleensä kalliita, ja katoamisten sekä lyhyen käyttöikänsä takia niiden uusiminen tulee kalliiksi. Koska fyysiset tokenit ovat erillisiä laitteita, eikä niissä ole suoranaista yhteyttä taustasovellukseen, soveltuvat ne eri sovellusalustoille. Uusien käyttäjien rekisteröinti on aikaa vievää, sillä fyysiset tokenit joudutaan toimittamaan käyttäjille erikseen. Fyysisillä tokeneilla voidaan suojautua brute-force- ja uudelleenlähetyshyökkäyksiä vastaan, sillä tokenin arvo vaihtuu lyhyin aikavälein. Vaikka fyysiset tokenit ovat hyvin pienikokoisia ja ne kulkevat esim. avaimenperässä mukana, lisää niiden jatkuva kuljettaminen katoamisriskiä. Vaikka fyysinen token joutuisi vääriin käsiin, tarvitsee kirjautumisessa myös käyttää käyttäjäkohtaista PIN-koodia, millä voidaan torjua väärinkäyttöyritykset laitteen kadotessa. Koska salainen avain on koodattu tokenille valmistusvaiheessa ja tokenin tietoihin ei pääse ulkoa päin käsiksi, ei salaisen avaimen selvittäminen laitteelta ole myöskään mahdollista. Fyysisien tokenien käyttö ei vaikuta mobiililaitekäytettävyyteen, sillä salasanan generointi tapahtuu erillään haittaamatta moniajoa. Fyysisiä tokene-

ja voi joutua ajoittain synkronoimaan, jos palvelimen kellonaika ajautuu liian kauas tokenin sisäänrakennetusta kellosta. Synkronointi viivästyttää sitä tunnistamiskertaa, jolloin synkronointi tarvitsee tehdä. Vaihtoehtoisesti tunnistamisvaatimuksia voidaan löysentää pidentämällä salasanojen voimassaolon aikaväliä.

Koska soft tokenit ovat helpommin manipuloitavissa kuin hard tokenit, pitää niiden jakelussa ja käytössä noudattaa erityistä varovaisuutta. Vahvan tunnistamisen määrityksen ne täyttävät mielestäni ainakin silloin, kun käyttäjän salaisen avaimen toimittaminen ja käsittely toteutetaan turvallisesti. Kun sovelluksen elinkaaren aikana varmistutaan, että avain on palvelimen lisäksi tiedossa vain yhden käyttäjän laitteella, voidaan soft token tulkita joksikin, mitä käyttäjällä on. Soft tokeneissa usein käytetyt HOTP- ja TOTP-salasanat on generoitavissa eri sovellusalustoilla standardinmukaisilla algoritmeilla, joten tunnistustapa soveltuu muillekin alustoille. Salasanojen generointi on lisäksi vaivatonta ja tehtävissä palveluntarjoaman toimesta, joten kustannukset jäävät pieniksi. Uusien käyttäjien rekisteröintiin vaikuttaa sovelluksen toteutustapa ja kehitykseen käytetty aika. HOTP-ja TOTP salasanoilla voidaan torjua salasanoihin kohdistuvat hyökkäykset ja uudelleenlähetyshyökkäykset. TOTP on näistä hieman parempi vaihtoehto, sillä salasana vaihtuu ajan myötä, jolloin sen arvaaminen on vaikeampaa. Joidenkin tunnuslukusovellusten, kuten Microsoft Authenticatorin heikkoutena on se, että sovellus ei pyydä käyttäjää tunnistamaan itseään ja lisäksi se näyttää käyttäjätilin, johon generaattori on liitetty. Käytettävyyttä haittaavia moniajo-ongelmia voi esiintyä pääosin kolmansien osapuolien toteuttamissa kertakäyttösalasanan generointiin tarkoitetuissa sovelluksissa varsinkin, jos tunniste tulisi kopioida samalla laitteella samanaikaisesti ajettavaan toiseen sovellukseen. Soft tokenien synkronisuusongelma voi esiintyä sekä TOTP:lla ja HOTP:lla, mutta ongelmaa voidaan helpottaa sallimalla myös esim. edellinen ja seuraava kertakäyttösalasana.

Biometrinen tunnistus ei menestynyt vertailussamme hyvin, sillä vanhemmassa Windows Phone 8.1 -alustassa se oli huonosti tuettuna. Koska biometriikat edustavat sitä mitä käyttäjä on, olisivat ne salasanan kanssa paritettuna muodostaneet vahvan tunnistamisen. Kuvaan, ääneen tai käyttäytymiseen perustuvien biometriikoiden käyttö olisi ollut teoriassa mahdollista, mutta se olisi vaatinut omien tunnistusalgoritmien kehittämistä. Kehityskustannukset olisivat siis nousseet todella suuriksi. Fyysisiä biometriikoita käytettäessä rekisteröinti olisi vaatinut käyttäjän kuvan tai ääninäytteen ottamista. Palvelimen ja eri mobiililaitteiden sensoreista johtuvien erojen takia kuvaan tai ääneen perustuva tunnistus olisi varmaan pitänyt toteuttaa mobiilisovelluksen puolella. Käyttäytymiseen perustuva tunnistaminen olisi vaatinut opettelujakson, jonka aikana mobiilisovellus tai palvelinpuoli muodostaa profiilin käyttäjän toimien perusteella. Käyttäytymiseen perustuva tunnistaminen olisi mobiilisovelluksessa voitu toteuttaa tarkkailemalla PIN-koodin tai sovelluksen muiden toimintojen välisiä näppäinpainalluksia. Koska kehitetyssä sovelluksessa on melko vähän toimintoja, tunnistaminen ei välttämättä olisi ollut tarpeeksi luotettavaa. Palvelinpuolella käyttäjäprofiili olisi voitu muodostaa

esim. pyyntöjen ajankohtien tai niiden välisten aikojen perusteella. Käyttäytymiseen liittyvässä tunnistuksessa olisi ollut hyödyllistä hyödyntää jotain toista tunnistustapaa profiilin muodostamisen ajan, sillä muutoin käytön alkuvaiheessa tunnistus olisi ollut heikkoa puutteellisen profiilin takia. Fyysisiä biometrisia tunnisteita käytettäessä haasteeksi olisi muodostunut kuvan tai salaa nauhoitetun puheen käyttämisen estäminen tunnistusprosessissa. Koska toisen kuva on mahdollista varastaa julkisen kuvagallerian kautta tai ottamalla toisesta kuva suoraan, olisi hyökkäyksien torjuminen ollut haastavaa. Kameratoiminnon integroiminen sovellukseen tai käyttäytymisen tarkkailu taustalla olisi tehnyt mobiilikäytöstä sujuvaa. Kuvaan perustuvassa tunnistamisessa ongelmaksi olisi voinut muodostua hieman pidempi tunnistamisaika, sillä ei ole varmaa tietoa kuinka nopeaksi kuvien vertailu olisi ollut toteutettavissa mobiililaitteen rajallisella laskentateholla. Mobiilikäytettävyyteen olisivat myös vaikuttaneet vaihtelevat olosuhteet taustamelun tai valaistuksen osalta, jolloin sovelluksen käyttö ei olisi ollut mahdollista optimaalisista poikkeavissa olosuhteissa.

Sertifikaatit voidaan luokitella tunnistamisessa joksikin, mitä käyttäjällä on. Käytettäessä niitä esim. PIN-koodin kanssa saadaan tunnistamisesta vahva. Sertifikaattien luomiseen ja jakamiseen vaaditun PKI-järjestelmän pystyttäminen ja ylläpito on kallista. Lisäksi asennuksiin liittyvät tukipyynnöt kasvattavat kustannuksia joidenkin käyttäjien osalta. Sertifikaatit toimivat eri sovellusalustoilla. Uusien käyttäjien lisääminen PKI-järjestelmää on hieman työlästä ja se vaatii käyttäjiltä myös normaalia enemmän tietotaitoa ja panostusta. Sertifikaatit ovat toimintaperiaatteensa ansiosta erittäin turvallisia. Käyttäjät allekirjoittavat tunnistuspyyntöjen mukana tulevat muuttuvat haasteet sertifikaateistaan löytyvillä yksityisillä avaimilla sen sijaan, että avain lähetettäisiin pyyntöjen mukana. Palvelin voi tarkistaa allekirjoituksen aitouden käyttämällä käyttäjän julkista avainta. Sertifikaatit tulee olla laitteella suojattuna PIN-koodilla tai salasanalla, jottei hyökkääjä voi suoraan käyttää haltuunsa saamaa laitetta ja sille asennettuja sertifikaatteja. Sertifikaattien yksityinen avain on turvassa sertifikaatin tiedossa, joten sitä ei pysty varastamaan laitteelta. Sertifikaattien asennus mobiililaitteilla on peruskäyttäjille vaikeaa, mikä haittaa mobiilikäytettävyyttä.

Microsoft Passport osoittautui ominaisuuksiltaan todella lupaavaksi tunnistustavaksi, jos kehitettävä järjestelmä tehdään puhtaasti Windows 10 Mobile alustalle. Microsoft Passport on vahva tunnistustapa, sillä se käyttää PIN-koodin tai biometrisen tunnisteen lisäksi toisena tunnistustapana laitekohtaista tarkoin varjeltua yksityistä avainta. Koska avaintenhallinta on toteutettu valmiiksi alustan rajapinnoissa, ei Microsoft Passportin käyttöönotto vaadi PKI-järjestelmän pystytystä. Tämä tekee sen käytöstä edullisen. Microsoft Passportin käyttö on valitettavasti rajoitettu ainoastaan Windows 10- ja Windows 10 Mobile- alustoille, joten tunnistustapa ei ole hyödynnettävissä esim. iOS- ja Android-laitteilla. Jos Microsoft Passportia halutaan käyttää vanhoille Windows 8.1 Phoneille asennetuissa Windows 10 Mobile käyttöjärjestelmissä, ei Windows Hellon biometriikat ole käytettävissä, vaan käyttäjä voi hyödyntää vain PIN-koodia. Uusien

käyttäjien rekisteröinti on helppoa, ja käyttäjiä lisättäessä riittää, että mobiilisovelluksen generoiman julkisen avaimen ja muut käyttäjätiedot lähetetään ja tallennetaan sovelluspalvelimelle. Microsoft Passportia käytettäessä palvelin lähettää sovellukselle allekirjoitettavaksi erilaisen haasteen joka kerralla, minkä takia uudelleenlähetyshyökkäykset ovat tehottomia sitä vastaan. Kuvaan perustuvaa tunnistamista hyödynnettäessä käyttäjän pitää käännellä päätään, jolloin pelkän kuvan esittäminen kameralle ei ole tehokas tapa huijata kameraa. Asiantuntijat ovat tosin arvioineet, että animaation näyttäminen kameralle voisi sen sijaan olla mahdollista. Silmän iirikseen perustuva tunnistus on luotettavampi vai laitteen haltuunsa saaneelle hyökkääjälle vaikeammin huijattavissa. Microsoft Passport sopii Windows 10 Mobile alustan mobiililaitteille, sillä tunnistustekniikka on helposti integroitavissa omiin sovelluksiin ja siihen on saatavissa Microsoftin laatimat kattavat ohjeet [79][82]. Pienenä heikkoutena voidaan pitää tunnistamisen hitautta, joka useamman sekunnin kestollaan on hitaampaa kuin PIN-koodin syöttäminen. Kiireisten tehokäyttäjien keskuudessa voikin PIN-koodi jäädä nopeutensa ansiosta ensisijaiseksi tunnistustekniikaksi.

Sovelluspohjaiset tokenit menestyivät vertailussa hyvin täyttäen kaikki asettamamme vaatimukset, joten valitsimme ne käytettäväksi tunnistustavaksi. Sovelluspohjaisista tokeneista TOTP koettiin HOTP:ia paremmaksi vaihtoehdoksi, se pysyy synkronoituna paremmin ja sen arvaaminen on vaikeampaa hyökkääjän toimesta. TOTP-salasanojen generoinnin integroiminen mobiilisovellukselle ja palvelinpuolelle oli yksikertaista ja siihen löytyi hyvät ja luotettavat referenssitoteutukset. Sovelluspohjaisten tokenien ongelmana on se, että ne ovat kopiotavissa salausavaimien joutuessa vääriin käsiin. Tämän takia tiukensimme rekisteröintivaihetta siten, että salausavaimen lähettämisessä käyttäjälle hyödynnettiin ulkoisena kanavana QR-koodia, mikä vähensi avaimen altistumista ulkopuolisille uhille. Salausavain ja muut tunnistetiedot tallennettiin QR-koodille, joka voitiin nopeasti kuvata käyttäjän puhelimeen suoraan tietokoneen ruudulta tai tulostetulta lapulta. Koska myös tiedonhakupalvelun yhteysosoite lähetettiin QR-koodin mukana, pystyimme estämään sovellusta ottamasta yhteyttä muuhun palveluun. Tällä tavalla pystymme myös asettamaan eri mobiilisovellukset käyttämään eri palveluja tarvittaessa. QR-koodin hyödyntäminen mahdollisti myös vahvempien salausavaimien käytön, ilman että käyttäjän tarvitsi niitä itse lisätä laitteelleen kirjoittamalla. Rekisteröintivaiheessa käyttäjän salausavain lukittiin sille laitteelle, josta rekisteröintivaiheen ensimmäinen tunnistuspyyntö on tullut. Vaikka joku saisi käsiinsä toisen henkilön salausavaimen, ei se hänen laitteeltaan toimisi. Sovellukselle siirtämisen jälkeen salausavain tallennettiin sovelluksen salattuun tietovarastoon ja sovelluksen käynnistäminen suojattiin PIN-koodilla. Täten laitteensa haltuun saanut henkilö ei voisi käyttää sovellusta luvatta tai urkkia salausavainta. TOTP salasanojen generoinnin päätimme integroida osaksi mobiilisovellusta. Tällä tavoin saimme sujuvoitettua sovelluksen toimintaa ja pystyimme käyttämään pidempää salasanan pituutta käyttäjää rasittamatta. Rajapinnan puolella huomioimme käyttäjän ja tiedonhakupalvelun välisen aikaeron, jota hyödynnettiin käyttäjän generoiman salasanan vertailussa. Tällä pystyimme vähentämään laitteiden kel-

lonaikojen siirtämisen tarvetta. Koska sovelluksemme tarvitsi vahvan tunnistustavan ja sovelluksessa on hyvin rajallinen määrä toimintoja, koimme riskiin perustuvan tunnistuksen olevan turhaa.

7. YHTEENVETO

Edellä tehtyjen havaintojen pohjalta kokoan tutkimustulokset tähän lukuun yhteenvedon muodossa. Yhteenvedossa kertaan tutkimuksen tavoitteet ja toteutustavan, kuvaan toteutetun järjestelmän ja arvioin teknisen tutkimuksen onnistumista. Esittelen myös pääpiirteittäin tutkimuksessa tehdyt havainnot ja annan suositukset liittyen tutkittuihin tekniikoihin.

Tämän työn tarkoituksena oli kehittää Windows Phone 8.1:lle vahvaa tunnistamista hyödyntävä mobiilisovellus, jolla on tarkoitus käyttää asiakastapaamisiin liittyviä tietoja turvallisesti. Tunnistamistavan ja tietoturvaratkaisujen valinnassa otimme huomioon kustannukset, käytettävyyden, turvallisuuden sekä muita järjestelmälle asetettuja vaatimuksia. Toteutimme tutkimuksen kahdessa eri vaiheessa: Kirjallisuusselvityksenä ja empiirisenä tutkimuksensa. Kirjallisuusselvitysosiossa perehdyimme vahvoihin tunnistamistapoihin, muihin yleisiin tietoturvatekniikkoihin ja Windows Phone -alustan tietoturvaratkaisuihin. Lisäksi tutustuimme tutkituilla tekniikoilla toteutettuihin järjestelmiin ja arvioimme, miten tekniikat soveltuvat Windows Phone -alustalle. Kirjallisuusselvitys valittiin tutkimusmenetelmäksi, sillä sen avulla saatiin kattava kuva olemassa olevien tunnistamistapojen toimintatavoista, vahvuuksista ja heikkouksista sekä käytännön sovellutuskohteista. Kirjallisuusselvitys toimi tukena empiiriselle tutkimukselle, jossa tekninen järjestelmä suunniteltiin ja toteutettiin kirjallisuusosissa tehtyjen havaintojen perusteella. Empiirisessä osuudessa toteutettiin vahvaa tunnistamista hyödyntävä tiedonhakupalvelu ja sitä käyttävä, tutkimuksen päätavoitteena ollut Windows Phone 8.1 -prototyyppisovellus. Mobiilisovelluksen avaintoimintoihin kuului tuleviin asiakastapaamisiin liittyvien tietojen tarkasteleminen ja asiakastapaamisten kuittaaminen suoritetuiksi. Toteutettu sovellus laitettiin yrityksen Microsoft Company Hubiin jakoon ja tunnistamispalveluun luotiin kymmenelle käyttäjälle testitunnukset.

Mobiilisovelluksen tunnistamistavaksi valittiin aikaan perustuvat kertakäyttösalasanat. Käyttäjäkohtaiset salaiset avaimet ja muu sovelluksen käyttämä tieto, kuten palvelun yhteysosoite, generoitiin QR-koodiin, joka voitiin siirtää mobiilisovellukseen puhelimen kameran avulla. Mobiilisovelluksen käynnistys suojattiin PIN-koodilla, joiden arvaamista myös rajoitettiin. Salainen avain ja PIN-koodi salattiin sovelluksen omaan tietovarastoon väärinkäytöstilanteiden estämiseksi.

Tiedonhakupalvelu toteutettiin salatun HTTPS-yhteyden yli REST-rajapintana, jota ei integroitu käytössä oleviin käyttäjä-tai asiakastietojärjestelmiin. REST-rajapinta palautti asiakastiedot mobiilisovellukselle JSON-muodossa. Tiedonhakupalveluun toteutimme myös käyttäjien tunnistamisen, jossa hyödynsimme käyttäjätunnuksen ja TOTP-

salasanan lisäksi laitetunnuksia ja aikaleimoja. Niiden avulla pystyimme rajoittamaan palvelun käytön oikeutetuille laitteille ja pystyimme kompensoimaan käyttäjien aikaeroja käytettävyyden parantamiseksi.

Tutkimustyön empiirisen osuuden yhteydessä toteutettu järjestelmä osoittautui toimivaksi ja sujuvaksi käyttää. TOTP-salasanojen generoinnin integroiminen mobiilisovellukselle ja palvelinpuolelle oli yksikertaista ja siihen löytyi hyvät ja luotettavat referenssitoteutukset. TOTP-salasanojen ehdoton vahvuus onkin, että niitä voidaan generoida ohjelmallisesti laitteella kuin laitteella ja toiminta on upotettavissa mihin tahansa sovellukseen. Windows Phone -alusta sisälsi hyödyllisiä tietoturvatoimintoja, kuten sovelluskohtaisen salatun tietovaraston. Tätä hyödyntämällä saimme suojattua laitteelle säilötyn tiedon. Prototyyppisovelluksen kehitys onnistui hyvin ja kokemusten perusteella varmistuttiin tunnistamistavan soveltuvuudesta myös oikeassa järjestelmässä. Prototyyppisovelluksen ja tunnistustavan integroiminen muuhun järjestelmään ei kuulunut tutkimuksen piiriin ja se jää mahdolliseksi jatkokehityskohteeksi. Prototyyppisovelluksen jatkokehityksestä ei ole toistaiseksi tietoa, eikä vastaavan sovelluksen tekemisestä muillekaan alustoille ole ollut keskustelua. Jatkokehitystoimenpiteisiin voisi kuulua vastaavan REST-rajapinnan lisääminen yrityksemme omaan tuotteeseen, TOTP:n hyödyntäminen tunnistamisessa sekä rekisteröinnin automatisointi ja QR-koodien generoimisen integrointi hallintakäyttöliittymään.

Käyttäjän kasvoihin tai ääneen perustuva tunnistus on ollut pidempään mahdollista, mutta niiden hankaluudeksi muodostuu olosuhteista, kuten valaistuksesta ja taustamelusta, johtuvat erot mittaustuloksissa. Kontrolloidussa ympäristössä niiden käyttö on mahdollista, mutta käyttö on rajattua kunnes mittauksia pystytään tekemään kehittyneemmillä tavoilla.

Tunnistustapana mobiilivarmenne on todella turvallinen, sillä se vaatii vaikeasti manipuloitavissa olevan fyysisen SIM-kortin. Mobiilivarmenne on käytettävissä toistaiseksi harvoissa kuluttajille hyödyllisissä palveluissa, ja mobiilivarmenteen hankkimista haittaa yhteensopivuusongelmat modernien dataliittymien kanssa.

Sähköpostilla tai tekstiviestillä lähetettävät kertakäyttöiset salasanat on palveluissa usein tuettuna rekisteröintiprosessin takia, mutta niiden käyttäminen mobiililaitteelta on hankalaa moniajoon liittyvien käytettävyysongelmien takia. Sähköpostiin lähetettävät salasanat ja vahvistuspyynnöt jäävät elämään lähinnä selain- tai työpöytäsovelluspohjaisissa ratkaisuihin vaihtoehtoisena tunnistamistapana tai rekisteröintivaiheessa.

Mobiililaitteille tiedotteiden lähettämiseen käytettävät push-viestit soveltuvat myös kertakäyttösalasanojen välitykseen tai tunnistamiseen. Push-viestit voidaan lähettää myös eri mobiililaitealustoille käyttämällä Azure Notification Hubia, mikä tekee palvelujen pystyttämisestä ja hallinnoimisesta helpompaa.

Kertakäyttösalasanoja generoivien mobiilisovellusten tarjonta on kasvanut viime vuosina huomattavasti ja ne soveltuvat varsinkin selainpohjaisten verkkopalvelujen tunnistamiseen. Osa kolmansien osapuolien sovelluksista, kuten Microsoft Authenticator ja Google Authenticator, ovat käytettävissä useammissa eri palveluissa. Sovelluspohjaiset tokenit ovat riittävän turvallisia ja varsinkin HOTP- tai TOTP- algoritmit ovat yhteensopivia muissa mobiililaite- tai työpöytäsovelluksissa. Lisäksi sovelluspohjaiset tokenit voidaan integroida omiin sovelluksiin käytettävyyden parantamiseksi. HOTP-ja TOTP-algoritmien perustuvien salasanageneraattorien yhteydessä tulee huolehtia laskurin tai ajan synkronoinnista sekä erityisesti salaisten avaimien turvaamisesta.

Riskiin perustuvassa tunnistamisessa voidaan hyödyntää helposti saatavilla ja tulkittavissa olevaa tietoa, kuten käyttäjien IP-osoitteita tai kirjautumisajankohtia. Riskiin perustuvan tunnistamisen hyödyntäminen on erittäin suositeltavaa, kun peruskäyttöä ei haluta häiritä kaksivaiheisella tunnistamisella, mutta korkean riskiluokituksen toiminnot halutaan suojata.

Ulkoisista tiedonsiirtokanavista QR-koodit osoittautuivat erittäin nopeiksi ja käteviksi tavoiksi siirtää tietoa nopeasti muodossa. Ne ovat erittäin turvallisia, sillä tietoa ei siirretä verkon yli. QR-koodien tai muiden ulkoisten kanavien käyttö esim. rekisteröintiprosessissa on suotavaa, jos siihen on mahdollisuus. QR-koodien käytössä pitää tosin huolehtia, että käytetyt koodit tuhotaan käytön jälkeen. Kehitetyssä järjestelmässä QR-koodin sisältämä käyttäjätunnus ja salausavain sidottiin turvallisuussyistä siihen laitetunnukseen, joka tunnukset ensimmäisenä otti käyttöön rekisteröintivaiheessa.

Vuonna 2015 julkaistut ensimmäiset Windows 10 Mobile puhelinmallit Lumia 950 ja Lumia 950 XL tukivat biometrisista tunnisteista silmän iirikseen ja kasvoihin perustuvaa tunnistusta. Microsoft on myös aivan hiljattain julkaissut tiedon, että Windows 10 Mobile saa tuen sormenjälkitunnistukselle kesällä 2016. Sormenjälkiskannerin käyttö tulee myös mahdolliseksi omissa sovelluksissa. Uskoisin näiden uudistuksien vaikuttavan merkittävästi biometristen tunnistustapojen yleistymiseen Windows 10 Mobile käyttäjien keskuudessa. Biometrisia tunnisteita ei päästä käyttämään Windows Phone 8.1 -laitteilla, joissa ne eivät ole tuettuja. Tuetuilla Windows 10 Mobile -laitteilla iirikseen ja kasvoihin perustuvat tunnistustavat ovat vielä hitaita, joten tehokäyttäjät saattavat vielä pitäytyä vanhanaikaisessa PIN-koodissa, ellei tunnistusprosessia saada nopeutettua tulevilla ohjelmistopäivityksillä.

Windows 10 Mobile-alustan suosio jää nähtäväksi, sillä Microsoft on ilmoittanut lopettavansa omien uusien Windows 10 Mobile-laitemallistonsa päivittämisen. Alustan menestykseen tulee vaikuttamaan sovelluskaupan tarjonta ja muiden laitevalmistajien kiinnostus alustaa kohtaan. Tästä huolimatta Microsoft Passport vaikuttaa todella lupaavalta ja koska sen hyödyntäminen omissa sovelluksissa on mahdollista, voisi sen ja erityisesti biometristen tunnisteiden tarkempi tutkiminen ja testaaminen jatkotutkimuksen yhteydessä olla aiheellista.

LÄHTEET

[1] Mobiilivarmenne, Mobiiliasiointivarmenne varmennepolitiikka operaattoreiden mobiilivarmenteita varten, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.mobiilivarmenne.fi/documents/Mobiiliasiointivarmenne-Varmennepolitiikka.pdf

[2] Mobiilivarmenne, Usein kysytyt kysymykset, verkkosivu. Saatavissa (viitattu 21.5.2016):

[3] Mobiilivarmenne, Mobiilivarmenne lisää turvallisuutta, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.mobiilivarmenne.fi/fi/security

[4] DNA Mobiilivarmenne, Tunnistaudu turvallisesti, verkkosivu. Saatavissa (viitattu 21.5.2016): https://www.dna.fi/mobiilivarmenne

[5] RFC 6238, TOTP: Time-Based One-Time Password Algorithm, Internet Engineering Task Force, 2011, verkkosivu. Saatavissa (viitattu 21.5.2016): https://tools.ietf.org/html/rfc6238

[6] RFC 4226, HOTP: An HMAC-Based One-Time Password Algorithm, Network Working Group, 2005, verkkosivu. Saatavissa (viitattu 21.5.2016): https://tools.ietf.org/html/rfc4226

[7] John R. Vacca, Computer and Information Security Handbook, 2nd Edition, Morgan Kaufmann Publishers, USA, 2012, 1200 p.

[8] Lorrie Faith Cranor, Simson Garfinkel, Security and Usability: Designing Secure Systems that People Can Use, O'Reilly Media Inc, USA, 2005, 740 p.

[9] Akhil Sakai, Sven Graupner, Web Services in the Enterprise: Concepts, Standards, Solutions, and Management, 1st Edition, Springer Science Business Media, USA, 2007, 312 p.

[10] Jason Andress, The Basics on Information Security: Understanding the Fundamentals of InfoSec in Theory and Practice, 1st Edition, Syngress, USA, 2014, 240 p.

[11] Dobromir Todorov, Mechanics of User Identification and Authentications: Fundamentals of Identity Management, CRC Press, USA, 2007, 760 p.

[12] Derrick Rountree, Ileana Castrillo, The Basics of Cloud Computing: Understanding the Fundamentals of Cloud Computing in Theory and Practice, Newnes, USA, 2013, 172 p.

[13] Ertem Osmanoglu, Identity and access management: Business Performance through Connected Intelligence, Newnes, USA, 2013, 648 p.

[14] RFC 5280, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, Network Working Group, 2008, verkkosivu. Saatavissa (viitattu 22.5.2016): https://tools.ietf.org/html/rfc5280

[15] Bernard Candaele, Dimitrios Soudris, Iraklis Anagnostopoulos, Trusted Computing for Embedded Systems, Springer, 2015, 299 p.

[16] Markus Jakobsson, Steven Myers, Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft, John Wiley & Sons, USA, 2006, 739 p.

[17] Microsoft, Microsoft selling feature phone business to FIH Mobile Ltd. and HMD Global Oy, verkkosivu. Saatavissa (viitattu 23.5.2016): http://news.microsoft.com/2016/05/18/microsoft-selling-feature-phone-business-to-fih-mobile-ltd-and-hmd-global-oy/#sm.000079sble1bwgeziuqab2ih9wvs6

[18] Simson Garfinkel, Gene Spafford, Web security, privacy & Commerce, 2nd Edition, O'Reilly Media Inc, USA, 2002, 756 p.

[19] Atul Kahate, Cryptography and Network Security, 3rd Edition, Tata McGraw-Hill Education, Intia, 2013, 501 p.

[20] Badrinarayanan Lakshmiraghavan, Pro ASP.NET Web API Security: Securing ASP.NET Web API, Apress, 2013, 416 p.

[21] Mike Shema, Hacking web apps: Detecting and Preventing Web Application Security Problems, Newnes, USA, 2012, 273 p.

[22] Joseph Migga Kizza, Guide to Computer Network Security, Springer Science & Business Media, USA, 2008, 476 p.

[23] Eric F Crist, Jan Just Keijser, Mastering OpenVPN, Packt Publishing Ltd, UK, 2015, 364 p.

[24] Dieter Gollmann, Felix C. Freiling, Information Security: 15th International Conference, ISC 2012, Passau, Germany, 19-21 Sept, 2012

[25] Micsosoft, Microsoft Social MSDN, Access Sim card information, verkkosivu. Saatavissa (viitattu 23.5.2016): https://social.msdn.microsoft.com/Forums/en-US/87a460f6-cdd9-47cc-9f18-88a49f3e09dc/access-sim-card-information?forum=winappswithcsharp

[26] Stamatios V. Kartalopoutos, Security of Information and Communication Networks, John Wiley & Sons, USA, 2009, 344 p.

[27] ITSECURITY, VPN: The Pros and Cons, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.itsecurity.com/features/vpn-popularity-021108/

[28] RFC 5246, The Transport Layer Security (TLS) Protocol Version 1.2, Network Working Group, 2008, verkkosivu. Saatavissa (viitattu 21.5.2016): https://tools.ietf.org/html/rfc5246

[29] Javier Lopez, Xinyi Huang, Ravi Sandhu, Network and system security: 7th International Conference, NSS 2013, Madrid, Spain, 3-4 June, 2013

[30] Henry Lee, Eugene Chuvyrov, Beginning Windows Phone App Development, Apress, 2012, 548 p. Saatavissa:

[31] Duo Security, Duo Mobile, verkkosivu. Saatavissa (viitattu 21.5.2016): https://duo.com/solutions/features/authentication-methods/duo-mobile

[32] Katherine Snedden, Mobilecommons, Benefits of Text Messaging vs. Push Notifications, verkkosivu. Saatavissa (viitattu 21.5.2016): https://www.mobilecommons.com/blog/2014/07/benefits-of-text-messaging-vs-push-notifications/

[33] Cisco, Cisco AnyConnect Secure Mobility Client, Release Notes for Cisco AnyConnect Secure Mobility Client, Release 4.1.x for Windows 10 Mobile and Phone 8.1, verkkosivu. Saatavissa (viitattu 21.5.2016): https://www.cisco.com/c/en/us/td/docs/security/vpn_client/anyconnect/anyconnect41/release/notes/b_Release_Notes_Windows_Phone_AnyConnect_4-1-x.html#concept_572FE97EF3FF465F974373ADCA8CFE42

[34] RFC 2637, Point-to-Point Tunneling Protocol (PPTP), Network Working Group, 1999, verkkosivu. Saatavissa (viitattu 22.5.2016): https://www.ietf.org/rfc/rfc2637.txt

[35] George Ponder, Windows Central, Microsoft's beefing up security with Windows Phone 8 may make custom ROMs a thing of the past, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.wpcentral.com/microsoft-beefing-security-windows-phone-8

[36] Alex Plaskett, Dave Chismon, MWR InfoSecurity, Security Considerations in the Windows Phone 8 Application Environment, verkkosivu. Saatavissa (viitattu 21.5.2016): https://www.mwrinfosecurity.com/articles/security-considerations-in-the-windows-phone-8-application-environment/

[37] Microsoft, Windows Phone 8.1 Security Overview, verkkosivu. Saatavissa (viitattu 21.5.2016): https://www.microsoft.com/en-us/download/details.aspx?id=42509

[38] RFC 2411, IP Security Document Roadmap, Network Working Group, 1998, verkkosivu. Saatavissa (viitattu 22.5.2016): https://www.ietf.org/rfc/rfc2411.txt

[39] Cisco, Windows Phone User Guide for Cisco AnyConnect Secure Mobility Client, Release 4.1.x, 2016, verkkosivu. Saatavissa (viitattu 22.5.2016): http://www.cisco.com/c/en/us/td/docs/security/vpn_client/anyconnect/anyconnect41/user/guide/b_Windows_Phone_AnyConnect_User_Guide_4-1-x.pdf

[40] Yubico, YubiKey Frequenty Asked Questions, verkkosivu. Saatavissa (viitattu 21.5.2016): https://www.yubico.com/faq/yubikey/

[41] Blizzard, Battle.net Authenticator, verkkosivu. Saatavissa (viitattu 21.5.2016): https://eu.battle.net/support/en/article/battlenet-authenticator

[42] Richard Devine, WindowsCentral, How to set up Windows Hello facial recognition in Windows 10, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.windowscentral.com/how-set-windows-hello-facial-recognition-windows-10

[43] Claudio A. Ardagna, Jianying Zhou, Information Security Theory and Practice: Security and Privacy of Mobile Devices in Wireless Communication: 5th IFIP WG 11.2 International Workshop, WISTP 2011, Heraklion, Crete, Greece, June 1-3, 2011

[44] Microsoft Azure, Notification Hub Pricing, verkkosivu. Saatavissa (viitattu 21.5.2016): https://azure.microsoft.com/en-us/pricing/details/notification-hubs/

[45] Erik Wilde, Cesare Pautasso, REST: From Research to Practice, Springer, USA, 2011

[46] Daniel Rubino, WindowsCentral, New Microsoft Lumia 950 and Lumia 950 XL slides confirm phones and specs, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.windowscentral.com/new-microsoft-lumia-950-and-lumia-950-xl-slides

[47] John Markoff, Ny Times, SecurID Company Suffers a Breach of Data Security, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.nytimes.com/2011/03/18/technology/18secure.html?_r=0

[48] Nordea, Miten käytän tunnuslukusovellusta asioidessani mobiilipankissa, verkkosivu. Saatavissa (viitattu 21.5.2016): http://video.nordea.fi/video/miten-

kaeytaen-tunnuslukusovellusta-asioidessani-mobiilipankissa-nordea-pankki?clip_spot=video-page_related_1

[49] Google, Google Authenticator, verkkosivu. Saatavissa (viitattu 21.5.2016): https://play.google.com/store/apps/details?id=com.google.android.apps.authenticator2&hl=fi

[50] Nordea, Tunnuslukusovellus, Pankkitunnukset uudistuvat - uusi tunnuslukusovellus nyt käytettävissä, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.nordea.com/fi/media/uutiset-ja-lehdistotiedotteet/News-fi/2015/2015-06-24-pankkitunnukset-uudistuvat-uusi-tunnuslukusovellus-nyt-kaytettavissa.html

[51] Mathew J. Schwartz, Dark Reading, Lockheed Martin Suffers Massive Cyberattack, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.darkreading.com/risk-management/lockheed-martin-suffers-massive-cyberattack/d/d-id/1098013?

[52] Laura Noonan, Daniel Thomas, CNCB, Deutsche Bank tests password-free mobile security, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.cnbc.com/2015/11/22/deutsche-bank-tests-password-free-mobile-security-including-location-facial-recognition.html

[53] Microsoft, Authenticator, verkkosivu. Saatavissa (viitattu 21.5.2016): https://www.microsoft.com/en-us/store/apps/authenticator/9wzdncrfj3rj

[54] Tom, Rowan, VPN technology: IPSEC vs SSL, Network Security Vol. 2007, Iss. 12, 2007, p. 13-17

[55] Microsoft, Todentajasovellukset: usein kysytyt kysymykset, verkkosivu. Saatavissa (viitattu 21.5.2016): http://windows.microsoft.com/fi-fi/Windows/identity-verification-apps-faq

[56] Microsoft, Microsoft Azure Notification Hubs, verkkosivu. Saatavissa (viitattu 21.5.2016): https://azure.microsoft.com/en-us/services/notification-hubs/

[57] Galen Gruman, InfoWorld, Mobile security: iOS vs. Android vs. BlackBerry vs. Windows Phone, InfoWorld, verkkosivu, Saatavissa: (viitattu 20.5.2016) http://www.infoworld.com/article/2987635/mobile-security/mobile-security-ios-vs-android-vs-blackberry-vs-windows-phone.html

[58] Leo Kelion, BBC, MWC 2016: Mastercard rolls out selfie ID checks, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.bbc.com/news/technology-35631456

[59] Chris Foxx, BBC, Mastercard testing facial recognition security app, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.bbc.com/news/technology-33379461

[60] DNA, DNA Mobiilivarmenne hinnat, verkkosivu. Saatavissa (viitattu 21.5.2016): https://mobiilivarmenne.dna.fi/mc/register/start#

[61] Alex Plaskett, Nick Wlaker, MWR Info Security, Windows Phone 8 Application Security Whitepaper, 2014, verkkosivu. Saatavissa (viitattu 23.5.2016): https://labs.mwrinfosecurity.com/system/assets/651/original/mwri_wp8_appsec-whitepaper-syscan_2014-03-30.pdf

[62] Saikat Basu, Makeuseof, 5 Web Apps Which Show Us How To Make Use Of QR Codes Differently, verkkosivu. Saatavissa (viitattu 23.5.2016): http://www.makeuseof.com/tag/5-web-apps-show-qr-codes-differently/

[63] Jamie Tolentino, The Next Web, verkkosivu. Saatavissa (viitattu 21.5.2016): http://thenextweb.com/future-of-communications/2015/02/09/sms-vs-push-vs-email/#gref

[64] Authy, Authy is the best rated two-factor authentication app, verkkosivu. Saatavissa (viitattu 21.5.2016): https://www.authy.com/app/mobile/

[65] Mobiilivarmenne, Tiedotteet: Elisa ja Oma Säästöpankki aloittavat yhteistyön mobiilitunnistamisessa, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.mobiilivarmenne.fi/fi/bulletin/elisa-ja-oma-saastopankki-aloittavat-yhteistyon-mobiilitunnistamisessa

[66] Elisa, Elisa Mobiilivarmenteen rekisteröinti ja hinnasto, verkkosivu. Saatavissa (viitattu 21.5.2016): https://elisa.fi/varmenne/hinnat/

[67] Al-Sakib Khan Pathan, Security of Self-Organizing Networks: MANET, WSN, WMN, VANET, CRC Press, USA, 2016, 638 p.

[68] Rao, H.R., Managing Information Assurance in Finacial Services, Idea Group Inc, USA, 2007, 346 p.

[69] Sonera, Sonera Mobiilivarmenne hinnasto, verkkosivu. Saatavissa (viitattu 21.5.2016): https://www.sonera.fi/media/131a92ac388f7417f8e32accf2a302b640e9059b/131a92ac388f7417f8e32accf2a302b640e9059b.pdf

[70] Labyrintti, SMS Gateway, verkkosivu. Saatavissa (viitattu 21.5.2016): https://www.labyrintti.com/sms-gateway

[71] Jesper M. Johansson, Steve Riley, Protect Your Windows Network: From Perimeter to Data, Addison-Wesley Professional, USA, 2005, p. 578

[72] RFC 7525, Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS), Internet Engineering Task Force, 2015, verkkosivu. Saatavissa (viitattu 21.5.2016): https://tools.ietf.org/html/rfc7525#section-6.3

[73] Henk C. A. van Tilborg, Sushil Jajodia, Encyclopedia of Cryptography and Security, Springer, 2011

[74] Mark Stamp, Information Security: Principles and Practice, John Wiley & Sons, USA, 2011, p. 606

[75] Weizhi Meng, Duncan S. Wong, Lam-For Kwok , (2014), "The effect of adaptive mechanism on behavioural biometric based mobile phone authentication", Information Management & Computer Security, Vol. 22 Iss 2 p. 155 - 166

[76] Sabrina De Capitani di Vimercati, Chris Mitchell, Public Key Infrastructures, Services and Applications,9th European Workshop, EuroPKI 2012 Pisa, Italy, September 2012 Revised Selected Papers

[77] Carl Campanile, New York Post, DEM POL'S SON WAS 'HACKER', verkkosivu. Saatavissa (viitattu 21.5.2016): http://nypost.com/2008/09/19/dem-pols-son-was-hacker/

[78] Laki vahvasta sähköisestä tunnistamisesta ja sähköisistä allekirjoituksista, 7.8.2009/617, 2009, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.finlex.fi/fi/laki/ajantasa/2009/20090617

[79] Daniel Rubino, Windows Central, Building biometric authentication into your Windows 10 app is easy with Windows Hello, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.windowscentral.com/microsoft-details-how-developers-can-easily-use-windows-hello

[80] Daniel Rubino, Windows Central, How to configure LastPass to work with the Surface Fingerprint ID Type Cover, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.windowscentral.com/setup-lastpass-surface-fingerprint-type-cover

[81] Daniel Rubino, Windows Central, Dropbox and Microsoft announce new universal Windows 10 app and expanded partnership, verkkosivu. Saatavissa (viitattu 21.5.2016): http://www.windowscentral.com/dropbox-and-microsoft-announce-new-universal-windows-10-app

[82] Windows Apps Team, Windows, Convenient two-factor authentication with Microsoft Passport and Windows Hello, verkkosivu. Saatavissa (viitattu 21.5.2016): https://blogs.windows.com/buildingapps/2016/01/26/convenient-two-factor-authentication-with-microsoft-passport-and-windows-hello/

[83] Alexander Koren, Microsoft, Microsoft Passport and Windows Hello, verkkosivu. Saatavissa (viitattu 21.5.2016): https://msdn.microsoft.com/en-us/windows/uwp/security/microsoft-passport

[84] Microsoft, Windows 10 puhelimellesi on täällä, verkkosivu. Saatavissa (viitattu 21.5.2016): https://www.microsoft.com/fi-fi/windows/windows-10-mobile-upgrade?4161f8f5-7e1c-4e98-a5ed-28262407d074=1

[85] InstantSSL, What is HTTPS, verkkosivu. Saatavissa (viitattu 21.5.2016):https://www.instantssl.com/ssl-certificate-products/https.html

[86] Erik Wilde, Cesare Pautasso, REST: From Research to Practice, Springer Science & Business Media, 2011, USA, p. 528

[87] The Transport Layer Security (TLS) Protocol Version 1.3 draft-ietf-tls-tls13-12,Network Working Group,2016,Saatavissa: (viitattu 20.5.2016) https://tools.ietf.org/html/draft-ietf-tls-tls13-12