Pekka Niemi

Kannettavan laitteen muistin salaus

Tietojenkäsittelyn koulutusohjelma

2014

samk
Satakunnan ammattikorkeakoulu
Satakunta University of Applied Sciences

Kannettavan laitteen muistin salaus

Niemi, Pekka
Satakunnan ammattikorkeakoulu
Tietojenkäsittelyn koulutusohjelma
Maaliskuu 2014
Ohjaaja: Nuutinen Petri
Sivumäärä:40
Liitteitä:

Asiasanat: Salaus, Salausmenetelmät, Kryptografia, TrueCrypt, tiivisteet, symmetrinen avain, julkinen avain

---

Opinnäytetyön tarkoituksena oli käydä läpi salausmenetelmien historiaa, sekä sitä, mitä erialisia kehittyneitä salausmenetelmiä on olemassa ja miten ne toimivat.

Alussa kävin läpi, mitä tietoturva tarkoittaan, sen jälkeen kävin läpi salausmenetelmien historiaa, jonka jälkeen esittelin erilaisia salausmenetelmiä ja niiden toimintaa. Käyn myös läpi muutaman erilaisen tavan, miten salauksia yritetään purkaa.

Lopuksi esittelin muutaman tietokoneen kiintolevyn salaamiseen tarkoitetun sovelluksen, sekä otin käyttöön esimerkkisovelluksen. Esittelin myös Android laitteille suunnitellun kokolaitteen salaussovelluksen, sekä Android laitteen paikannus ja etätyhjennyssovelluksen

Memory encryption for portable devices

Niemi, Pekka
Satakunnan ammattikorkeakoulu, Satakunta University of Applied Sciences
Degreee Programme in Information and Communication Technology
March 2014
Supervisor: Nuutinen, Petri
Number of pages:40
Appendices:

Keywords: Encryption, Encryption standards, Cryptography, TrueCrypt, symmetric key, asymmetric key

---

The purpose of this thesis was to demonstrate the history of the cryptography and what kind of encryption algorithms we have and how do they work.

First in this thesis I presented what does the information security mean and after that I looked at history of encryption methods. Then I presented different sort of encryption algorithms and show how they work. I also tell you a couple of ways to break encryptions.

In the end of this thesis I presented couple of applications for computer hard-disk encryption and initialization of one example program.
I also presented a full-disk encryption application to android devices. Lastly I looked at positioning and remote wipe application for android devices.

SISÄLLYS

1 JOHDANTO

Nykyään ihmiset liikkuvat työssään paljon ja heillä on mukana kannettavia laitteita, joilla he hoitavat työ ja yksityisasioita. Kannettavat tietokoneet ja mobiililaitteet ovat pienehköjä ja kun ollaan paljon liikkeellä riski laitteen hukkaamisesta tai varkauden kohteeksi joutumisesta on suuri. Kun laitteilla on hoidettu työasioita, niiden mukana saattaa olla paljon kallista ja ehkä arkaluonteistakin tietoa, joka vääriin käsiin joutuessaan voi aiheuttaa taloudellista vahinkoa ja ei kukaan pidä siitä, että omat yksityiset asiat joutuisivat vieraan tahon käsiin. Kannettavien laitteiden muistien salaus on hyvä ja helppo tapa estää tietojen vääriin käsiin joutuminen.

Idean tähän työhön sain, kun tutkiskelin mahdollisuuksia erilaisiin opinnäytetöihin. Tarkoitukseni oli ensin toteuttaa tietoturvaopas pk-yrityksille, mutta päädyin kuitenkin salausmenetelmistä kertovaan työhön.

Tässä työssä esittelen salausmenetelmien historiaa ja salausmenetelmiä. Esittelen tunnetuimmat symmetriset salausalgoritmit DES, AES, SERPENT ja TWOFISH. Julkisen avaimen algoritmit RSA ja DIFFIE HELMAN, sekä tiiviste eli HASH algoritmit MD5 ja SHA1. Esittelen myös muutaman yleisimmistä salausten murtamiskeinoista.

Työn toiminnallisen osuuden ensimmäisessä osassa esittelen yleisimpiä Windows käyttöjärjestelmien kiintolevyjen salaukseen tarkoitettuja ilmais-sovelluksia Microsoft Bitlockeria, TrueCryptia, sekä Secustar DriveCryptia, sekä toteutan yhden esimerkkisovelluksen käyttöönoton, joka tässä tapauksessa on TrueCrypt. Käyttöönoton toteutan virtuaalityöasemalle, joka sijaitsee koulumme palvelimella.

Toisessa toiminnallisessa osuudessa toteutan Android käyttöjärjestelmällä varustetun mobiililaitteen muistin salauksen, sekä käyttöönotan sovelluksen, jolla varastettu tai kadonnut android laite voidaan paikantaa ja etätyhjentää.

2 TIETOTURVA

Tietoturva on käsitteenä hyvin laaja ja siitä kirjoitetuissa teoksissa tietoturvaa käsitellään hieman eri näkökulmista. Kuitenkin tietoturvan perus-ajatus on tiedon turvaaminen niin, että tieto on luottamuksellista, eheää, sekä käytettävissä kun sitä tarvitaan. Nykyään on myös laillisia vaatimuksia tietoturvalle, esimerkiksi sähköisessä kaupankäynnissä pitää pystyä todentamaan, kukan on tehnyt mitä ja milloin

Tiedon luottamuksellisuudella tarkoitetaan tietoturvassa sitä, että tieto on niiden henkilöiden saatavilla, kenelle se on tarkoitettu, eikä ulkopuolinen henkilö saa sitä tietoonsa, se on tietoturvassa asia johon yleensä panostetaan eniten. Siihen pystytään vaikuttamaan suojaamalla tieto salasanoilla sekä käyttäjätunnuksilla. Myös tieto, joka ei ole sähköisessä muodossa tulee suojata, esimerkiksi tulosteet joissa on arkaluontoista tietoa pitää säilyttää asiaankuuluvasti, sekä hävittää oikea-oppisesti.
(Hakala & Vainio 2006, 4)

Eheydellä tarkoitetaan sitä että tietoa ei ole tahallisesti tai tahattomasti muutettu. Tiedon eheys voidaan menettää hyökkäyksen seurauksena tai inhimillisestä syystä. Eheyteen pystytään vaikuttamaan suunnittelemalla tietoturvallisuutta, sekä suunnittelemalla ja pitämällä palomuurit oikein.

Tiedon saatavuuteen pystytään vaikuttamaan pitämällä tiedon käsittelyyn vaikuttavat laitteet ajan tasalla niin että ne vastaavat käyttäjämääriä, sekä ovat yhteensopivia keskenään. Myös varmuuskopiointi on osa käytettävyyden ylläpitoa. On myös tärkeää, että henkilöt jotka vastaavat tietoturvasta pystyvät tietoturvapalveluita tarvittaessa käyttämään (Krutz & dean vines, 60)

Pääsynvalvonta on myös osa tietoturvaa. Pääsynvalvonnalla tarkoitetaan sitä, että käyttäjä tai ohjelma pystyy käyttämään sille kuuluvia tietoja. Tietojen käyttö voi olla eri-asteista. Eri tiedot voivat olla eri käyttäjillä käytettävissä eri tavalla, toista tietoa voi vain lukea ja toista kirjoittaa, kun taas joitakin tietoja pitää pystyä lukemaan, että kirjoittamaan.

Kiistämöttömyydellä tarkoitetaan sitä kun tietojärjestelmässä tehdään jotakin niin siitä jää lokiin jälki. Esimerkiksi potilastietojärjestelmässä pitää näkyä, ketä on katsonut ja kenen mahdollisesti arkaluonteisia tietoja.

Ehkä yksi aliarvioiduimmista tietoturvan periaatteista on todennus. Todennuksella tarkoitetaan sitä, että henkilö on todellakin henkilö kuka hän väittää olevansa. Joku voi kiusantekomielessä luoda esimerkiksi sähköpostiosoitteen jonkun toisen henkilön nimiin ja ruveta tämän toisen henkilön nimissä lähettelemään asiattomia viestejä ja viestien vastaanottaja pitää viestejä aitoina. Muista tietoturvaperiaatteista ei ole hyötyä, jos todennusta ei ole hoidettu kunnolla. (Järvinen 2003b, 31–33)

3 SALAUKSEN HISTORIA

Viestien salauksesta puhuttaessa tarkoitetaan yleensä kryptografiaa. Kryptografia tulee kreikan kielen sanoista kryptó=piilo ja gráfo=kirjoitus. (http://www.logicalsecurity.com/resources/whitepapers/Cryptography.pdf).

Kryptografiaa tutkimaan on syntynyt myös oma tieteenlajinsa kryptologia, viimeaikoina tosin kryptologia on käsitetty väärin, niin että se käsittäisi tietoturvatieteen kokonaisuutena, näin ei kuitenkaan ole. Kryptologia tieteenalana sisältää salaukseen liittyvien koodien, sekä algoritmien tutkimista, sekä koodien kehittämistä eli kryptografiaa ja purkamista eli kryptoanalyysia. (Kerttula, 65)

3.1 Salaaminen ennen kehittyneitä menetelmiä

Spartalaiset salasivat viestejänsä eräänlaisella sauvalla (kuvassa 1.), lähettäjä ja vastaanottaja omistivat mitoiltaan tarkasti samanlaiset puusauvat. Lähettäjä kiersi sauvan ympäri nahkanauhaa ja kirjoitti viestin siihen, sen jälkeen hän lähetti nauhan vastaanottajalle, joka kieritti nauhan omaan sauvaansa ja viesti näkyi, ilman sauvaa nauhassa oli vain sekava määrä erilaisia kirjaimia.(http://visual.ly/history-encryption)

Kuva 1. Spartalaisten käyttämä scytale (http://www.elkriver.k12.mn.us/webpages/sbraun/research.cfm?subpage=26862)

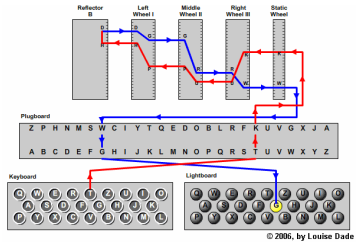
Myöskään Julius Caesar ei vallassa ollessaan pitänyt viestinviejiään luotettavina, kun lähetti kenraaleilleen viestejä. Ainoastaan kenraalit, jotka tiesivät avaimen joka siinä tapauksessa oli ”shift by 3” saivat luettua viestin.(http://fisher.osu.edu/~muhanna.1/pdf/crypto.pdf). Viestien salauksessa periaatteena voidaan pitää sitä, että vain viestin haluttu vastaanottaja voi avata viestin.

George Washingtonin sihteeri Thomas Jefferson kehitti vuonna 1797 ”Jeffersons Wheel” laitteen, joka koostui pyörivistä puulevyistä metalliakselin ympärillä. Jokainen levy oli jaettu 26 osaan. Haluttu viesti pyöritettiin levyistä ja valittiin muualta rullasta joku satunainen rivi ja käytettiin sitä viestin salaamiseen. Vastaanottajan pyörittäessä omasta rullastansa levyt salakirjoituksen antamaan asentoon hän sai selville alkuperäisen viestin.(http://visual.ly/history-encryption)

3.2 1900-luvun alusta nykyaikaan

Ehkä kaikista tunnetuin salaukseen liittyvä laite on saksalaisen insinöörin Arthur Scherbiuksen kehittämä Enigma, josta saksan armeija kehitti oman versionsa, jota se käytti 2.maailmansodassa.(http://www.bbc.co.uk/history/topics/enigma) Enigmassa oli näppäimistö ja kun näppäintä painoi, niin painaluksesta syntyi sähköinen signaali (kuvassa2.), joka kulki staattiselle roottorille, josta se jatkoi matkaa kolmen roottorin läpi. Jokainen kolmesta roottorista muutti kirjainta toiseksi kirjaimeksi. Sen jälkeen signaali palautettiin roottoreiden läpi lamppunäytölle, jossa syttyi lamppu kryptatun kirjaimen kohdalle. Vastaanottaja sai viestin auki, kun laite oli samoissa asetuksissa lähettäjän laitteen kanssa, asetukset muuttuivat joka päivä. (http://enigma.louisedade.co.uk/howitworks.html). 3-roottorisella Enigmalla pystyttiin koodaamaan viestejä 10 000 000 000 000 000 eri tavalla, mutta myöhemmin so-

dan aikana roottorien määrä lisättiin viiteen ja kaikista salaisimmat viestit lähetettiin vieläkin kehittyneimmillä Enigmoilla. (Järvinen 2003b,)



Kuva 2. Enigman toiminta (http://enigma.louisedade.co.uk/howitworks.html)

Salausmenetelmät olivat aikaisemmin pitkälti sodankäynnin viestinnän turvaamiseen kehitettyjä, mutta 1970 luvulla asia alkoi muuttua. Pankit alkoivat sähköistää rahaliikennettä ja tätä sähköistä rahaliikennettä piti turvata. (Järvinen 2003b, 24)

3.3 Esimerkkejä tiedon salaamisesta nykypäivänä

Nykyään salauksia käytetään moniin eri käyttötarkoitukseen ja niihin voi törmätä lähes kaikkialla. Salauksia ei nykyään enää hoideta käsin, vaan salaamisen hoitavat erilaiset sovellukset ja ohjelmat, jotka käyttävät salaamiseen erilaisia salaus-algoritmeja ja hoitavat salaus-avaimien hallinnan.

3.3.1 SSL ja SHTTP

SSL (Secure sockets layer) on Netscapen vuonna 1994 kehittämä suojausmenetelmä, joka on kehitetty suojaamaan internetissä asiakkaan ja palvelimen välistä liikennettä. Siitä on tullut yksi yleisimmistä ja tärkeimmistä tietoturvaprotokollista. SSL tukee RSA- algoritmeja, IDEA, DES ja 3DES algoritmeja ja MD5 tiivistealgoritmia. SSL on sovelluksista riippumaton protokolla, koska se toimii TCP/IP protokollan ja sovellusten välissä.

Otettaessa SSL-työasemalta yhteyttä palvelimeen koneet neuvottelevat suojauspaketin. Aluksi koneet yrittävät ottaa käyttöön vahvimman salausmenetelmän, joka löytyy kummaltakin koneelta. Jos yhteyttä ottava työasema käyttää 40-bitin istuntoavainta

käyttävää riisuttua selainta, niin koneet neuvottelevat salausavaimensa 40-bitin avaimiksi. (Krutz& Dean Vines 2003, 170-173, Kerttula 1998, 297-298)

S-HTTP (Secure Hypertext Transfer Protocol) on SSL:lle kehitetty vaihtoehto, jolla voidaan suojata web-yhteyksiä, se on erityisesti elektronisen kaupankäynnin ja verkkopankkiasioinnin turvaamiseksi kehitetty. S-HHTP:llä voidaan suojata liikenne yksittäisille web-sivuille toisin kuin SSL:llä , joka suojaa koko istunnon. (Krutz&Dean Vines 2003, 170-173)

3.3.2 IPSEC

IPSEC on joukko protokollia, joiden avulla mahdollistetaan TCP/IP- yhteyksien suojaaminen kryptografisesti verkkokerroksella. IPSEC:llä on tavoitteena suojata luotettujen tietokoneiden välistä liikennettä väärennyksiltä ja salakuuntelulta, antaa suojaa käytössä oleville internet viestintäohjelmistoille, suojata liikennettä jo valmiiksi rakennetussa epäluotettavassa verkossa

IPSEC määrittelee IP-paketille kaksi paketti-otsikkoa. Otsikot sisältävät SPI parametrin (security parameter index), joka on numeroarvo. Isäntäkone käyttää SPI-parametria kryptoavainten, sekä niiden käytön tunnistukseen.

Otsikot ovat autentikointiotsikko (autehntication header, AH), joka sisältää informaatiota, josta voidaan tarkistaa paketin eheys ja saada selville onko se mahdollisesti muuttunut matkan aikana. Toinen otsikko on koteloitu salattu data (Encapsuling Security Payload, ESP), joka hoitaa varsinaisen salauksen ja salaa paketin loppuosan sisällön. ESP otsikon formaatti vaihtelee riippuen käytettävästä salausalgoritmista. Käytettävä salausavain valitaan SP1 parametrin avulla.(Kerttula 1998, 198-220)

3.3.3 Secure Shell

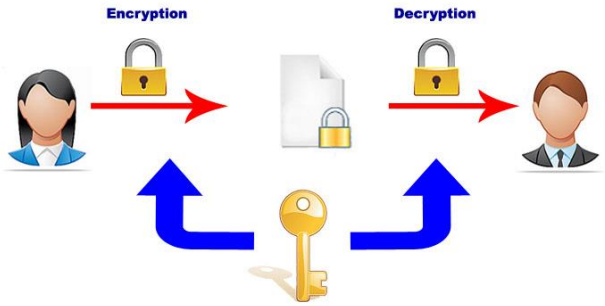
Secure Shell eli SSH on ohjelma, jota pystytään käyttämään kun halutaan luoda salattu tiedonsiirtopolku kahden koneen välille. SSH1 otettiin käyttöön vuonna 1995. Vuonna 1997 IETF julkaisi uudemman version SSH2, jossa parannettiin SSH1:n toiminnallisuutta ja tietoturvaominaisuuksia.

Yksinkertaistettuna SSH muodostaa TCP/IP yhteyden isäntäkoneeseen ja todennus suoritetaan käyttäjätunnuksella ja salasanalla, jonka jälkeen SSH aloittaa tiedon salauksen. Nykyisessä SSH2 versiossa voidaan käyttää seuraavia salaus-menetelmiä: 3DES, BLOWFISH, TWOFISH, ARCFOUR sekä CAST-128-CBC.(Thomas 2005, 145-150)

4 SALAUSALGORITMIT JA NIIDEN MURTAMINEN

4.1 Symmetriset salausalgoritmit

Symmetrisistä salaus-algoritmeista (Symmetric key algorithms) puhuttaessa tarkoitetaan salausmenetelmiä, joissa viestin lähettäjä sekä vastaanottaja käyttävät viestin salaamiseen ja purkuun samaa avainta (kuvassa 3.). Symmetrisiä salausmenetelmiä kutsutaan myös salaisen avaimen salausmenetelmiksi tai klassisiksi salausmenetelmiksi. Julkisen avaimen ns. Epäsymmetriset salausjärjestelmät ilmestyivät vasta 1970-luvulla.(Kerttula, 74)



Kuva 3. Symmetrinen salaus (http://en.kryptotel.net/encryption.html)

Symmetriset salausmenetelmät perustuvat avaimen salaisuuteen. Avaimen valintaan on kiinnitettävä huomiota sillä niin kauan kun avain on heikko, ei tehokkaimmastakaan algoritmista saada irti kaikkea hyötyä.. Käyttäjän ei tulisi itse pystyä luomaan avainta, koska oletettavasti käyttäjä ei pystyisi luomaan tarpeeksi monimutkaista avainta, toisin kuin algoritmin satunaislukugeneraattori. Avaimen kuljettamiseen osapuolten välillä on myös kiinnitettävä huomiota, siihen on kehitetty erilaisia avaimenvaihtoprotokollia.(Kerttula1998, 140)

4.1.1 DES

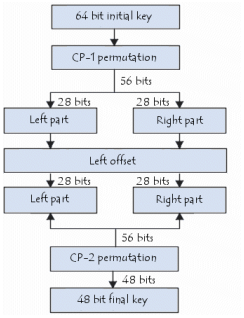
DES (Data Encryption Standard) on IBM:n kehittämä salausmenetelmä. Se sai alkunsa 1960-luvull kun IBM:llä tutkijana työskennellyt Horst Feistel kehitti oman nimensä saaneen Feistelin salaimen. Feistel kehitteli salaimestaan useita versioitaan ja vuonna 1971 valmistui lucifer niminen salain.

IBM alkoi kehittää pankeille omaa 64-bittistä salainta turvaamaan sähköistä rahaliikennettä. Tähän IBM pyysi apua NSA:lta, koska epäonnistua ei saanut. Tämä salain kulki työnimellä DSD. DSD:n pituutta lyhennettiin 56-bittiin kehittämisen aikana NSA:n pyynnöstä. DES julkaistiin vuonna 1975 vuonna 1977 siitä tuli virallinen salaus-standardi NIST 46.(Järvinen 2003b, 87-88)

DES sisältää 16-kierrosta ja se voidaan jakaa kahteen osaan. Ensimmäisessä osassa algoritmiin syötetään 64-bittinen salattava lohko, sekä 64-bittinen salaus-avain. En-

simmäisellä tasolla algoritmi järjestää bitit uudestaan eli permutoi, tällä ei ole varsinaisen salauksen kannalta tosin ole merkitystä.

DES:lle syötetystä 64-bittisestä salaus-avaimesta poistetaan pariteettibitit ja jäljelle jää 56 käytettävää bittiä. Nämä jaetaan kahteen osaan L ja R osaan (kuvassa 4), joissa kummassakin on 28-bittiä. Tämän jälkeen bittejä siirretään yhdellä tai kahdella bitillä vasempaan ja vasemmalta ylitulevat bitit siirretään oikealle. Tämän jälkeen L ja R-lohkot yhdistetään taas yhdeksi 56-bittiseksi lohkoksi, josta muodostetaan 48-bittinen avain.(http://en.kioskea.net/contents/134-introduction-to-encryption-with-des)



Kuva 4. DES avaimen muodostaminen (http://en.kioskea.net/contents/134-introduction-to-encryption-with-des)

Toisella tasolla 64-bittinen selväkielinen lohko jaetaan kahteen osaan, jotka ovat L eli vasen-lohko ja R eli oikea-lohko. R-lohkon bitit ja kierroksen oma salaus-avain syötetään funktiolle, jonka jälkeen tulokselle tehdään XOR L-lohkon kanssa. Tätä syntynyttä tulosta käytetään seuraavalla kierroksella R-lohkona ja vastaavasti taas R-lohkoa käytetään L-lohkona. Kun tämä on tehty 16-kertaa L ja R lohko yhdistetään ja yksi kokonainen lohko on käsiteltynä.
(http://www.youtube.com/watch?v=UgFoqxKY7cY, Järvinen 2003b, 87-90)

Funktio koostuu kolmesta osasta E eli laajennos, S-laatikko ja permutointi. Laajennos pidentää R-lohkosta tulevat 32 bittiä 48-bittiin käyttäen eräitä lohkon bittejä kahdesti. Tälle 48-bittiselle lohkolle suoritetaan XOR operaatio kierrosavaimen kanssa.

Laajennoksesta saatu tulos tuodaan S-laatikoille, joita on kahdeksan kappaletta S1,S2,S3,S4,S5,S6,S7 ja S8. Jokaiseen laatikkoon viedään 6-bittiä ja laatikot muut-

tavat bittien järjestyksen. S-laatikoiden jälkeen jäljelle jää 32-bittinen tulos joka viimeisenä taas permutoidaan, eli järjestetään uudestaan. Nyt algoritmi on suoritettu. (http://en.kioskea.net/contents/134-introduction-to-encryption-with-des)

Tietokoneiden tehojen lisääntyessä voidaan monen eri koneen yhteistytöllä DES murtaa, tähän on kehitetty ratkaisu, 3DES. 3DES salaa viestin kolmeen kertaan ja sen avaimen pituus on 168-bittiä, se on siis melko varma salausmenetelmä. Kuitenkin avaimen pituus tuo mukanaan myös heikkouden, 3DES on nimittäin kolme kertaa hitaampi kuin tavallinen DES. (Krutz&Vine 2003s, 151-152)

4.1.2 AES

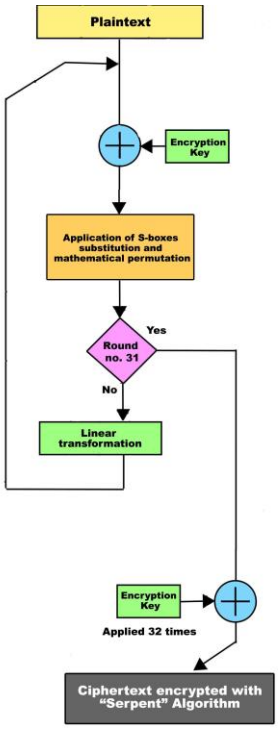
Kun DES ei enää ollut tarpeeksi turvallinen salausmenetelmä NIST (National institute of standards and techology) järjesti kilpailun algoritmeista, joista voisi tulla DES:n seuraaja eli AES. Sopivia ehdokkaita uudeksi salaimeksi tuli viisitoista, joista viisi pääsi finaaliin. Finalistit olivat MARS, RC6, Rijandel, Serpent ja Twofish, näistä voittajaksi 2.10.2000 NIST julisti Rijandel salalaimen, jonka olivat kehittäneet belgialaiset tutkijat Joan Daemen ja Vincent Rijmen (Krutz&Dean Vines 2003, 152; Järvinen 2003b, 96)

Rijndael salain on lohkosalain, jonka lohkot ja avaimet voivat olla eripituisia. Standardoituja pituuksia ovat 128,192, ja 256 bittiä ja kierrosten lukumäärä voi olla10,12 tai 14. Rijandaelissa voi olla 128-bittisiä avaimia$3{,}4*10^{38}$, 192-bittisiä $6{,}2*10^{57}$ ja 256-bittisiä $1{,}1*10^{77}$. Tämä tarkoittaa sitä, että tietokoneella, joka saa DES salauksen murrettua yhden sekunnin aikana, Rijandael salauksen murtamiseen kuluisi 149 driljoonaa vuotta. Rijandael voidaan jakaa kolmeen kerrokseen, joissa tapahtuu kierrosmuunnos. (Krutz&Dean Vines 2003, 152)

4.1.3 SERPENT

Myöskin AES finaaliin päässyt Serpent salain on käyttökelpoinen. Serpent-salaimen kehittivivät Rossa Andersson, Eli Biham ja Lars Knudsen. Serpent on todella turvallinen salain, mutta kilpailun voittanut Rijandael salain on Serpent salainta kolme kertaa nopeampi.(http://www.truecrypt.org/docs/serpent#Y113, Järvinen 2003b, 97)

Serpent salain käyttää 256-bittistä avainta ja 128-bittisiä lohkoja ja toimii XTS mallella. Serpent salain käyttää myös DES salaimesta tuttuja S-bokseja, jokaiseen boksiin tulee 4-bittiä ja lähtee 4 bittiä. Serpent salauksessa on 32 kierrosta(kuvassa 6) (http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.86.2107&rep=rep1&type=pdf)



Kuva 5. Serpent salauksen toiminta(http://en.kryptotel.net/serpent.html)

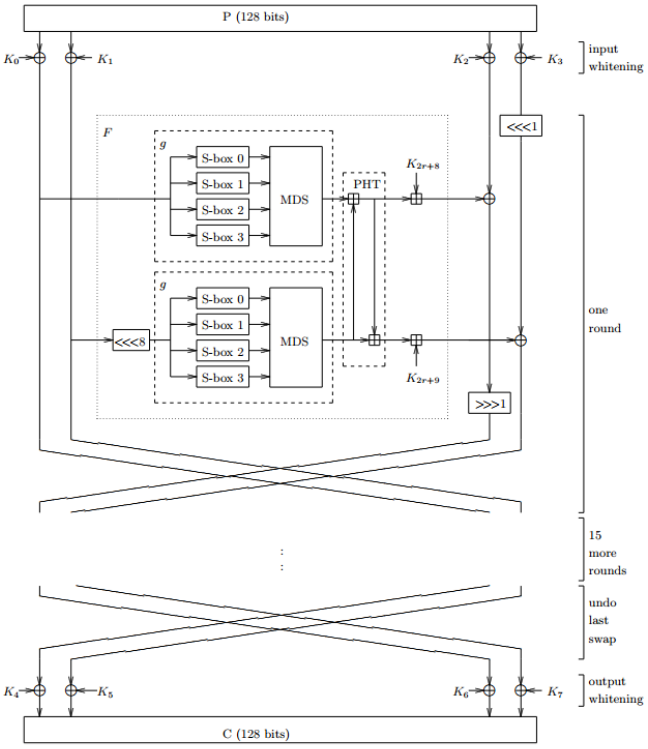
4.1.4 TWOFISH

Twofish joka myöskin AES kilpailussa menestyi on symmetrinen lohkosalain. Siinä on 16 kierrosta ja se käsittelee 128-bittisiä lohkoja, twofishin avainpituus voi olla korkeintaan 256-bittiä.(Kruz&Dean Vines 2003, 154)

Toimintaperiaatteeltaan Twofish perustuu Feistelin verkkoon, samoin kuin DES. Ensin selväteksti jaetaan neljään 32-bittiseen osaan, jonka jälkeen niille tehdään xor funktio. 32-bittiset osat jaetaan oikeaan ja vasempaan lohkoon. G-funktio jakaa 32-bittiset osat edelleen neljän tavun osiin.

Sen jälkeen osat jaetaan neljään avainriippuvaiseen S-boksiin ja S-boksit muuttavat tavut. Tämän jälkeen tavut yhdistetään MDS matriisiin, josta saadaan kaksi 32-bittistä osaa. Osat yhdistetään PHT:lla (Pseudo-Hadamard Transform) ja osiin lisätään kaksi kierrosavainta.

Osat XOR:ataan yhteen oikeanpuoleiseen lohkoon. Seuraavalla kierroksella taas oikeanpuoleinen ja vasemmanpuoleinen lohko vaihtavat paikkaansa. Ennen XOR operaatiota tapahtuu myös yhdenbitin siirto vasempaan jokaisella kierroksella. Kuvassa 8. twofishin toimintaperiaate. (https://www.schneier.com/paper-twofish-paper.pdf)



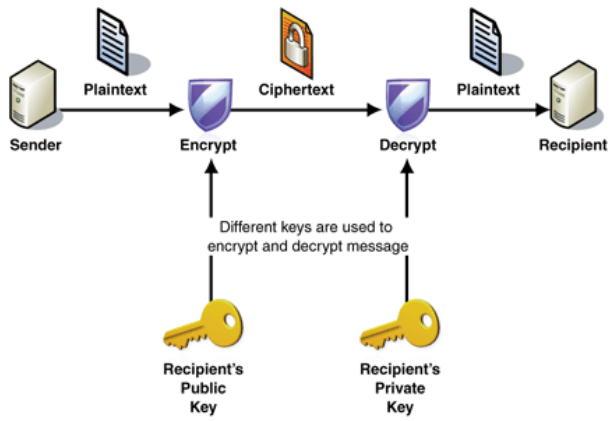
Kuva 6. Twofish toiminta (http://ilmukripto.wordpress.com/2009/12/16/algoritma-block-cipher-twofish-finalis-aes-candidate/)

4.2 Julkisen avaimen salausalgoritmit

Julkisen avaimen kryptgorafia sai alkunsa W.Diffien ja M-E Hellmanin työstä ”New directions of cryptography”. Työssä näytettiin, että salaisia viestejä voidaan vaihtaa ilman salaisen avaimen toimitusta. Julkisen avaimen menetelmiä käytetään pääasiassa avainten hallintaan, autentikointiin ja salaamiseen.

Julkisen avaimen salaus-algoritmeissa (public key algorithms) viestin lähettäjä käyttää salaamiseen eri avainta, kuin viestin vastaanottaja salauksen purkamiseen (kuvassa 8). (Järvinen 2003a, 1999, Kerttula 1998, 155-156)

Julkisen avaimen algoritmit eivät vastoin yleistä käsitystä ole aina turvallisempia, kuin salaisen avaimen algoritmit. Esimerkiksi salaisen avaimen IDEA salaimessa on pitkä avain ja se voi olla turvallisempi kuin julkisen avaimen RSA salain. Julkisen avaimen algoritmit ovat myös hitaampia, kuin salaisen avaimen algoritmit, johtuen runsaasta laskennasta, siksi niitä ei voida käyttää joka paikassa. Tämän takia on kehitetty hypridi-järjestelmiä jolla lähetetään sanoman avaamiseen tarvittava salainen avain julkisen avaimen salauksella. (Kerttula 1998, 155-156)



Kuva 7. Julkisen avaimen toiminta (http://msdn.microsoft.com/en-us/library/ff647097.aspx)

4.2.1 RSA

R. Rivestin, A.Shamirin ja L. Adlemanin vuonna 1978 julkaisema RSA on salaukseen, digitaaliseen allekirjoittamiseen, sekä jakeluun tarkoitettava julkisen avaimen salausmenetelmä, josta on tullut yksi tärkeimmistä salausmenetelmistä. (Kerttula 1998, 173)

RSA menetelmän toiminta perustuu alkulukuihin, niin kuin moni muukin salausmenetelmän toiminta. RSA menetelmässä tarvitaan kaksi isoa ja suunnilleen yhtä pitkää alkulukua, joita merkitään kirjaimilla p ja q. Tarvitaan myös kaksi lukua joiden potenssit osoittavat salatekstiä ja selväkielistä tekstiä, lukuja merkitään kirjaimilla e ja d.

Ensin lasketaan p*q, jonka jälkeen t=(p-1)*(q-1) valitaan e, täyttäen ehdon s.y.t(t,e)=1 ja 1<e<t. Tämän jälkeen lasketaan $d=e^{-1}$ mod t. Luvuista e ja modulus d muodostuu julkinen avain ja luvuista d ja n yksityinen avain. Muut luvut on pidettävä salassa ja ne myös tuhotaan avainten luomisen jäleekn.

Salaus tehdään korottamalla selväkielinen teksti P potenssiin salauseksponentti e josta otetaan modulo n ($C=P^e$ mod n). Purku taas toteutetaan korottamalla C:n potenssiin d ja ottamalla siitä mod n($P=C^d$ mod n)(Kerttula 1998, 141-142)

4.2.2 DIFFIE-HELLMAN

Diffie-Hellman algoritmi on ensimmäinen julkiseen avaimeen perustuva algoritmi ja se tarkoitettu symmetristen avainten jakeluun. Sen esittelivät vuonna 1976 Whitfield Diffie ja Martin Hellman. Algoritmia käytettäessa kummallakin osapuolella on yksityinen avain, josta diffie hellman luo julkisen avaimen, siten että yksityistä avainta ei voida päätellä julkisesta avaimesta (Thomas 2005, 257)

Ensin osapuolista toinen valitsee kaksi vähintään 150numeroista kokonaislukua ja ilmoittaa niistä toiselle osapuolelle, jonka jälkeen osapuolet valitsevat jälleen suuret

salaikseksi jäävät luvut satunaisesti. Salainen luku, sekä aikaisemmin valitut luvut syötetään funktioon jonka jälkeen osapuolet vaihtavat tulokset, jonka jälkeen tulokse syötetään jälleen funktioon. Osapuolila on operaation jälkeen identtiset avaimet, joita he voivat käyttää salaisina avaimina (Kerttula 1998, 169)

4.3 Tiivisteet (hash)

Tiivisteistä puhuttaessa tarkoitetaan tiivistefunktiota, jolle annetaan suuri määrä bittejä jonka jälkeen tiivistefunktio laskee biteistä tiiviste-arvon. Tiivistefunktiota käytettäessä alkuperäistä sanomaa ei voida tiivisteestä palauttaa. (Järvinen 2003b,122-124)

Tiivistefunktioita käytetään digitaaliseen allekirjoitukseen. Sen avulla pystytään varmistamaan allekirjoittajan henkilöllisyys ja onko esimerkiksi sanoma muuttunut ajan kuluessa.

Kun käytetään digitaalista allekirjoitusta, silloin lähetettävästä tiedostosta lasketaan tiiviste tiivistefunktiolla. Tiiviste on uniikki ja kahta samanlaista tiivistearvoa ei ole olemassa. Lähettäjä salaa tiivisteen yksityisellä avaimellaan jonka jälkeen tiiviste lähetetään alkuperäiseen tiedostoon liitettynä vastaanottajalle. Vastaanottaja avaa tiivisteen lähettäjän julkisella avaimella. Vastaanottajan pystyessä avaamaan tiivisteen lähettäjän todellisella julkisella avaimella on lähettäjän henkilöllisyys varma.

Kun vastaanottaja on saanut tiedoston perille hän voi laskea tiedostosta tiivisteen käyttämällä samaa tiivistefunktiota kun lähettäjä. Tiedosto ei ole matkan aikana muuttunut, jos tiivisteearvo on sama kun lähettäjän saama tiiviste arvo. (Kruz & Dean Vines 2003, 159-160)

4.3.1 MD5

MD5(Message digest 5) on Ron Rivestin kehittämä tiivistealgoritmi, jonka hän kehitti vuonna 1991. MD5 on yleisin tiivistealgoritmi ja sitä käytetään yleisesti kaikissa salausta vaativissa kohteissa sähköpostista elektroniseen rahaliikenteeseen. MD5 on myös RFC 1321 standartoitu. (Kerttula 1998, 142)

Ensimmäisessä vaiheessa algoritmiin syötetään viesti, josta halutaan tiivisten. Viestin pituudella ei ole merkitystä. Tämän jälkeen algoritmi laajentaa viestin niin, että viestin pituus bitteinä on 448 modulo 512. Eli viestin pituudeksi tulee 64 bittiä. Viestin laajennoksessa ”1” bitit liitetään itse viestiin ja ”0” bitit ovat liitteitä.

Kolmannessa algoritmin vaiheessa 64-bittiseen(alkuperäisestä laajennettuun) viestiin lisätään toisen vaiheen tulos. Nyt viestin pituus täytebitteineen on täsmälleen 512 bittiä. Tämä siis sisältää 16 32-bittistä tavua. Neljännessä vaiheessa luodaan tiiviste neljän puskurin avulla. Jokainen neljästä puskurista ottaa vastaan kolme 32-bittistä sanaa ja tuottaa niistä 32-bittisen sanan. (http://tools.ietf.org/html/rfc1321)

MD5 algoritmia käytetään digitaalisiin allekirjoituksiin niin kuin muitakin tiivistealgoritmeja. Algoritmilla tiivistettyä tietoa ei voida myöskään palauttaa alkuperäiseen muotoon. (Anonymous 2002, 396)

4.3.2 SHA1

SHA1(Secure hash algorithm) on MD5 ohella yleisimpiä tiivistealgoritmeja. Se on NIST:in ja yhdysvaltojen kehittämä. Yhdysvaltojen hallinto käyttää SHA1 algoritmia digitaalisissa allekirjoituksissa, koska se on turvallisempi kuin MD5. SHA1 algoritmin lisänumero yksi johtuu lisätystä 1-bitin rotaatiosta, jota alkuperäisessä SHA algoritmissa ei ollut.

Toiminnaltaan SHA1 algoritmi vastaa MD5 algoritmia, mutta erojakin löytyy. SHA tuottaa 160-bittisen tiivisteen, kun MD5 tuottaa 128-bittisen, SHA1 algoritmi on siis hieman hitaampi kuin MD5. (Kerttula 1998, 143-14)

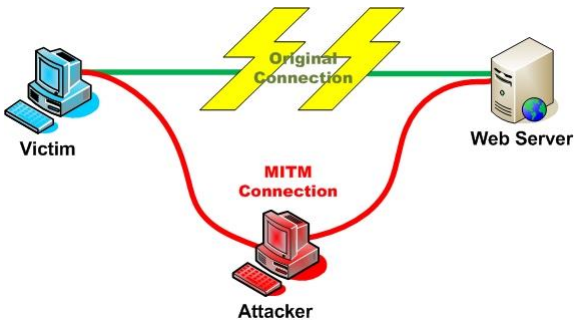
4.4 Salausten murtaminen

Siitä lähtien, kun viestejä ja tietoja on salattu, on myös salauksia yritetty purkaa. Salausten purkamiseen saattaa löytyä erilaisia intressejä, jotkut tahot tavoittelevat salausten purkamisella taloudellista hyötyä, jotkut taas yrittävät purkaa salauksia kiusantekomielessä.

Hyökkäys ei välttämättä kohdistu itse salausmenetelmään, vaan se voi kohdistua salauksen osapuoliin salaus-avaimen käsiin saamiseksi. Seuraavaksi esittelen yleisimpiä hyökkäystapoja. On siis olemassa monia muitakin hyökkäystapoja, kuin seuraavaksi esitellyt.

4.4.1 “Mies keskellä”

Mies keskellä hyökkäystavassa hyökkääjä on salauksen osapuolien välissä (Kuva 8.). Mies keskellä hyökkäystä voidaan käyttää heikosti suunniteltuja julkisen avaimen järjestelmiä vastaan.



Kuva 8. Mies keskellä hyökkäys (https://www.owasp.org/index.php/Man-in-the-middle_attack)

Mies keskellä hyökkäyksessä hyökkääjä asettuu osapuolien välille, kun he ovat vaihtamassa julkisia avaimiaan. Kun hyökkääjä saa 1.osapuolen julkisen avaimen hal-

tuunsa hän tekeytyy toiseksi 1.osapuoleksi, ja luo uuden avaimen, jonka lähettää 2.osapuolelle ja 2.osapuoli olettaa, että avain on aito. Hyökkääjä toimii näin myös toisin päin eli lähettää 2.osapuolen nimissä uuden avaimen 1.osapuolelle.

1.osapuoli salaa lähettämänsä viestinsä julkisella avaimellaan. Myös hyökkääjällä on tiedossa kyseinen avain ja hän pystyy lukemaan viestin avaimeen liittyvällä salaisella avaimella. Hyökkääjä siis on osapuolten välissä ja kun toinen osapuoli lähettää viestin toiselle hyökkääjä lukee viestin välissä ja lähettää sen edelleen. Niin kauan kun hyökkääjän onnistuu saada haltuunsa kaikki osapuolten viestit, hänestä ei tiedetä mitään. (http://www.netlab.tkk.fi/opetus/s38118/s98/htyo/39/hyokkaykset.shtml)

4.4.2 Kryptoanalyysit

Differentiaalisessa kryptoanalyysissa pyritään selvittämään salaimen toiminta. Siinä lasketaan kahdesta selvätekstistä XOR-erotus, jonka jälkeen sen etenemistä seurataan salaimessa f-funktiosta toiseen ja verrataan vastaavista selvätekstipareista saatuun XORiin, jos bittiyhdistelmä kuvautuu tietyllä avaimen osalla tarpeeksi niin tällöin analysoija voi laskea lopullisen bittien järjestyksen todennäköisyyden. Differentiaalinen analyysi vaatii, että analysoija voi syöttää omaa selkotekstiään salaimeen.

Lineaarinen kryptoanalyysi muistuttaa huomattavasti differentiaalista. Se on kuitenkin uudempi ja sitä on helpompi käyttää. Lineaarisessa analyysissa salaimen kierroksen sisään ja ulostulevia bittejä vertaillaan ja etsitään kohtia, joissa sisään menevän ja ulos tulevan bitin välinen todennäköisyys on eri kuin 0,5. Kierrosten väliset riippuvuudet yhdistetään ensin toisiinsa ja sen jälkeen avaimen bitteihin. Lineaarisen yhtälön löytäminen riippuu salaimen funktiosta. Kaiken onnistuessa hyvin analysoija voi saada haltuunsa lähes kaikki avaimen bitit ja loput analysoija voi selvittää vertailemalla tuntemiansa selvätekstejä niitä vastaaviin salateksteihin.
(Järvinen 2003b, 230)

4.4.3 Raa'an voiman murto

Raa'an voiman menetelmissä hyökkääjällä on tiedossa selväkielinen sanoma, sekä salasanoma. Raa'an voiman menetelmissä siis kokeillaan kaikki mahdolliset avaimet läpi. Kuitenkaan raa'an voiman tekniikkaa ei ole järkevää käyttää esimerkiksi DES-salaukseen, koska jokaisen ~7.2×10^{16} avaimen kokeilemiseen menisi aikaa 2000 vuotta miljoonan avaimen sekuntivauhdilla.

Raa'an voiman menetelmää voidaan nopeuttaa rinnakkainlaskennalla, jossa murtajalla on käytössään laite, jolla murtaja pystyy suorittamaan 10^6 rinnakkaista DES purkua. Löytyy myös laitteita, joilla pystytään kokeilemaan kaikki DES avaimet läpi 28-sekunnin aikana. (Kerttula 1998, 127-128)

5 KANNETTAVAN TIETOKONEEN KIINTOLEVYN SALAUS

Nykyään ihmisillä on käytössä paljon kannettavia tietokoneita, joilla he hoitavat sekä yksityisiä, että työasioitaan. Kannettavalla tietokoneella saatetaan pitää tietoa, joka ei saisi päästä ulkopuolisten käsiin. Silloin syytä harkita tietokoneen kiintolevyn salaamista. Salaamisen toteuttamista varten on kehitetty sovelluksia, joilla on erilaisia ominaisuuksia. On myös syytä muistaa, että myös työhuoneessa tai kotona pöydällä olevan tietokoneen kiintolevy voidaan salata samoilla menetelmillä, kuin kannettavan tietokoneen kiintolevy.

5.1 Tarkoitukseen soveltuvien sovellusten esittely

Esittelen muutaman kiintolevyjen salaamistarkoitukseen kehitetyn sovelluksen ominaisuuksia, sekä valitsen yhden esimerkkisovelluksen, jonka käyttöönoton dokumentoin. Valitsin esiteltäväksi eniten käytettyjä ilmais-sovelluksia.

5.1.1 Microsoft bitlocker encryption

Bitlocker Encryption on Microsoftin kehittämä salaussovellus, joka sisältyy Windows Vista Enterprise, sekä Ultimate ja Windows 7 versioihin, sekä Windows 8 Professional ja Enterprise versioihin. Bitlockerissa voi käyttää AES-salausalgoritmia diffuserilla, joko 128 tai 256-bittisellä salausavaimella. Oletuksena avaimen pituus on 128-bittiä.(http://technet.microsoft.com/en-us/library/ee449438(v=ws.10).aspx)

Bitlocker käyttää TPM-turvapiiriä, joka on tietokoneeseen sisäänrakennettu piiri, jonne Bitlocker tallentaa salausavaimen. Bitlocker salaa koko käyttöjärjestelmäaseman ja tallentaa salausavaimen TPM-turvapiiriin. Jos TPM-Turvapiiriä ei ole, niin salausavaimen voi tallentaa esimerkiksi muistitikulle. Sillä voi salata myös muut asemat, sekä Bitlocker to go toiminnolla esimerkiksi USB-muistitikun.(http://windows.microsoft.com/fi-fi/windows/protect-files-bitlocker-drive-encryption#1TC=windows-8)

Tietokonetta käynnistettäessä vertailee TPM-turvapiiri käyttöjärjestelmästä otettuja arvoja aiemmin tallentamiinsa arvoihin. TPM-turvapiirin havaitessa Windowsin käynnistyksessä jotain aiemmasta poikkeavaa niin se ei vapauta avainta ja käyttäjä ei pääse käsiksi tiedostoihin. Jos mahdollinen poikkeus havaitaan, niin silloin tarvitaan erillistä Bitlocker palautus-avainta. (http://windows.microsoft.com/fi-fi/windows-vista/bitlocker-drive-encryption-overview)

5.2 Truecrypt

Truecrypt on reaaliaikaiseen tiedon salaukseen tarkoitettu sovellus, eli se purkaa ja salaa tiedon ilman käyttäjän erilisiä toimenpiteitä, käyttäjä ei siis huomaa salausta tai salauksen purkua. Tiedostoja voi avata ja tallentaa normaaliin tapaan. Sillä voi salata kokonaisen kiintolevyn, kiintolevyosion, USB-muistin. Truecryptillä voi salata myös virtuaali-osioita.

Truecryptin eri versioita on ladattu yhteensä yli 29 miljoonaa kertaa. Sen ensimmäinen versio 1.0 julkaistiin 2.2.2004 ja uusin versio 7.0 2.2.2012.

Truecryptistä on olemassa versiot seuraaville käyttöjärjestelmille:

- Windows 7 (32-bit ja 64-bit)
- Windows Vista (32-bit ja 64-bit)
- Windows XP (32-bit ja 64-bit)
- Windows Server 2008 R2 (64-bit)
- Windows Server 2008 (32-bit ja 64-bit)
- Windows Server 2003 (32-bit ja 64-bit)
- Windows 2000 Service pack 4
- Mac OS X 10.8 Mountain Lion (32-bit ja 64-bit)
- Mac OS X 10.7 Lion (32-bit ja 64-bit)
- Mac OS X 10.6 Snow Leopard (32-bit)
- Mac OS X 10.5 Leopard
- Mac OS X 10.4 Tiger
- Linux (32-bit ja 64-bit versiot 2.6 tai yhteensopivalle kernelille)

Truecryptissä on valittavissa seuraavat salausalgoritmit:

- AES
- Serpent
- Twofish
- AES-twofish
- AES-twofish-serpent
- Serpent-AES
- Serpent-twofish-AES
- Twofish-serpent

Sekä seuraavat hash-algoritmit: RIPEMD-160, SHA-152, Whirpool.
(http://www.truecrypt.org/)

5.2.1 Secustar Drivecrypt

Drivecrypt on Secustarin kehittämä ilmais-sovellus kiintolevyjen ja ulkoisten muistien salaamiseen. Standard edition sallii 16 Teratavun salaamisen ja Home edition 4 Gigatavun

Drivecryptissä voi käyttää seuraavia salaustekniikoita:

- AES
- BLOWFISH
- TEA16
- TEA32
- DES
- Triple DES
- Misty 1
- Square.

Drivecrypt tukee seuraavia käyttöjärjestelmiä:

- Windows 8 (32-bit ja 64-bit)
- Windows 7 (32-bit ja 64-bit)
- Windows Vista (64-bit)
- Windows XP (32-bit)
- Windows 2003 (32-bit)
- Windows 2000 (32-bit)

Drivecryptissa on mahdollista asettaa pääsalasana, jolla käyttäjän unohtama salasana voidaan palauttaa. Käyttäjä voi käyttää oman tunnistautumiseensa sormenjälkeä, älykorttia ja salasanaa tai salasanaa ja sormenjälkitunnistusta tai älykorttia. (http://www.securstar.com/products_drivecrypt.php)

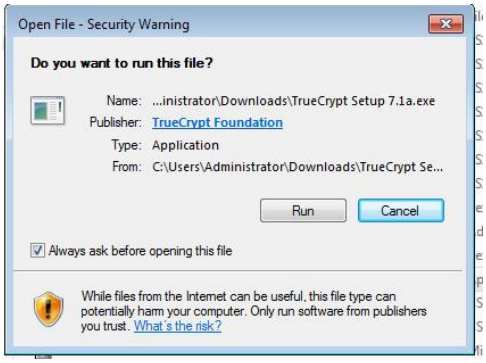
5.3 Sovelluksen valinta ja valitun sovelluksen käyttöönotto

Asennettavan sovelluksen valinta oli helppo. Asennettavaksi sovellukseksi valitsin Truecryptin. Valintaani perustelen sillä, siinä on valittavissa useimpia salausvaihtoehtoja. Se on myös eniten käytetty, joten se on myös todennäköisesti hyvin luotettava sovellus.

Käyttöönoton toteutan virtuaalityöasemalle, jossa on käyttöjärjestelmänä Windows 7 Professional, koneessa on kaksi virtuaaliosiota, toinen Windows järjestelmälle ja toinen tiedostojen säilyttämiseen, salaus toteutetaan tiedostojen säilyttämiseen tarkoitetulle osiolle.

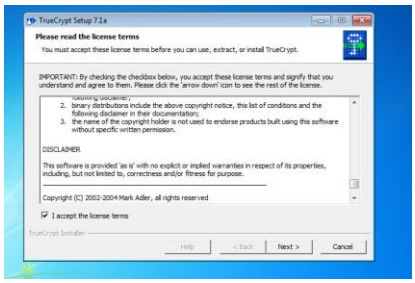
Kuva 9 Truecryptin kotisivu

Ensin ladataan sovellus koneen kiintolevylle TrueCryptin kotisivulta. (kuvassa 9.)



Kuva 10 Turvallisuusvaroitus

Latauksen ollessa valmis aloitetaan sovelluksen asennus. Windows pyytää hyväksyntää sovelluksen käynnistykselle. (kuvassa 10.)

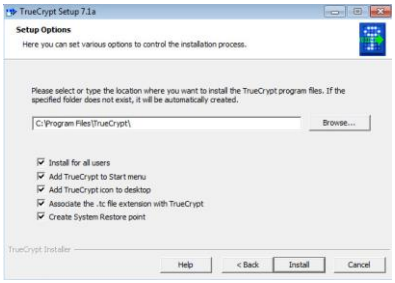


Kuva 11 Lisenssin hyväksyminen¨

Heti asennuksen alussa TrueCrypt pyytää hyväksymään lisenssin. Luetaan ja hyväksytään lisenssi, jonka jälkeen seuraavaan kohtaan. (Kuvassa 11.)

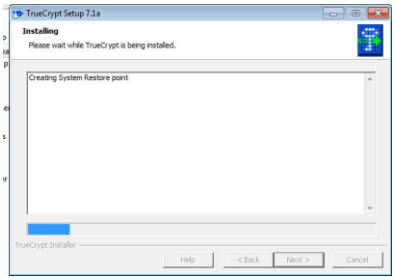
Kuva 12 Asennus tai asennustiedostojen purku.

Asennetaan TrueCrypt ja valitaan oletuksena oleva ”install”. (Kuvassa 12.)



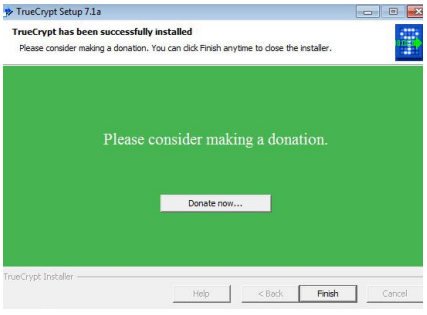
Kuva 13. Asennuskansion valinta.

Asennetaan TrueCrypt tässä tapauksessa oletuskansioon. (kuvassa 13)



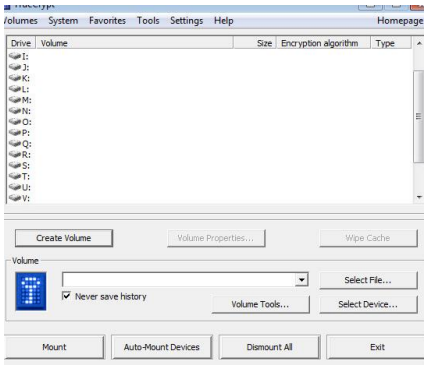
Kuva 14 Asennus

Asennusohjelma asentaa sovelluksen. (kuvassa 14)



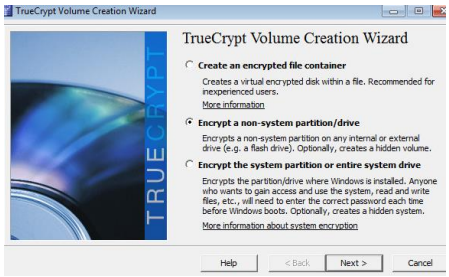
Kuva 15 Asennus loppu.

Kun asennus on valmis, niin sovelluksen kehittäjät pyytävät mahdollisia lahjoituksia. (kuvassa 15).



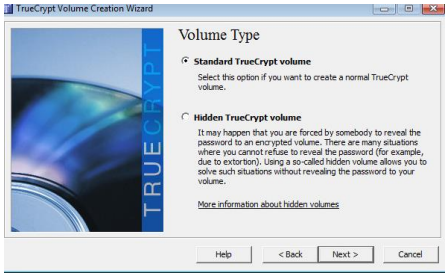
Kuva 16. TrueCrypt aloitusvalikko.

Valitaan valikosta ”Create Volume” (kuvassa 16), jonka jälkeen avautuu asennusvelho. (Kuvassa 17.)



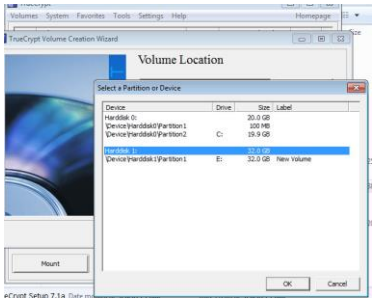
Kuva 17. Valitaan mitä halutaan salata.

Tässä tapauksessa kun ei salata asemaa johon Windows on asennettu valitaan ”Encrypt a non-system partition/drive”.



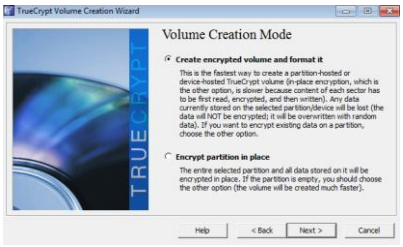
Kuva 18. Osion tyypin valinta.

Tämän jälkeen valitaan halutaanko luoda normaali vai piilotettu osio, tässä tapauksessa luodaan normaali osio. (Kuvassa 18.)



Kuva 19. Salattavan levyn valinta.

Valitaan, mikä levy tai levyosio halutaan salata. Tässä tapauksessa salataan kirjaimella E näkyvä osio, valitaan siis ”\Device\Harddisk 1\Partition 1”.(Kuvassa 19.)

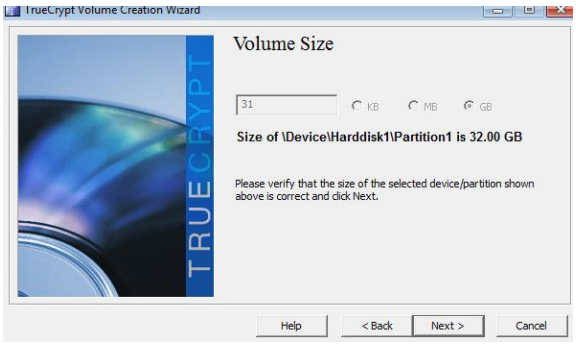


Kuva 20. Alustus vai ei.

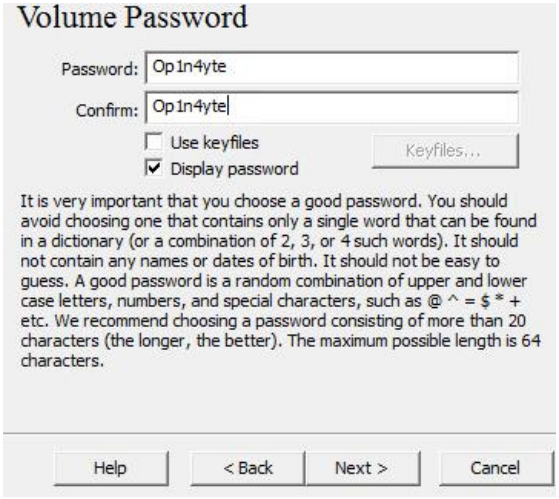
Valitaan miten salattu osio luodaan, tässä tapauksessa kun asemalla ei vielä ole ainuttakaan tiedostoa voidaan asema alustaa. Jos osiolla olisi tiedostoja, voisin valita alemman vaihtoehdon. (Kuvassa 20.)

Kuva 21. Salausmenetelmän valinta.

Valitaan salausmenetelmä, oletuksena on AES joka on riittävän tehokas. (Kuvassa 21.)

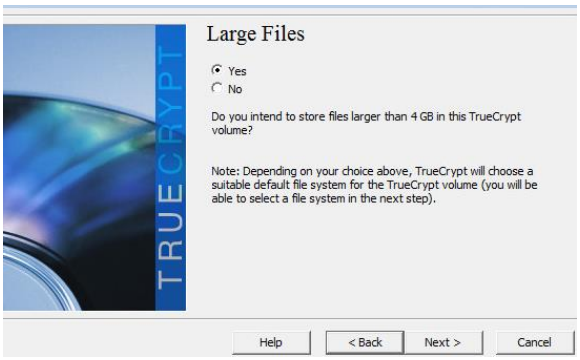


Kuva 22. Osion koon määritys.

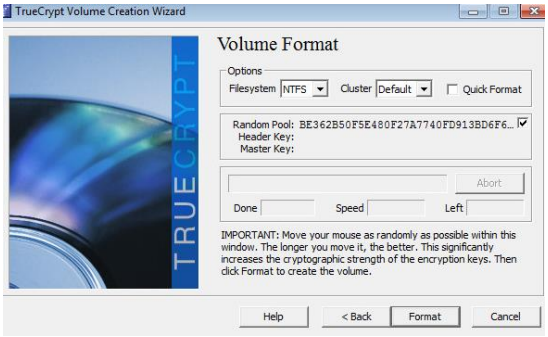


Kuva 23. Salasana ja avaintiedosto.

Annetaan salasana, jota käytetään salauksessa. Tämä on tärkeä pitää tallessa, jos salasanan unohtaa, silloin salatulla asemalla olevia tiedostoja ei voida enää käyttää. ”Use keyfiles” valinta voidaan jättää tyhjäksi, mutta jos halutaan salasanan lisäksi vielä enemmän turvaa voidaan käyttää avaintiedostoa. Avaintiedostoa ei tule muuttaa mitenkään, koska se perustuu tarkistus-summaan.



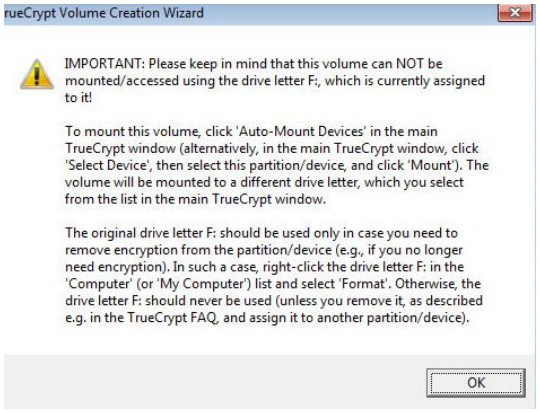
Kuva 24. Säilytettävien tiedostojen koko.



Kuva 25. Tiedostojärjestelmän valinta.

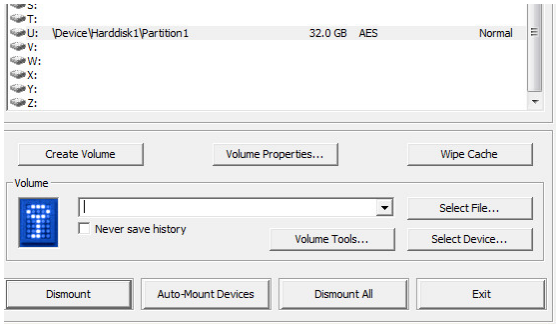
Valitaan säilytetäänkö salatulla asemalla alle vai yli 4GB tiedostoja (Kuvassa 24.), tämä vaikuttaa käytettävään tiedostojärjestelmään. Jos valitaan alle 4GB, niin tiedos-

tojärjestelmän oletusarvona on FAT, jos taas sallitaan kaiken kokoisten tiedostojen säilytys, niin oletuksena on NTFS. On kannattavaa valita tuki 4GB tiedostoille, vaikka niiden säilyttämiselle ei heti olisi tarvetta, tulevaisuudessa saattaa kuitenkin olla. (Kuvassa 25.)

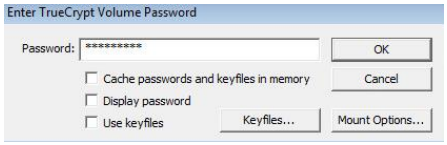


Kuva 26. Huomautus aseman tunnuskirjaimesta.

Kun alustaminen ja salatun aseman luominen on suoritettu, niin TrueCrypt ilmoittaa, että aseman normaalia tunnuskirjainta ei voida käyttää kuin poistettaessa kaikki tieto kyseiseltä osiolta. (Kuvassa 26) Osio tulee ottaa käyttään TrueCrypt sovelluksen kautta.

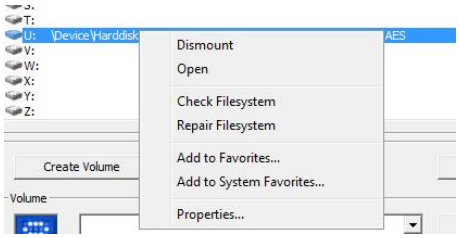


Kuva 27. TrueCrypt aloitusnäkymä.

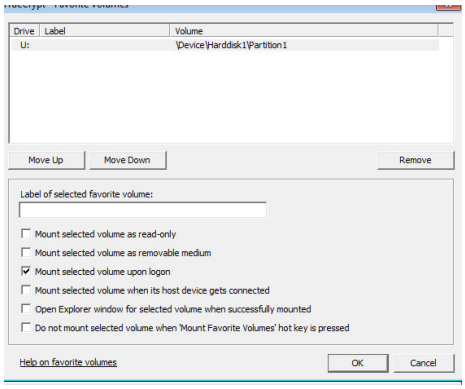


Kuva 28. Salasanan antaminen.

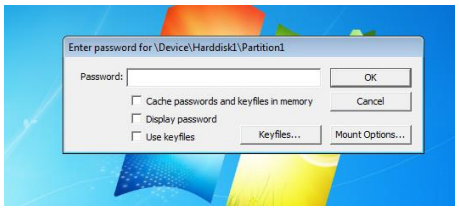
Valitaan ”Auto-Mount Devices”, joka ottaa kaikki salatut asemat käyttöön. Tämän jälkeen pyydetään salasanaa ja mahdollista avaintiedostoa, joka annettiin osion luomisen yhteydessä. Oletuksena tämä joudutaan tekemään aina, kun käyttäjä uloskirjautuu windowsista tai käynnistää tietokoneen uudestaan. (Kuvassa 28)



Kuva 29. Add to Favorites

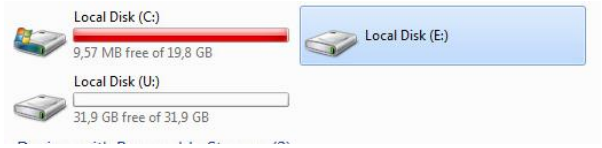


Kuva 30. Mount selected volume upon logon



Kuva 31. TrueCrypt pyytää salasanaa.

Muutetaan asetuksia niin, että Windows pyytää salatun aseman salasanaa heti kirjautumisen jälkeen (kuva 31). Asetusten muuttaminen on helppo työ (kuvat 29 ja 30). Lisätään siis salattu asema suosikkeihin ja sen jälkeen muutetaan asetus niin, että liitetään salattu asema kirjautumisen yhteydessä.



Kuva 32. Näkymä Oma Tietokoneessa.

Resurssienhallinnassa salattu osio näkyy tässä tapauksessa kirjaimella E. TrueCryptillä käyttöönotettuna asema näkyy kirjaimella U. (Kuvassa 32)

6 TIETOJEN SALAUS ANDROID LAITEESSA JA ANDROID LAITTEEN ETÄTYHJENNYS

Android laitteille on olemassa monia erilaisia tietojen salaamiseen soveltuvia sovelluksia, niin ilmaisia kuin maksullisiakin. Sovellusten laadussa on suuria eroja, joten on hyvä miettiä mitä sovellusta oikeasti kannattaa käyttää. Lähiaikoina on ollut paljon puhetta siitä, kuinka eri sovellukset vuotavat tietoa ulkopuolisille.

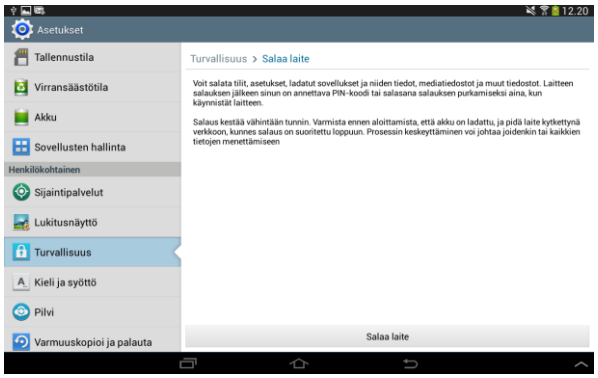
Seuraavaksi esittelen Androidin oman koko laitteen salaamiseen tarkoitetun sovelluksen, sekä otan sen käyttöön.

6.1 Androidin oma salaussovellus

Android 2.3.4 ja uudemmissa Android versioissa on sisäänrakennettu salaussovellus, jolla voidaan salata koko laite ja kaikki laitteen sisällä oleva tieto. Ennen laitteen salaamista on kuitenkin hyvä ottaa laitteessa olevista tärkeistä tiedostoista varmuuskopio, sillä jos salauksessa jokin menee pieleen voivat tiedot olla sen jälkeen käyttökelvottomia. Salauksen jälkeen laite saattaa olla hieman hitaampi, kuin ennen salausta.

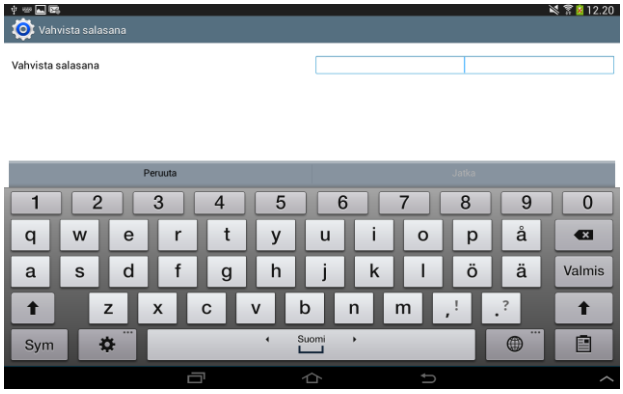
Kuva 33. Android valikko

Android asetuksista turvallisuusvalikosta löytyy salaa laite kohta, josta laitteen salauksen voi aloittaa. (kuvassa 33)



Kuva 34. Laitteen salauksen aloitus.

Laitteen salaaminen itsessään on helppo toimenpide. Ensin pitää varmistua, että laitteen akun varaus on yli 80% ja sen jälkeen laite pitää liittää laturiin. Laitteen salaus kestää laitteen muistin koosta riippuen noin tunnin, tässä tapauksessa se kesti noin puoli tuntia. (kuvassa 34)



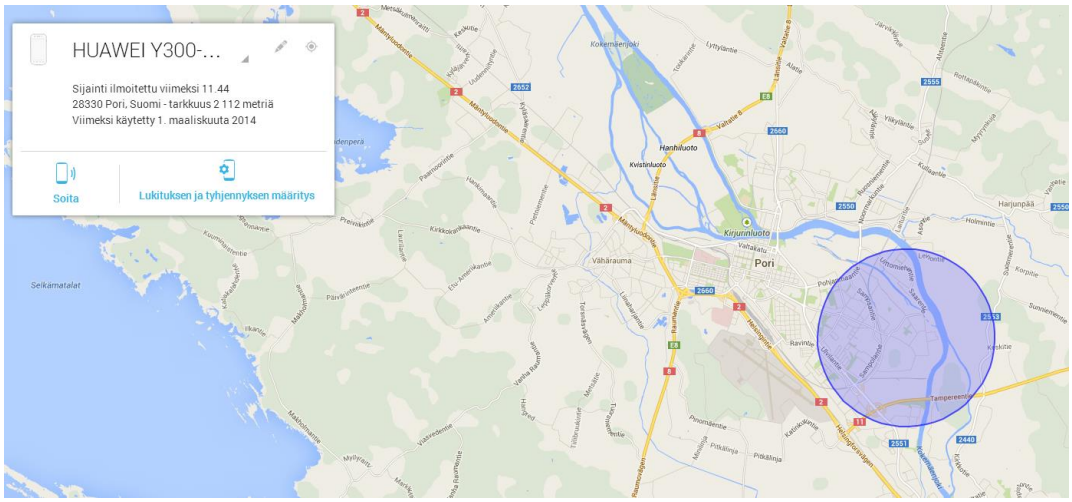
Kuva 35. Salasanan vahvistus.

Laitteelle annetaan salasana, jota käytetään salauksessa.. Salasanan muistaminen on tärkeää, koska jos se unohtuu, niin laitteesta ei saada mitään tietoa pois. (Kuvassa 35.)(http://www.howtogeek.com/141953/how-to-encrypt-your-android-phone-and-why-you-might-want-to/)

6.2 Laitteen paikannus ja etätyhjennys Android Device Managerilla

Loppuvuodesta 2013 Google julkaisi Androidille sovelluksen, jonka avulla kadonnut laite voidaan paikantaa ja etätyhjentää. Android Device Manager vaatii Android 2.2 tai uudemman käyttöjärjestelmän, sekä datayhteyden ja Google tilin. Sovellus on tarkoitettu lähinnä älypuhelimille, mutta sen voi asentaa myös tablet tietokoneeseen.

Sovelluksen avulla voidaan Google Mapsin kautta seurata puhelimen sijaintia, samalla näkyy, kuinka tarkka sijainti on (Kuvassa 36). Sen avulla voi myös soittaa kadonneeseen laitteeseen, vaikka laite olisi äänettömällä, puhelun ajan äänenvoimakkuus on täysillä. Sovellus soittaa laitteeseen viiden minuutin ajan, tai kunnes laite löydetään. Tietojen etätyhjennys onnistuu myös tuota kautta, kun myös laitteesta on etätyhjennyksen salliva asetus päällä. (http://taskumuro.com/google-julkaisi-android-laitteiden-paikannuspalvelun)



Kuva 36. Android paikannus.

7 YHTEENVETO

Työtä tehdessäni huomasin, että kuin helppoa laitteiden muistien salaaminen on. Se on helppo tapa suojata laitteen sisällä oleva tieto, jos laite joutuu vääriin käsiin. Kuitenkaan tämä yksinkertainen tapa suojata tietoa ei ole saanut niin suurta kiinnostusta osakseen, kuin sen pitäisi saada.

Työni painottui jonkin verran enemmän teoriaosuuteen, kuin käytännön osuuteen, käytännönosuuteen kulutettu aika on ehkä 1/5 osa koko työhön kuluneesta ajasta. Teoriaosuuden suurin haaste oli rajata aihe niin, että se kertoisi tasapuolisesti kaikista esitellyistä salausmenetelmistä, joistain menetelmistä löytyi tietoa paljon helpommin kuin toisista. Oli myös haastavaa valita, minkälaisen käytännönosuuden työhöni tekisin, koska työni suunnittelun aloitin siitä, mihin aiheeseen haluisin tutustua ja valitsin aiheeksi salausmenetelmät. Lopulta kuitenkin olen ihan tyytyväinen tähän työhön, tosin käytännönosuus olisi voinut olla haastavampi.

LÄHTEET

Anonymous 2002: Hakkerin Käsikirja. Edita Prima Oy. Helsinki

BBC:n verkkosivut. Viitattu 23.12.2013
Saatavissa:
http://www.bbc.co.uk/history/topics/enigma

Elk Riverin alueen koulujen verkkosivut. Viitattu 22.12.2013
Saatavissa:
http://www.elkriver.k12.mn.us/webpages/sbraun/research.cfm?subpage=26862

Fisher Business collegen verkkosivut. Viitattu 22.12.2013
Saatavissa:
http://fisher.osu.edu/~muhanna.1/pdf/crypto.pdf

Goole Playn kotisivut. Viitattu 5.3.2014
Saatavissa:
(https://play.google.com/store/search?q=encryption&c=apps&hl=fi)

Järvinen Petteri 2003a: IT-Tietosanakirja. Docendo Oy. Jyväskylä.

Järvinen Petteri 2003b: Salausmenetelmät. Docendo Oy. Jyväskylä.

Kerttula Esa 1998: Tietoverkkojen tietoturva. Oy Edita Ab. Helsinki

Kioskean verkkosivut. Viitattu 24.12.2013
Saatavissa:
http://en.kioskea.net/contents/134-introduction-to-encryption-with-des)

Logicalsecurityn verkkosivut. Viitattu 21.12.2013
Saatavissa:
http://www.logicalsecurity.com/resources/whitepapers/Cryptography.pdf

Ronald L. Krutz & Russel Dean Vines 2003: Tietoturva Sertifikaatti. Edita Prima Oy. Helsinki.
Thomas Tom 2005: Verkkojen tietoturva perusteet. Edita Prima Oy. Helsinki.

Secustarin kotisivut. Viitattu 3.1.2014
Saatavissa:
http://www.securstar.com/products_drivecrypt.php

Simon Singh 1999: Koodikirja. Gummerus Kirjapaino Oy. Jyväskylä

TrueCryptin kotisivut. Viitattu 4.3.2014
Saatavissa:
http://www.truecrypt.org/

Visual inc. verkkosivut. Viitattu 22.12.2013
Saatavissa:
http://visual.ly/history-encryption

YouTube video Open course online . Viitattu 24.12.2013
Saatavissa:
http://www.youtube.com/watch?v=UgFoqxKY7cY