Digitaaliset todennukset mobiiliympäristössä

Taneli Virkkala

Kandidaatin tutkielma
HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos

Helsinki, 20. huhtikuuta 2014

HELSINGIN YLIOPISTO — HELSINGFORS UNIVERSITET — UNIVERSITY OF HELSINKI

Tiedekunta — Fakultet — Faculty	Laitos — Institution — Department
Matemaattis-luonnontieteellinen	Tietojenkäsittelytieteen laitos

Tekijä — Författare — Author
Taneli Virkkala

Työn nimi — Arbetets titel — Title

Digitaaliset todennukset mobiiliympäristössä

Oppiaine — Läroämne — Subject
Tietojenkäsittelytiede

Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
Kandidaatin tutkielma	20. huhtikuuta 2014	21

Tiivistelmä — Referat — Abstract

Mobiililaitteiden teknologinen kehitys on mahdollistanut monien viestien vaihtoon soveltuvien protokollien käytön nykypäivänä. Näihin protokolliin kuuluu laitteen käyttäjän todennus, viestin lähettäjän todennus, viestin datan eheys ja saapuminen oikealle vastaanottajalle. Digitaalinen allekirjoitus, MAC-funktio sekä Diffien ja Hellmanin menetelmä mahdollistavat turvallisen viestien vaihton kahden osapuolen välillä.

Mobiililaite koostuu SIM-kortista ja prosessorista. Digitaaliseen allekirjoitukseen tarvittavaa avainta voidaan säilyttää laitteen muistissa tai SIM-kortilla. Digitaalisen allekirjoituksen voi tehdä kokonaan laitteen avulla tai delegoida operaation palvelimelle. Palvelimelle on hyvä luoda turvattu VPN-yhteys, jossa kaikki tieto voi kulkea salattuna.

Luotettavaan tietoturvaan kuuluu lisäksi käyttäjän todentaminen mobiililaitteelle. Koska todentamisen tulee olla nopeaa, tehokkuus on otettava huomioon protokollien käytössä.

ACM Computing Classification System (CCS):

Avainsanat — Nyckelord — Keywords
mobiiliympäristö, digitaalinen todennus, digitaalinen allekirjoitus, MAC-funktio, RSA, Diffie-Hellman

Säilytyspaikka — Förvaringsställe — Where deposited

Muita tietoja — Övriga uppgifter — Additional information

Sisältö

1 Johdanto

Digitaalisten allekirjoitusten käyttö on kasvanut huomattavasti mobiililaitteilla nykypäivänä. Mobiiliympäristössä turvallinen yhteys on varmistettava, koska tietoturvariskit langattomissa verkoissa ovat erittäin suuret [SY13]. Teknisen kehityksen ansiosta allekirjoituksia voidaan luoda yleisesti mobiililaitteilla, ja parempi tietoturva on mahdollistanut useiden sovellusten käytön. Tietokonelaitteistolla käytettävät menetelmät, kuten esimerkiksi PKI-malli (julkisen avaimen infrastruktuuri), ovat siirtyneet mobiiliympäristöön sellaisenaan, eivätkä nämä mallit ole tarvinneet suuria muutoksia toimiakseen. Kehittyneemmän laskentatehon ansiosta monet algoritmit, kuten RSA sekä Diffien ja Hellmanin menetelmä, on pystytty ottamaan käyttöön kannettavilla laitteilla [SY13]. Yhteiseen salaisuuteen perustuva MAC-funktio on myös käytössä tekstiviestien (SMS) lähettämisessä [SCP12]. Operaatiot, joita laitteella ei pystytä suorittamaan, voidaan delegoida palvelimille. Allekirjoitus voidaan luoda tarvittaessa palvelimella tai asiakkaan laitteessa, mutta allekirjoituksen tulee täyttää kaikki sille asetetut tietoturvaehdot. Palvelinpohjaisen allekirjoituksen yleensä luo välissä oleva kirjautumispalvelin eikä lopullinen palveluntarjoaja [SSA10].

Digitaalisten allekirjoitusten käyttö tulisi mobiiliympäristössäkin olla nopeaa ja turvallista. Monet nykyaikaiset sovellukset vaativat jokaisen viestin lähetyksen yhteydessä uuden allekirjoituksen. On siis luotava allekirjoitus nopean ajan sisällä tietoturvasta karsimatta. Esimerkkinä tästä voisi toimia eräänlainen huutokauppasovellus, jossa jokaisen huudon on oltava kiistaton ja todennettu [SSA10]. Lisäksi viestin sisältämän datan tulee olla eheää ja varmennettu. Varmenne on varmenneviranomaisen tarjoama osa digitaalisen allekirjoituksen luontiin. Jokainen allekirjoitus tarvitsee varmenteen, jonka lähettäjä liittää allekirjoitukseen [RB12].

Schwabin ja Yangin mukaan [SY13] mahdollisiin tietoturvariskeihin mobiiliympäristössä liittyy urkinta, välimieshyökkäys, datan muuntaminen, toisena osapuolena esiintyminen ja laitteen kadottaminen. Mobiililaitteen sisäisen toiminnan ja verkkoviestinnän tulee olla suojattu mahdollisilta riskeiltä. Jos palvelun tarjoajan ja asiakkaan välissä on välityspalvelin, siihen tulee myös muodostaa luotettava yhteys. Koska digitaaliset allekirjoitukset perustuvat julkisen avaimen infrastruktuuriin, on äärimmäisen tärkeää pitää salainen avain mahdollisimman turvassa. Laitteen SIM-kortti on turvallinen paikka säilyttää salaista avainta, joka ei silloin paljastu laitteen käyttöjärjestelmälle. Sen sijaan prosessorilla avaimen ja allekirjoituksen luonti on nopeampaa kuin SIM-kortilla, mutta silloin käyttöjärjestelmä näkee laitteen prosessorin luoman avaimen kokonaan [SSA10].

2 Digitaaliset todennukset

Digitaalisiin todennuksiin kuuluvat epäsymmetristä todennusta käyttävä digitaalinen allekirjoitus ja muut julkisen avaimen infrastruktuuriin pohjautuvat menetelmät. Symmetrisiä tapoja todentaa ovat yhteiseen salaisuuteen perustuvat Diffien ja Hellmanin menetelmä sekä MAC-funktio. Kaikkien todennusten on täytettävä seuraavat kriteerit: viesti on lähtenyt oikealta lähettäjältä, viesti on saapunut muuttumattomana perille ja vain vastaanottaja pystyy lukemaan viestin. Vastaanottaja pystyy kuitenkin muuttamaan viestiä saatuaan sen tai kiistämään viestin saapumisen perille.

2.1 Digitaalisen allekirjoituksen määritelmä

Digitaalinen allekirjoitus on menetelmä, jolla voidaan todentaa tietyn lähettäjän lähettäneen viestin vastaanottajalle muuttumattomana. Lähettäjä ei voi siis jälkikäteen kiistää lähettämäänsä viestiä (kiistämättömyys). Viesti voi olla lisäksi salattu käyttäen jotain salausmenetelmää kuten esimerkiksi AES-lohkosalausta. Vaikka viestin salaaminen ei suoranaisesti liity digitaaliseen allekirjoitukseen, on salaus tärkeä osa tietoturvaa. Digitaalinen allekirjoitus pohjautuu julkisen avaimen infrastruktuuriin, jossa käytetään kahta eri avainta salaukseen ja purkamiseen. Allekirjoitus vaatii toimiakseen tiivistefunktioita ja varmenteen varmenneviranomaiselta. Viesti on siis saapunut oikeana perille, mikäli viestin tiiviste on sama kuin allekirjoitettu tiiviste. Mobiiliallekirjoituksella puolestaan tarkoitetaan digitaalista allekirjoitusta mobiililaitteella.

2.2 Toiminta

Tiivisteistä ja datan eheydestä voidaan todentaa tiedon muuttumattomuus ja lähettäjän kiistämättömyys [Cam03]. Vastanottaja verifioi viestin allekirjoituksen allekirjoittajan lähettämällä julkisella avaimella eli purkaa allekirjoitetun tiivisteen salauksen. Lisäksi viestistä tulee laskea tiiviste ennalta sovitulla tunnetulla tiivistefunktiolla. Molempien tiivisteiden arvojen tulisi olla samat, jotta viesti voidaan hyväksyä. Jos viestin sisältöä muutetaan, viestistä laskettu tiiviste ei ole enää sama vastaanottajan avaimella puretun allekirjoitetun tiivisteen kanssa. Tiivistefunktioina voidaan käyttää SHA-2 ja MD5. On syytä huomata, että SHA-1 ja MD5 ovat jo vanhentuneita tietoturvan kannalta ja uusi standardi SHA-3 on tuloillaan [nis14].

Viestin varmenteessa on mukana lähettäjän liittämä julkinen avain, jollei sitä ole lähetetty aiemmin. Varmenteen kelpoisuus on varmistettava varmenneviranomaiselta. Myönnetty varmenne on voimassa vain tietyn aikaa. Vastaanottajan on tunnettava kyseinen varmenneviranomainen viestin kelpoisuuden tarkastamiseksi. Lisäksi varmenteen voimassaolo on tarkistettava. Kelpoisuus on siis tarkastettava viranomaisen julkisella avaimella, joka on saa-

tettu lähettää saman viestin mukana. Seuraavien ehtojen on oltava voimassa allekirjoituksessa: uskottavuus, muuttumattomuus, uudelleenkäyttämättömyys ja kiistattomuus [TX10]. Digitaalisella allekirjoituksella voidaan siis todentaa vain yksi viesti kerrallaan, ja jokaiselle viestille on luotava uusi allekirjoitus. Menetelmä on yksi turvallisimmista tavoista varmentaa luotettava viestinkulku vastaanottajan ja lähettäjän välillä. Sen sijaan allekirjoitus on raskasta luoda, joten menetelmä vaatii merkittävää laskentatehoa toimiakseen [SSA10]. Erityisen vaikeaksi tekee tilanne, jossa salausta joudutaan käyttämään joka palvelimelle siirryttäessä. Digitaalisia allekirjoituksia käytetään siis yleensä yhteydenmuodostuksen aluksi, mutta niistä voidaan luopua myöhemmissä viestien lähteyksissä.

2.3 Historia

RSA algoritmina kehitettiin jo vuonna 1977. Ron Rivest, Adi Shamir ja Leonard Adleman [RSA78] keksivät julkisen avaimen infrastruktuuriin pohjautuvan menetelmän, jossa salaus hoidetaan julkisella avaimella ja viestin purkaminen salaisella avaimella. Tätä salausmuotoa kutsutaan epäsymmetriseksi salaukseksi erillisten avainten takia. Vain vuotta aikaisemmin Whitfield Diffie ja Martin Hellman [DH76] loivat Diffien ja Hellmanin menetelmän (DH), joka mahdollisti kahden osapuolen viestien salauksen. DH luo yhteisen avaimen, jolla viestittävät osapuolet voivat salata ja purkaa viestit toisillensa. Käyttötarkoitus on vain erilainen. RSA:ta käytetään digitaalisen allekirjoituksen luontiin, mutta DH perustuu käyttäjien yhteiseen salaisuuteen (symmetrinen salaus) viestin salauksen mahdollistamiseksi. Tämä kyseinen salaisuus DH:ssa voidaan sopia julkisia yhteyksiä pitkin, mutta lopputuloksena vain kaksi osapuolta saavat tietää tämän salaisuuden. DH:ssa viestin salaus ja purku tehdään samalla avaimella, eikä erillistä käyttäjäkohtaista salaista avainta ole.

Vuonna 1988 luotiin tarkat vaatimukset digitaalisille allekirjoituksille. Goldwasserin, Micalin ja Rivestin [GMR88] mukaan allekirjoitukset eivät saa noudattaa mitään selkeää kaavaa, josta selväkielinen teksti saataisiin yhdistettyä salattuun tekstiin. Julkinen avain voi olla hallussa ulkopuolisella hyökkääjällä, jolla hän purkaa salatut tekstit ja pystyy päättelemään allekirjoituksen rakenteen tuleville viesteille. DH:n aikana vuonna 1976 oli käytössä takaporttifunktio, joka oli vielä turvattomampi [DH76]. Jo pelkällä avaimen hallussapidolla kuka tahansa pystyi satunnaisella todennäköisyydellä luomaan pätevän tiivisteen viestille. Lyhyellä viestillä M ja julkisella avaimella k hyökkääjä pystyi verifioimaan lyhyen viestin, jolloin hän pääsi lähettämään ja purkamaan yksinkertaisia sanomia.

Internetin laajentuessa ympäri maailmaa 1990-luvun alkupuolella monet sähköiset palvelut yleistyivät. 1993 kehitetty ensimmäinen ohjelmointirajapinta SSL:lle oli alku myös myöhemmin käyttöön tulleelle TLS:lle. Nämä kummatkin protokollat toivat salauksen Internet-yhteyden välillä toimiviin

sovelluksiin. Esimerkiksi HTML-sivujen siirrossa voidaan käyttää turvallista HTTPS-protokollaa, jossa data kulkee salatussa muodossa. Salaus ja purku voivat noudattaa PKI-mallia käytettäessä TLS:llää. Suomessa keksitty SSH-protokolla mahdollisti puolestaan etäyhteyden asiakkaan ja palvelimen välillä. Allekirjoitusalgoritmit kuten RSA tulivat tämän kaltaisessa yhteyden muodostuksessa yleisiksi ja salaisella avaimella voitiin korvata salasanat. Nykyisin myös mobiililaitteella voi muodostaa etäyhteyksiä, ja aivan samoja TLS- ja HTTPS-protokollia käytetään osapuolten todennuksissa mobiilisovelluksissa.

2000-luvun alussa digitaaliset allekirjoitukset yleistyivät mobiililaitteissa. WAP-sovellusten (Wireless Application Protocol) tullessa käyttöön digitaalisia todennuksia pystyttiin käyttöönottaa mobiiliympäristössä ja monet yritykset alkoivat todentaa asiakkaitaan mobiiliallekirjoituksilla. Suomessa vasta vuonna 2010 kehitettiin mobiilivarmennejärjestelmä, johon mobiiliverkko-operaattorit lähtivät mukaan [mob14]. Virossa jo vuonna 2007 käynnistyi M-ID tunnistautuminen, jossa kansalainen pääsee käyttämään sähköisiä palveluita todentamalla henkilöllisyytensä matkapuhelimeltaan [est14].

2.4 Matemaattisia selityksiä

Algoritmien tulkinta vaatii pohjaksi tietämystä matemaattisista funktioista. Kongruenssirelaatiota käytetään monissa kaavoissa, joissa käytetään modulaarista aritmetiikkaa. Kongruenssirelaatio eli merkki $\equiv$ tarkoittaa kahden luvun b ja c erotuksen jaollisuutta luvulla m. Laskusta $(b - c)/m$ täytyy tulla lopputulokseksi kokonaisluku. Voidaan siis merkitä $b \equiv c \pmod{m}$, josta saadaan tulokseksi aina 0. Todetaan siis luvun b olevan kongruentti c:n kanssa $\pmod{m}$. Esimerkiksi luku 6 on kongruentti 9 $\pmod 3$. Merkitään $6 \equiv 9 \pmod 3$, koska 6 - 9 $\pmod 3$ = -3 $\pmod 3$ eli 0. [con14]

Keskenään jaottomien lukujen (suhteellinen alkuluku) idea perustuu siihen, että toista lukua ei voi päätellä toisesta jakojäännösten perusteella. Tämä sama sääntö pätee myös potenssiin korotuksissa. Oletetaan lukujen p ja q olevan alkulukuja. Koska $\text{syt}(p, q) = 1$, niin positiivisilla eksponenteilla m ja n tämä sääntö ei muutu. Voidaan todeta siis aina alkuluvulla itsellään kertomisen säilyttävän suurinpana yhteisenä tekijänä luvun 1 eli $\gcd(p^m, q^n) = 1$. [rel14]

RSA:ssa käytettävä Eulerin funktio eli $\phi(n)$ kertoo niiden positiivisten kokonaislukujen k lukumäärän ehdolla $1 \leq k \leq n$, joiden suurin yhteinen tekijä n:nän kanssa on 1. Eli luku k ja luku n ovat tällöin suhteellisia alkulukuja keskenään. Esimerkiksi $\phi(12) = 4$ sillä lukujen 1,5,7 ja 11 ainoa yhteinen tekijä luvun 12 kanssa on 1. Jos kyseessä on alkuluku p, sille on olemassa aina keskenään jaottomia lukuja $p - 1$ verran. Voidaan todeta $\phi(p) = p - 1$. Eulerin funktiolla saadaan tulokseksi lukujoukko, josta saadaan tarvittavat tekijät avainpareihin RSA:ssa. [tot14]

Primitiivijuuri tarkoittaa, että kokonaisluku g potenssiin n ja jakojäännös alkuluvusta p ehdoilla $1 \leq n \leq p$ tuottaa kaikki kokonaisluvut väliltä 1 ja

$p-1$. Kyseessä on $\phi(p) = p-1$ määrän verran lukuja. Eli $g^n \pmod p \neq g^{n+1} \pmod p$, jossa $g, p, n \in \mathbb{Z}$. Esimerkiksi luvun 2 primitiivijuuri on mod 3, koska $2^1 \pmod 3 = 2$ ja $2^2 \pmod 3 = 1$. Luvut 1 ja 2 ovat pienempiä kuin 3 ja keskenään erisuuria, joten kaikki jakojäännökset väliltä 1 ja 2 (eli 3-1) on käyty. [pri14]

Diskreettiä logaritmia pidetään tällä hetkellä ratkeamattomana alle eksponentiaalisessa ajassa. Kuvitellaan kokonaisluku a, joka on suhteellinen alkuluku luvulle n. Lisäksi g on primitiivijuuri luvulle n. Eulerin funktiolla $\phi(n)$ saadaan etsittyä joukko lukuja, joista yksi luku μ ratkaisee kaavan $a \equiv g^\mu \pmod n$. Näin ollen μ on luvun a g-kantainen diskreetti logaritmi $\pmod n$, joten voidaan todeta $\mu = \mathrm{ind}_g\ a \pmod n$. Indeksiä kuvaava ind tarkoittaa diskreettiä logaritmia. Lukua a ei voida saada selville ilman μ, vaikka g ja n olisikin tiedossa. On siis helppo laskea a mikäli μ, tunnetaan, mutta lähes mahdoton laskea μ vaikka a olisikin selvillä. [dis14]

2.5 Salaus

AES ja 3DES ovat lohkosalausalgoritmeja, jotka salaavat selväkielisen tekstin. Ne kummatkin ovat voimassa vielä nykypäivän tietoturvastandardeissa ja ne soveltuvat esimerkiksi tekstiviestien (SMS) salaukseen. Viestin salaaminen ei ole osa digitaalista allekirjoitusta, mutta salaamalla viesti voidaan estää sen paljastuminen matkalla ulkopuoliselle tarkkailijalle. Viesti voi olla ennen lähetystä aluksi salattu, jonka jälkeen tiiviste luodaan. Vastaanottaja tekee toimenpiteet käänteisessä järjestyksessä eli verifioi allekirjoituksen ja purkaa salauksen. Salaamisen luonnollisesti kuluu paljon laskentatehon resursseja. Saxenan ja Chaudharin tutkimuksessa AES oli ajallisesti ja salauksen lopputulokseltaan paras vaihtoehto tekstiviestin salauksessa. Myös purkamisessa AES oli parempi verrattuna 3DES:sään. [SC12]

2.6 Varmenteet

Varmenteet ovat kolmannen osapuolen antaman varmenneviranomaisen todistuksia. Myös välityspalvelin voi luoda varmenteen käyttäjälle [SSA10]. Jokin käyttäjä, välityspalvelin tai lopullinen palveluntarjoaja, tarvitsee varmenteen jatkuvaa yhteydenpitoa varten, koska varmenne kuuluu digitaalisen allekirjoituksen protokollaan. Varmenne voi olla voimassa päiviä, kuukausia tai vuosia, mutta tietoturvan kannalta varmenteiden ei tulisi olla ikuisia. Varmennetta voidaan pitää luotettavana, jos sen tarjoaa ulkopuolinen varmenneviranomainen. Digitaalinen allekirjoitus vaatii toimiakseen aina varmenteen, mutta se voidaan myöhemmin liittää viestiin lähetyksen jälkeen esimerkiksi välityspalvelimen toimesta [SSA10]. PKI-mallin avulla varmenne voidaan luoda luovuttamalla julkinen avain varmenneviranomaiselle ja lähettämällä varmennepyyntö. Tämän jälkeen käyttäjä vahvistaa vielä itsensä salaamalla viestinsä salaisella avaimellaan. Varmenneviranomainen vastaa luovuttamalla

varmenteen käyttäjälle. Varmenteeseen on yleensä merkitty seuraavat tiedot: voimassaoloaika, sarjanumero, versio ja käyttäjän tunniste [RB12]. Vastaanottajan tulee siis ottaa huomioon vanhentunut varmenne. Koska varmenne on yksilökohtainen, hyökkääjä ei tee varastetulla varmenteella mitään.

2.7 Julkisen avaimen infrastruktuuri

Julkinen ja salainen avain muodostavat PKI-mallin [RB12]. Digitaalinen allekirjoitus perustuu julkisen avaimen infrastruktuuriin ja siksi salaisen avaimen on pysyttävä vain lähettäjän hallussa. Sen sijaan julkinen avain annetaan vastaanottajalle, mikä mahdollistaa salatun tiivisteen purkamisen. Kyseessä on siis epäsymmetrinen salaus. Koska avaimia luodaan valtavista määristä suuria alkulukuja RSA:ssa, on toisen identtisen avainparin syntyminen erittäin epätodennäköistä. Allekirjoitus voidaan liittää viestiin tai lähettää erillisenä [Cam03]. Hyvänä pituutena tietoturvan kannalta molemmille avaimille voidaan pitää vähintään 1024 bittiä [RB12].

2.8 RSA

RSA on salausalgoritmi, joka jakojäännöksen avulla suorittaa viestin M salauksen ja purkamisen. Aluksi valitaan kaksi alkulukua p ja q, jotka eivät saa olla samat. Näiden lukujen tulo on n. Luku $\phi(n)$ saadaan selville kaavalla $(p-1)(q-1) = \phi(n)$. Tämän jälkeen valitaan kokonaisluku e väliltä $1 < e < \phi(n)$. Lisäksi e:n on oltava suhteellinen alkuluku luvulle $\phi(n)$. Vielä tulee valita luku d, joka kerrottuna e ja vähennettynä yhdellä on jaollinen mod $\phi(n)$. Eli siis $de \equiv 1 \pmod{\phi(n)}$. Tämä voidaan muuttaa muotoon $de \bmod \phi(n) = 1$. Julkinen avain on pari (n, e) ja salainen avainpari on (n, d).

Viesti M salataan E:ksi seuraavalla kaavalla julkisella avaimella:

$$E = M^e \pmod{n}$$

Vastaanottajalla on salainen avain d. Purkaminen tapahtuu kaavalla:

$$M = E^d \pmod{n}$$

Luvut n, e ja d eivät saa olla pääteltävissä toisistaan. E:stä ja e:stä ei voi päätellä viestiä M. Yleisesti tunnetusta n:stä ei voi päätellä p:tä tai q:ta. Jos p tai q olisi tiedossa, $\phi(n)$ ja avainparit voitaisiin saada selville. Tekijöihin jako on yhtä vaikea ongelma kuin diskreetti logaritmi. [mat14b]

2.9 Diffien ja Hellmanin menetelmä

Aivan RSA:n tavoin Diffien ja Hellmanin menetelmä (DH) käyttää modulaariaritmetiikkaa avainten luonnissa. Perusversio DH:sta on tietoturvaltaan altis välimieshyökkäykselle, sillä kolmas osapuoli on voinut saada julkisen

avaimen haltuunsa esimerkiksi salakuuntelun avulla. Protokolla toimii yksinkertaisuudessaan seuraavalla tavalla [mat14a].

Aloittaja A ja vastaaja V sopivat etukäteen luvuista p ja g. Alkuluku p ja sen primitiivijuuren g tulee olla molempien osapuolten tiedossa. Näiden lisäksi viestinvaihdon aloittavan osapuolen A on valittava salainen kokonaisluku a. Aloittaja lähettää vastaanottajalle V viestin $A = g^a \pmod p$. Tämän jälkeen V valitsee kokonaisluvun b, joka myös säilyy salaisena. V lähettää A:lle viestissä luvun $B = g^b \pmod p$. A laskee luvun $(g^b \pmod p)^a \pmod p$. V laskee luvun $(g^a \pmod p)^b \pmod p$.

Diffien ja Hellmanin menetelmää voidaan käyttää istuntoavaimen luomiseen yhteyden muodostamiseksi. Lisäksi DH:lla pystytään luomaan AES-istuntoavain, jolla voidaan salata viestejä tai laskea tiivisteitä. Toista osapuolta ei välttämättä tarvitse kokoajan tunnistaa digitaalisilla allekirjoituksilla, sillä yhteistä istuntoavainta voidaan käyttää myöhemmin tunnistamisessa. Tämä menetelmä säästää laskentatehoa, koska digitaaliset allekirjoitukset ovat raskaita luoda. Asiakas/käyttäjä voi olla palveluntarjoajan tiedossa jonkin aikaa, mutta pidemmän ajan kuluessa avain tulisi vaihtaa tai vastaavasti istunto sulkea. Satunnaislukujen käyttö avaimen luonnissa tekee tietoturvasta paremman. Luonnollisesti avaimen pituuden sekä alkioiden tulee olla suuria, jotta niiden arvaaminen on ulkopuoliselle tunkeutujalle vaikeampaa. [SY13]

2.10 MAC-funktio

Yhteistä salaisuutta pystytään käyttämään tiivistefunktioissa. Funktioon syötetään parametriksi jokin arvo, jolla saadaan luotua tai verifioitua tiiviste. Tämän kriteerin täyttää MAC-funktio (Message Authentication Code). MAC-funktiolla lasketaan tiiviste salaisella avaimella, joka on hallussa molemmilla osapuolilla. Jos tiivisteen arvo on odotetusti oikea, on viesti saapunut eheänä perille. Aivan kuten allekirjoituksissakin viesti salataan aluksi esimerkiksi AES-salausta käyttäen. Sen jälkeen lasketaan tiiviste käyttäen jotain julkista tunnettua MAC-funktiota. Tiiviste siis voidaan luoda avaimella tai salaisuudella, kunhan viestin vastaanottajalla on sama tieto käytössään. Vastaanottajan tulee myös tietää kyseinen MAC-funktio, jota käytetään.

Toimivalle MAC-funktiolle on olemassa selkeät ehdot. Jokaiselle viestille on oltava erilainen pätevä tiiviste. Uutta viestiä ei voi lähettää samalla tiivisteellä, eikä samalla tiivisteellä pysty lähettämään erilaista viestiä. Lähettäjä luo tiivisteen t MAC-funktiolla $S(k, m) = t$. Viestistä m vastaanottaja luo tiivisteen t salaisella avaimella k. Avain on voitu sopia aikaisemmin esimerkiksi Diffien ja Hellmanin menetelmällä. Vastaanottaja puolestaan verifioi viestistä m tiivisteen t salaisella avaimella k. Jos verifioinnista tulee odotettu arvo lopputulokseksi, lähetys on onnistunut. [DKP12]

Salasanan vaihto onnistuu MAC-funktiolla. Esimerkkinä voitaisiin ottaa käyttäjä A ja B. Luku q on jokin tunnettu alkuluku. Luku n on primitiivijuuri

luvulle p. Lisäksi n:nän on oltava pienempi kuin luku q. A:lla on salainen avain 'a' ja B:llä 'b'. Tämän lisäksi julkinen avain 'x' on A:lla ja avain 'y' B:llä. Alla oleva Saxenan, Chaudharin ja Prajatin [SCP12] esittelemä protokolla selittää salasananvaihtoa tarkemmin.

1. User A and User B know their private keys (a random number) 'a' and 'b' respectively.
2. User A calculates its password (public key) 'x' as [n^a mod q] and encrypted message digest code C_k(password) using a MAC function/algorithm.
3. User B calculates its password (public key) 'y' as [n^b mod q] and C_k(password) using a MAC function/ algorithm.
4. User A sends its ID_A and E_{k2} [(password) || C_{k1}(password)] to User B.
5. User B sends its ID_B and E_{k2} [(password) || C_{k1}(password)] to User A.
6. User A generates a shared secret key 'k' by the password of User B and its private key 'a'. { Decrypt the message digest code as D_{k2} [(password) || C_{k1}(password)] to get the password and use k1 shared key to calculate C_{k1}(password) and match with the actual C_{k1}(password) send by User B to detect the error}
   $k = y^a \bmod q = [n^b \bmod q]^a \bmod q = n^{ab} \bmod q$
7. User B generates a shared secret key 'k' by the password of User A and its private key 'b'.{[use D_{k2} [(password) || C_{k1}(password)] to get the password and use k_1 shared key to calculate C_{k1}(password) and match with the actual C_{k1}(password) send by User A to detect the error}
   $k = x^b \bmod q = [n^a \bmod q]^b \bmod q = n^{ab} \bmod q$

Yhteisenä avaimena toimii siis 'k', jolla toisen osapuolen lähettämä salasana lasketaan. MAC-funktiolla tässä tapauksessa lasketaan salasana julkisille avaimille ja tiiviste salasanalle lähetystä varten.

2.11 Mobiilikaupankäynti

Mobiilikaupankäynnillä tarkoitetaan mobiililaitteella tehtäviä maksutransaktioita tai ostotapahtuman vahvistavia viestejä. Menetelmä on siis osa elektronista kaupankäyntiä, jossa käytetään digitaalisia allekirjoituksia [TX10]. Schwab ja Yang toteavat [SY13] mobiililaitteilla henkilökohtaisen, rahallisen ja kaupallisen datan varastoinnin olevan yleisessä käytössä. Samadanin, Shajarin ja Ahanihan artikkelissa [SSA10] esitellään huutokauppasovellus, joka vaatii jokaisen huudon varmistuksen lyhyen ajan sisällä laitteella. Allekirjoitusten luonti tulee olla siis nopeaa mobiililaitteilla tietoturva huomioon ottaen. Sekä laite- että palvelinpohjaisia allekirjoituksia käytetään mobiilikaupankäynnissä [SSA10]. Verkkopankki, maksusuoritukset, terveydenhoito

ja äänestys ovat mahdollisia kannettavilla laitteilla, mutta langaton verkko tuo ongelmansa kaistanleveyden kanssa [RB12].

3 Laitepohjaiset allekirjoitukset

Mobiililaite koostuu SIM-kortista ja laitteesta, jossa allekirjoituksen luonti tapahtuu prosessorilla. Laitteen käyttöjärjestelmän on tuettava yleisesti käytettyjä protokollia, jotta salaus, tiivistefunktiot, varmenteet ja digitaaliset allekirjoitukset ovat mahdollisia. Tietoturvan kannalta SIM-korttia voidaan pitää parempana vaihtoehtona, mutta allekirjoitusten luomisen nopeudessa prosessori on tehokkaampi. Salaisen avaimen säilytyspaikka tulee kuitenkin valita turvallisesti, jotta ulkopuolinen tunkeutuja ei saa tietää salaista avainta. Lisäksi on olemassa malli, jossa SIM-kortti ja laitteen prosessori yhdessä osallistuvat allekirjoituksen luontiin (hybridimalli). Seuraavat alaotsikot perustuvat Samadanin, Shajarin ja Ahanihan malleihin [SSA10].

3.1 SIM-kortilta luonti

Laitteen SIM-korttia voidaan pitää turvallisimpana paikkana säilyttää salaista avainta. Edes käyttäjä itse tai laitteen käyttöjärjestelmä ei pääse käsiksi salaiseen avaimeen kortilla. Kuitenkin SIM-kortin laskentakapasiteetti on huomattavasti pienempi kuin laitteen prosessorin. Allekirjoituksen luonti SIM-kortilla on erittäin hidasta.

3.2 Laitteen prosessorilla luonti

Salaisen avaimen säilytys voi tapahtua myös laitteen muistissa. Digitaalinen allekirjoitus luodaan tällöin laitteen prosessorilla, joka on laskentateholtaan huomattavasti tehokkaampi kuin SIM-kortti. Käyttöjärjestelmä voi myös tarjota kirjastoja ja työkaluja allekirjoitusten luontiin. Laitteen käyttöjärjestelmässä voi kuitenkin olla tietoturva-aukko, jota hyväksikäyttäen tunkeutujat voivat saada haltuunsa käyttäjän salaisen avaimen.

3.3 Hybridimalli

Hybridimallissa salainen avain joudutaan hetkellisesti paljastamaan laitteen käyttöjärjestelmälle. Tässä menetelmässä on siis olemassa pieni tietoturvariski. Hyvänä puolena hybridimallissa on sen lähes yhtä nopea tehokkuus kuin prosessorilla luonnissa. Monet graafisen käyttöliitymän vaativat ohjelmat tarvitsevat prosessorin laskentatehoa, mutta SIM-kortti voi toimia tietoturvan kannalta avaimen yleisenä säilytyspaikkana. Mallissa allekirjoitus siis luodaan prosessorilla, jolloin salaista avainta käytetään vain hetkellisesti laitteessa.

4 Palvelinpohjaiset allekirjoitukset

Palvelin voi luoda digitaalisen allekirjoituksen käyttäjän puolesta, kunhan käyttäjä voidaan todentaa palvelimelle. Palvelinten rooli digitaalisten allekirjoitusten luonnissa oli merkittävä aikana, jolloin laitteissa ei ollut tarpeeksi tehoa allekirjoituksen luomiseen. Nykyään laitepohjaiset allekirjoitukset ovat yleistyneet. [SSA10]

4.1 Välityspalvelin

Välityspalvelin toimii siis eräänlaisena kirjautumispalvelimena käyttäjän ja lopullisen palveluntarjoajan välissä. Välityspalvelin voi luoda allekirjoituksen, mutta oikean käyttäjän todennus vaaditaan. Varmennus voi perustua algoritmeihin kuten RSA tai DSA. On myös mahdollista, että käyttäjälle tehdään varmenne, jolla hän on tunnistettavissa jatkossa palvelimelle [SSA10].

4.2 NRS ja NRR

Kiistämättömyys on olennainen osa digitaalista allekirjoitusta. NRS (Non-Repudation of Sender) tarkoittaa, lähettäjä ei voi jälkikäteen kiistää lähettäneensä viestin. NRR (Non-Repudation of Receiver) puolestaan merkitsee vastaanottajan kiistämättömyyttä. Tiivistefunktioilla varmistetaan datan eheys kuten esimerkiksi MD5:llä. Sekä lähettäjän että vastaanottajan on luotava julkiset avaimet ja merkit kirjautumispalvelimelle tunnistettavaksi. Kirjautumispalvelin pyytää varmenteen varmenneviranomaiselta ja muodostaa oman varmenteen lähettäjälle. Näin ollen kirjautumispalvelin voi jatkossa toimia pysyvämpänä vahvistajana lähettäjän ja vastaanottajan välillä. [LCJ04]

4.3 Yhdistetty allekirjoitus

Laitteella pystyy delegoimaan allekirjoituksen luonnin palvelimelle kokonaan, osittain tai valtakirjalla. Välityspalvelin voi kokonaan luoda allekirjoituksen käyttäjän salaisella avaimella. Tämä tyyppi ei ole tietoturvan kannalta suotavaa. Osittaisessa allekirjoituksessa käyttäjä luo omasta salaisesta avaimestaan välityspalvelimelle uuden avaimen. Välityspalvelimella on tällöin mahdollisuus tehdä allekirjoitus käyttäjän puolesta. Kiistattomuus nousee näissä kahdessa menetelmässä ongelmaksi. Vastaanottaja ei voi tietää, onko allekirjoitus tullut välityspalvelimelta vai käyttäjältä. Kolmas menetelmä on valtakirjan luovuttaminen välityspalvelimelle. Käyttäjä siis kertoo valtakirjallaan luovuttaneensa allekirjoitusoikeuden toiselle palvelimelle. Valtakirja luodaan käyttäjän salaisella avaimella. Valtakirjan laatiminen saattaa viedä huomattavasti aikaa ja paljon laskentatehoa. [HZ04]

4.4 Protokollapyynnöt

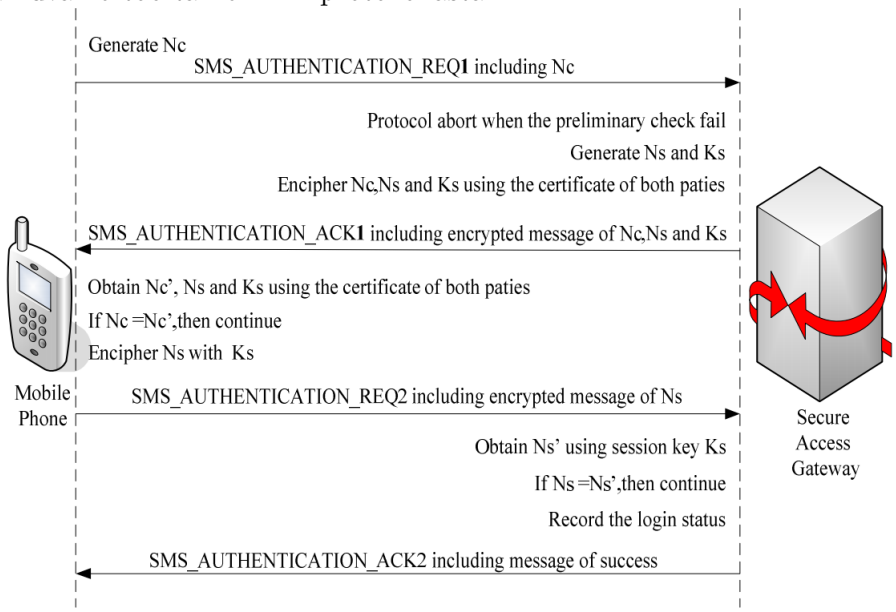
Kun viestiä lähetetään välitys- tai tarjoajapalvelimelle, on muistettava käyttää protokollan mukaisia pyyntöjä ja vastauksia laitteiden ja palvelinten välillä. He ja Zhang [HZ04] esittelevät kaksi funktiota, joita käytetään yhdistetyssä allekirjoituksessa käyttäjän, välityspalvelimen ja kirjautumispalvelimen välisessä kommunikaatiossa. Tiivistefunktion ja MAC-funktion yhdistelmä eli HMAC-funktio (Hash Message Authentication Code) varmistaa tiedon eheyden ja viestin todennuksen. Lisäksi voidaan käyttää HOAC-funktiota (Hash Origin Authentication Code), joka toimii HMAC-funktion tapaan. Käyttäjä MS lähettää viestin m, joka on allekirjoitettu HOAC-funktiolla. Tälle allekirjoitettulle tiivisteelle luodaan vielä uusi tiedon eheyttä varmistava tiiviste HMAC-funktiolla. Nämä kaikki kolme tiivistettä (m, HOAC ja HMAC) välityspalvelin HE allekirjoittaa uudelleen omalla salaisella avaimellaan. Lopullinen palvelin verifioi kaikki tiivisteet ja saa tiedon, että viesti m kulki käyttäjältä MS välityspalvelimen HE kautta. HOAC-funktio varmistaa viestin kiistämättömyyden lähettäjältä MS.

4.5 Tekstiviestitodennus

Langattomien lähiverkkotekniikoiden yleistymisestä huolimatta tekstiviestillä (SMS) voidaan todentaa käyttäjä. Tarvitaan vain terminaaliemulaattori (T) käyttäjän laitteelle ja turvallinen yhteydenmuodostusportti (GW) kommunikaatiota varten. Terminaali voi olla laitteella erillinen ohjelma tai suoraan integroitu SIM-kortilla oleviin työkaluihin. GW koostuu todennuspalvelimesta ja tietokantapalvelimesta. Todennuspalvelin varmistaa oikean ohjelman käyttävän palvelua ja tietokantapalvelin hallinnoi käyttäjän julkista avainta sekä GW:n salaista avainta.

Seuraavaksi esitelty kuva perustuu Shun, Tanin ja Wangin [STW09] tutkimuksiin turvallisesta mobiilikäyttäjän todentamisesta. Sekä T:n että GW:n on luotava tunnistemuuttuja (VF), jolla osapuolet todentavat toisensa. Muuttuja voi olla esimerkiksi tietyn pituinen tiiviste, joka perustuu satunnaislukuihin. Lähtevät paketit luonnollisesti salataan. Oletetaan tiivisteen nimeksi vaikka Nc. Terminaali tekee yhteydenmuodostuspyynnön REQ1, jossa on mukana Nc. GW tutkii pyynnön ja tarkistaa käyttäjän identiteetin sekä puhelinnumeron. Tietokantakyselyllä GW varmistaa käyttäjän löytyvän tietokannasta. GW luo oman VF:n nimeltä Ns ja istuntoavaimen nimeltä Ks. Nämä molemmat muuttujat sekä aiemman Nc:n GW allekirjoittaa salaisella avaimellaan. ACK1-vastauksessa GW hyväksyy yhteydenmuodostuksen ja lähettää salatussa muodossa Nc:n, Ns:sän ja Ks:sän. Tämän jälkeen T:llä allekirjoitetaan saapunut ACK1-vastaus käyttäjän salaisella avaimella. Jos allekirjoitettu Nc' on sama kuin käyttäjän alunperin lähettämä Nc-muuttuja, T salaa Ks:sän ja Ns:sän uudessa pyyntöviestissä REQ2 ja hoitaa lähetyksen. Nyt puolestaan GW allekirjoittaa Ns:n. Mikäli Ns' on sama kuin alunperin

lähetetty Ns, yhteys käyttäjän ja palvelimen välille on luotu ja GW lähettää terminaalille kuittauksen ACK2 yhteydenmuodostuksen onnistumisesta. Alla oleva kuva kertoo tarkemmin protokollasta.



Koska protokolla perustu satunnaisiin muuttujiin, se on resistanssi välimieshyökkäykselle. Myös kiistämättömyys luodaan, sillä käyttäjän salainen avain säilyy vain laitteessa ja terminaali käyttää avainta. Tätä kyseistä protokollaa voidaan siis pitää yhdistettynä mallina palvelin- ja laitepohjaisesta allekirjoituksesta.

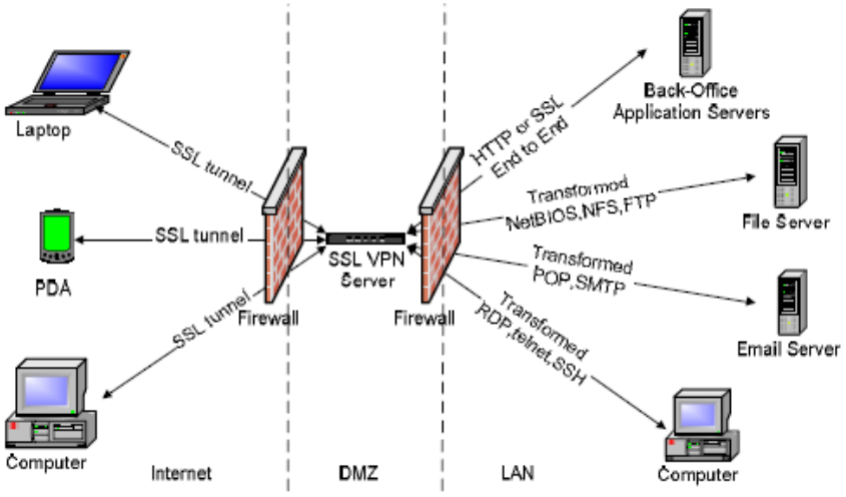
5 Turvallinen etäyhteys

On tilanteita, joissa käyttäjä haluaa muodostaa suoran yhteyden palvelimelle tai toiseen tietokoneeseen. Tällöin jatkuva viestien todentaminen osapuolten välillä käy rasittavaksi. Sen sijaan tiedon salausta on korostettava, ja käytettävän verkon turvallisuudesta huolehdittava. Voidaan muodostaa mobiililaitteen ja palvelimen välille virtuaalinen erillisverkko (VPN), jossa käyttäjä tunnistautuu palveluntarjoajalle erillisen VPN-palvelimen avulla. Osapuolten on sovittava yhteisistä standardeista ja protokollista yhteyden muodostamisen ajaksi.

5.1 Salattu VPN-yhteys

Yu, Chen ja Tan [YCT09] esittelevät turvallisen käytännön mobiililaitteelle muodostaa VPN-yhteys käyttäen SSL-salausta (Secure Sockets Layer). Täytyy kuitenkin huomata kyseessä olevan nykyaikainen turvallisempi TLS-salaus (Transport Layer Security), jota kutsutaan vain yleensä SSL:ksi. SSL:llään kuuluu paljon protokollia, joissa esimerkiksi vaihdetaan salaisia parametreja,

luodaan salausavain ja käytetään X.509-varmenteita. Mobiililaitteella tunnistaudutaan VPN SSL- palvelimelle, minkä jälkeen yhteys palveluntarjoajalle muodostetaan palvelimen toimesta. Käyttäjän ei siis tarvitse tunnistautua vielä erikseen palveluntarjoalle tässä tapauksessa. Alla oleva kuva havainnollistaa yhteyden muodostamista asiakkaan ja palveluntarjoajan välillä.



Mobiililaitteella tulee olla työkalut VPN-yhteyden, varmenteiden ja SIM-kortin hallintaan. Lisäksi salauksen/purkamisen asetuksia on päästävä muuntamaan. VPN-palvelimella on turvallisuusjärjestelmä (SAS), jossa on työkalut portin yhteysasetuksiin, yhteys varmenneviranomaisiin, salaus- tai purkuasetukset ja käyttäjätileille liittyvät hallintamekanismit. VPN-yhteys muodostetaan käyttäjän terminaalin ja VPN-portin välille. Palveluntarjoajalla ja VPN-palvelimella on omat protokollat yhteyttänsä varten.

VPN-yhteyden voi muodostaa käyttäen WLAN:ia tai GPRS:ää. Käyttäjän yhteydenmuodostuspyynnöt VPN-porttiin muistuttavat TCP-protokollaa (Transmission Control Protocol). Istuntoa varten sovitaan protokollien versoista, istunnon tunnisteesta, salausavaimesta ja käytettävistä funktioista. SSL-yhteys luodaan, kun käyttäjä on todennettu VPN-palvelimen turvallisuusjärjestelmän avulla. Seuraavaksi tarkastetaan käyttäjän oikeudet palveluun. Palveluntarjoaja antaa tarvittavan ohjelmatiedon käyttäjälle VPN-palvelimen vahvistettua käyttäjäoikeudet. Kaikki tieto käyttäjän ja VPN-palvelimen välillä kulkee salattuina paketteina. Jatkuva suojattu yhteys mahdollistaa huipputurvallisuutta vaativien sovellusten kuten verkkopankin käytön mobiiliympäristössä.

6 Käyttäjän todennus

Viestien todennusten lisäksi on tärkeää varmistaa oikean käyttäjän käyttävän laitetta. Luonnollisesti on keksittävä keino todentamiseen, jossa vain käyt-

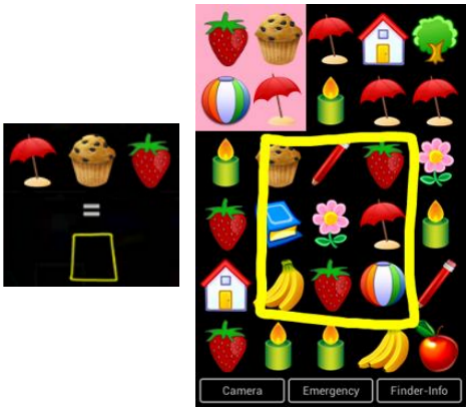
täjä tietää salaisuuden päästäkseen hallitsemaan laitetta tai lähettääkseen viestin. Schlöglhoferin ja Sametingerin [SS12] artikkelissa mainitaan mahdollisiksi todennustavoiksi PIN-koodi, salasana ja kosketuksella tunnistaminen. Tietoturvauhaksi mainitaan laitteen kadottaminen ja päätyminen väärälle käyttäjälle. Tietämykseen perustuvat tunnistamiset ovat vain oikean käyttäjän tiedossa, mutta pätevää tunnistautumista tässä tapauksessa ei voida liittää oikeaan käyttäjään. Myös hyökkääjällä on mahdollisuus arvata pätevä salasana riippuen hyväksyttyjen yritysten määrästä. Todennukset voivat perustua muuhunkin kuin oikean vastauksen tietämykseen. Uusia menetelmiä ovat elektroniset NFC-tarrat ja kameralla otettava kuva. Androidille on tarjolla sovellus nimeltä SecureLock. Sillä voi valita erilaisia tapoja todentaa käyttäjä laitteelle.

6.1 PIN-koodi ja salasanatodennus

PIN-koodi ja salasana ovat yleisesti käytettyjä tunnistautumismenetelmiä. SIM-kortille todennus tapahtuu PIN-koodilla, joka voi olla käyttäjän tai verkko-operaattorin määrittelemä. Salasanalla tunnistaudutaan osaltaan laitteelle. SecureLock-sovelluksessa kuitenkin salasana voi olla pituudeltaan rajoittamaton. Salasanassa voi olla isoja ja pieniä kirjaimia ja lisäksi sisältää erikoismerkkejä. Lisäksi on olemassa PUK-koodi (Personal unblocking code), joka voidaan syöttää laitteeseen PIN-koodin tai muun todennustavan unohduttua.

6.2 Kosketustodennus

Kosketustodennukseen kuuluu graafisen salasanan syöttäminen. SecureLock tarjoaa kosketustunnistautumisen, jossa käyttäjä rajaa sormenliikkeellään tietyn alueen laitteen ruudulta. Tähän ruutuun täytyy lisäksi sisältyä tietty määrä symboleita, jotka voivat olla esimerkiksi hedelmiä. Alla olevassa kuvassa keltainen neliö on rajattu alue, jolta täytyy löytyä sateenvarjon, suklaamuffinssin ja mansikan kuvakkeet. Käyttäjä piirtää valitsemansa alueen kosketusnäytöllä. Alueen on oltava myös oikean kokoinen, jotta todennus hyväksytään. Tässä tunnistautumistavassa yhdistyvät käyttäjän tietämys oikean alueen koosta, sijainnista ja tarvittavista symboleista.

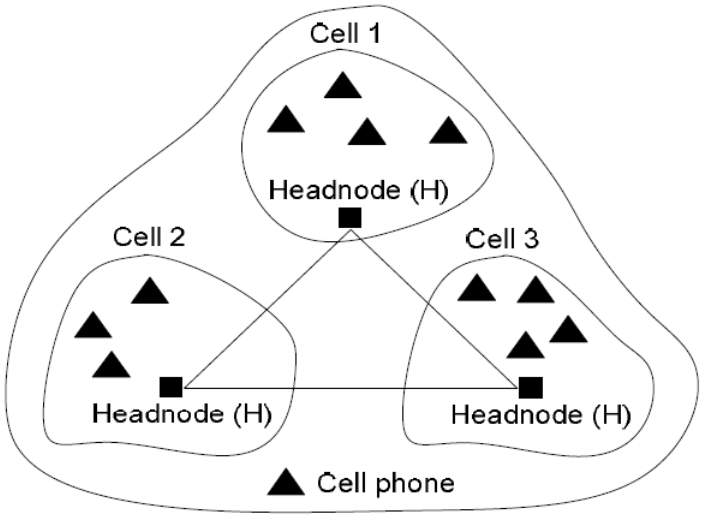


6.3 Muita todennustapoja

NFC on uusi teknologia, joka tarjoaa elektronisten tarrojen avulla tunnistautumisen laitteelle. Laite lukee tarrasta koodin, joka mahdollistaa todennuksen. Tarroja voi olla useampi ja laitteen on teknisiltä ominaisuuksiltaan tuettava NFC-todennusta. Laitteen kameralla on lisäksi mahdollisuus tunnistautua. Kuvan voi ottaa ilman tunnistautumista, mutta aikaisempia otettuja kuvia ei pysty tarkastelemaan laitteella. Kuvan kohteena voisivat toimia omat kasvot tai henkilökortti. Näin ollen käyttäjän tarvitse muistaa vain kohde, jota tarvitaan todennukseen.

6.4 Tunnistautuminen GSM-verkossa

Langattomassa verkossa on tärkeää, että viestin lähettäjä tietää lähettävänsä viestin oikealle kohteelle ja vastaanottaja tietää saavansa viestin oikealta lähettäjältä. Osapuolten sijainti ja aikaleimat saattavat muuttua viestien vaihdon aikana. Jaiswal ja Kumar [JK12] esittelevät GSM-verkon toimintaa kahden mobiililaitteen käyttäjän näkökulmasta. Tähän väliin tarvitaan pääsolmu, joka luo yhteisen avaimen osapuolille viestien vaihtoa varten. Verkko voidaan jakaa useampaan kenttään, jossa jokaisessa on oltava vähintään yksi pääsolmu laitteiden tunnistautumista varten. Pääsolmun on tiedettävä kaikki oman kenttänsä laitteet ja muita pääsolmuja tiedon vaihtoa varten. Pääsolmun luo yhteisen avaimen osapuolille tulevaa tiedonvaihtoa varten tunnistetietojen perusteella. Pääsolmu tarvitsee laitteilta tärkeät tiedot tunnistamiseen parametrien ja sijainnin avulla.



Laitteiden on siis lähetettävä viesteissä parametreina tarvittavat tiedot aina solmulta solmulle. Tiedonkeruun jälkeen pääsolmu luo yhteisen avaimen osapuolille tulevaa kommunikaatiota varten. Avaimella voi hoitaa viestien salauksen ja tiivisteen käytön. Koko yhteyden aikana on päivitettävä tunnistetietoja solmuille.

7 Vertailu

Tehokkuus ja tietoturva ovat tärkeitä ominaisuuksia koskien digitaalisia allekirjoituksia. Vaikka nämä kaksi seikkaa eivät ole suoraan toisensa poissulkevia, on syytä ottaa huomioon kummankin prioriteetti. Erityisesti mobiililaitteilla tehokkuudesta joudutaan yleensä karsimaan, joten valitaan vähemmän tehokas allekirjoitusalgoritmi. Tällöin allekirjoittaminen on hidas prosessi [SSA10].

7.1 Tietoturva

Mobiililaitteilla voidaan havaita seuraavia tietoturvariskejä: urkinta, välimieshyökkäys, datan muuntaminen, toisena osapuolena esiintyminen ja laitteen kadottaminen. Urkinnalla tarkoitetaan viestien kuuntelua, mutta se voidaan torjua helposti viestin salakirjoituksella esimerkiksi väliaikaisella istuntoavaimella. Välimieshyökkäys tarkoittaa kolmannen osapuolen asettumista lähettävän ja vastaanottavan osapuolten väliin. DH:ssa voi piileä tämä riski mutta ei yleensä RSA:ssa [SY13]. Datan muuntaminen voidaan estää salakirjoituksella sekä käyttämällä tiivistefunktioita. Toisena osapuolena tekeytyminen ja laitteen kadottaminen voidaan estää salasanan kirjoittamisella laitteelle tai visuaalisella todennuksella.

Julkisen avaimen infrastruktuuri eli PKI-malli toimii, jos salainen avain säilyy suojassa. Mikäli on pienikin riski, että salainen avain on jokun muun

tiedossa tulee avainpari vaihtaa heti. Niin kauan kun diskreetin logaritmin ongelmaa ei pystytä ratkaisemaan järkevässä ajassa, ovat RSA ja DH turvallisia protokollia. Tiivistefunktioiden tulee olla myös ajan tasalla, jotta allekirjoitusten salaus toimii. Esimerkiksi tulevaa SHA-3 standardia kehitellään paremmaksi tulevaa käyttöönottoa varten [nis14].

7.2 Tehokkuus

Suorituskyky on parantunut vuosien saatossa niin tietokoneilla kuin mobiililaitteilla. Prosessorien teknologia on kehittynyt mahdollistaen tiheämmät kellopulssit ja moniydinsuorituksen. Myös tietoliikennenopeuksien kasvamisella on ollut suuri merkitys digitaalisten allekirjoitusten luonnissa. Tehokkuutta tarvitaan nopeisiin allekirjoituksiin lyhyellä aikavälillä. Artikkelissa Self-Proxy Mobile Signature [SSA10] esitelty huutokauppasovellus tarvitsee jokaiselle huudolle uuden allekirjoituksen lyhyen ajan sisällä. Tietoturvasta on tässä tapauksessa erittäin vaikea tinkiä, joten käyttäjän olisi hyvä luoda allekirjoitus omalta laitteeltaan. Tehokkuudessa tulee ottaa huomioon siis salauksen nopeus, tiivisteen luominen ja varmenteen hankinta [SSA10]. Luonnollisesti myös palvelinpuolella esimerkiksi klusterointi on luonut mahdollisuuden tehokkaaseen allekirjoitusten/varmenteiden luomiseen monelle käyttäjälle samaan aikaan.

7.3 Nykyaikaisten menetelmien käyttö

Laitepohjaiset allekirjoitukset ovat vakiintuneet kokoajan mobiililaitteiden laskentatehon kasvun ansiosta. RSA:n lisäksi elliptiset käyrät ovat yleistyneet niiden paremman tietoturvan ansiosta suhteessa avainten pituuteen bitteinä [RB12]. AES algoritmia voidaan pitää murtumattomana, mutta DSA on murrettavissa jo muutaman bittivuodon avulla [SC12]. Elliptisen käyrän DSA:ta käytetään myös mobiililaitteilla [XDC09]. RSA:n avaimen pituuden on hyvä olla vähintään 1024 bittiä. Elliptisissä käyrissä riittää 160 bittiä tällä hetkellä [RB12].

Android-käyttöjärjestelmä tukee Javan virtuaalikonetta (JVM). Java käyttää digitaaliseen allekirjoitukseen tarvittavia protokollia, joita tarvitaan monilla mobiililaitteilla nykypäivänä. Bouncy Castle- paketti tarjoaa Javassa monenlaista kryptografisia algoritmeja tiedon salaukseen ja purkamiseen. Javalla myös satunnaisten olioiden luominen on helppoa. [SY13]

8 Yhteenveto

Tässä tekstissä olemme tarkastelleet digitaalisia todennuksia mobiiliympäristöissä ja mobiililaitteissa. Todennuksiin kuuluvat digitaalinen allekirjoitus, MAC-funktio ja symmetriseen salaukseen perustuva Diffien ja Hellmanin menetelmä. Digitaalisen allekirjoituksen ehtoina ovat vastaanottajan todennus,

datan eheys ja lähettäjän kiistämättömyys. Menetelmät allekirjoitusten luontiin vastaavat tietokoneilla samanlaisia menetelmiä. Olemme tarkastelleet julkisen avaimen infrastruktuuria, RSA:n ja Diffien ja Hellmanin menetelmää tarkemmin sekä mobiilikaupankäyntiä. Digitaalisten allekirjoitusten luonti voidaan jakaa kahteen pääryhmään: laite- ja palvelinpohjaisiin allekirjoituksiin. Laiteella allekirjoituksen voi luoda prosessori tai SIM-kortti. Lisäksi hybridimallin olemassaolo tunnetaan. Palvelinpuolella tulee korostua käyttäjän tunnistaminen ja kirjautumispalvelimen merkitys. Kiistattomuuden tulee toimia delegoinnin yhteydessä. Digitaalinen allekirjoitus voidaan delegoida palvelimelle kokonaan, osittain tai valtakirjalla. Varmenteet ja kiistattomuus luovat digitaalisen allekirjoituksen pohjan. Olemme tarkastelleet tekstin lopussa tietoturvan ja tehokkuuden merkitystä digitaalisissa allekirjoituksissa mobiiliympäristö huomioon ottaen. Nykyaikaisiin menetelmiin voimme luetella RSA:n, DSA:n, DH:n ja elliptisten käyrien algoritmit, joita muun muassa Android-käyttöjärjestelmä tukee.

Tarkista 2.10, tarkista euler ja diskreetin logaritmin etsintä,

Lähteet

[Cam03] Campbell, S.: *Supporting digital signatures in mobile environments.* Teoksessa *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003. WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on*, sivut 238–242, June 2003.

[con14] *Congruence*, huhtikuu 2014. `http://mathworld.wolfram.com/Congruence.html`.

[DH76] Diffie, W. ja Hellman, M.E.: *New directions in cryptography.* Information Theory, IEEE Transactions on, 22(6):644–654, Nov 1976, ISSN 0018-9448.

[dis14] *Discrete Logarithm*, huhtikuu 2014. `http://mathworld.wolfram.com/DiscreteLogarithm.html`.

[DKP12] Dodis, Yevgeniy, Kiltz, Eike ja Pietrzak, Krzysztof: *Message Authentication, Revisited.* Teoksessa *Proceedings of the 31st Annual International Conference on Theory and Applications of Cryptographic Techniques*, EUROCRYPT'12, sivu 361, Berlin, Heidelberg, 2012. Springer-Verlag, ISBN 978-3-642-29010-7. `http://dx.doi.org/10.1007/978-3-642-29011-4_22`.

[est14] *Facts about e-Estonia*, huhtikuu 2014. `https://www.ria.ee/facts-about-e-estonia/`.

[GMR88] Goldwasser, Shafi, Micali, Silvio ja Rivest, Ronald L.: *A Digital Signature Scheme Secure Against Adaptive Chosen-message Attacks.* SIAM J. Comput., 17(2):281–308, huhtikuu 1988, ISSN 0097-5397. `http://dx.doi.org/10.1137/0217017`.

[HZ04] He, Li Sha ja Zhang, Ning: *A New Signature Scheme: Joint-signature.* Teoksessa *Proceedings of the 2004 ACM Symposium on Applied Computing*, SAC '04, sivut 807–812, New York, NY, USA, 2004. ACM, ISBN 1-58113-812-1. `http://doi.acm.org/10.1145/967900.968066`.

[JK12] Jaiswal, C. ja Kumar, V.: *Pairwise Key Generation Scheme for Cellular Mobile Communication.* Teoksessa *Reliable Distributed Systems (SRDS), 2012 IEEE 31st Symposium on*, sivut 412–417, Oct 2012.

[LCJ04] Lei, Yu, Chen, Deren ja Jiang, Zhongding: *Generating Digital Signatures on Mobile Devices.* Teoksessa *Proceedings of the 18th International Conference on Advanced Information Networking and Applications - Volume 2*, AINA '04, sivut 532–, Washington,

DC, USA, 2004. IEEE Computer Society, ISBN 0-7695-2051-0. http://dl.acm.org/citation.cfm?id=977394.977538.

[mat14a] *Diffie-Hellman Protocol*, helmikuu 2014. http://mathworld.wolfram.com/Diffie-HellmanProtocol.html.

[mat14b] *RSA Encryption*, helmikuu 2014. http://mathworld.wolfram.com/RSAEncryption.html.

[mob14] *Mobiilivarmennus käynnistyy Suomessa*, huhtikuu 2014. http://www.mobiilivarmenne.fi/fi/bulletin/mobiilivarmennus-kaynnistyy-suomessa.

[nis14] *SHA-3 STANDARDIZATION*, helmikuu 2014. http://csrc.nist.gov/groups/ST/hash/sha-3/sha-3_standardization.html.

[pri14] *Primitive Root*, huhtikuu 2014. http://mathworld.wolfram.com/PrimitiveRoot.html.

[RB12] Ray, Sangram ja Biswas, G. P.: *An ECC Based Public Key Infrastructure Usable for Mobile Applications*. Teoksessa *Proceedings of the Second International Conference on Computational Science, Engineering and Information Technology*, CCSEIT '12, sivut 562–568, New York, NY, USA, 2012. ACM, ISBN 978-1-4503-1310-0. http://doi.acm.org/10.1145/2393216.2393310.

[rel14] *Relatively Prime*, huhtikuu 2014. http://mathworld.wolfram.com/RelativelyPrime.html.

[RSA78] Rivest, R. L., Shamir, A. ja Adleman, L.: *A Method for Obtaining Digital Signatures and Public-key Cryptosystems*. Commun. ACM, 21(2):120–126, helmikuu 1978, ISSN 0001-0782. http://doi.acm.org/10.1145/359340.359342.

[SC12] Saxena, N. ja Chaudhari, N.S.: *A Secure Approach for SMS in GSM Network*. Teoksessa *Proceedings of the CUBE International Information Technology Conference*, CUBE '12, sivut 59–64, New York, NY, USA, 2012. ACM, ISBN 978-1-4503-1185-4. http://doi.acm.org/10.1145/2381716.2381729.

[SCP12] Saxena, N., Chaudhari, N.S. ja Prajapati, G.L.: *An extended approach for SMS security using authentication functions*. Teoksessa *Industrial Electronics and Applications (ICIEA), 2012 7th IEEE Conference on*, sivut 663–668, July 2012.

[SS12] Schlöglhofer, Roland ja Sametinger, Johannes: *Secure and Usable Authentication on Mobile Devices*. Teoksessa *Proceedings of the*

10th International Conference on Advances in Mobile Computing & Multimedia, MoMM '12, sivut 257–262, New York, NY, USA, 2012. ACM, ISBN 978-1-4503-1307-0. http://doi.acm.org/10.1145/2428955.2429004.

[SSA10] Samadani, Mohammad Hasan, Shajari, Mehdi ja Ahaniha, Mohammad Mehdi: *Self-Proxy Mobile Signature: A New Client-Based Mobile Signature Model*. Teoksessa *Proceedings of the 2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops*, WAINA '10, sivut 437–442, Washington, DC, USA, 2010. IEEE Computer Society, ISBN 978-0-7695-4019-1. http://dx.doi.org/10.1109/WAINA.2010.125.

[STW09] Shu, Minglei, Tan, Chengxiang ja Wang, Haihang: *Mobile Authentication Scheme Using SMS*. Teoksessa *Services Science, Management and Engineering, 2009. SSME '09. IITA International Conference on*, sivut 161–164, July 2009.

[SY13] Schwab, David ja Yang, Li: *Entity Authentication in a Mobile-cloud Environment*. Teoksessa *Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop*, CSIIRW '13, sivut 42:1–42:4, New York, NY, USA, 2013. ACM, ISBN 978-1-4503-1687-3. http://doi.acm.org/10.1145/2459976.2460024.

[tot14] *Totient Function*, huhtikuu 2014. http://mathworld.wolfram.com/TotientFunction.html.

[TX10] Tianhuang, Chen ja Xiaoguang, Xu: *Digital signature in the application of e-commerce security*. Teoksessa *E-Health Networking, Digital Ecosystems and Technologies (EDT), 2010 International Conference on*, nide 1, sivut 366–369, April 2010.

[XDC09] Xuan, Zuguang, Du, Zhenjun ja Chen, Rong: *Comparison Research on Digital Signature Algorithms in Mobile Web Services*. Teoksessa *Management and Service Science, 2009. MASS '09. International Conference on*, sivut 1–4, Sept 2009.

[YCT09] Yu, Dingguo, Chen, Nan ja Tan, Chengxiang: *Design and Implementation of Mobile Security Access System (MSAS) Based on SSL VPN*. Teoksessa *Education Technology and Computer Science, 2009. ETCS '09. First International Workshop on*, nide 3, sivut 152–155, March 2009.