RSA – Julkisen avaimen salakirjoitusmenetelmä

Perusteet, algoritmit, hyökkäykset

Matti K. Sinisalo, FL

GeraCap

Alkuluvut

- Alkuluvuilla tarkoitetaan lukua 1 suurempia kokonaislukuja, jotka eivät ole tasan jaollisia millään muulla nollaa suuremmalla kokonaisluvulla, kuin luvulla 1 ja itsellään.
- Pienimmät alkuluvut ovat 2, 3, 5, 7, 11, 13, 17, 19, ...
- Alkuluvut ovat lukuja, jotka eivät esiinny kertotaulussa ensimmäistä riviä ja ensimmäistä saraketta lukuunottamatta.
- Yksinkertaisin tapa todeta luku alkuluvuksi, on kokeilujako.
- Kokeilujaossa lukua yritetään jakaa kokonaisluvuilla, jotka ovat suurempia kuin 1 mutta pienempiä kuin tutkitun luvun neliöjuuri.
- Kokeilujako on hyvin tehoton menetelmä suuria lukuja käsiteltäessä.
- Alkulukuja merkitään yleisesti kirjaimilla p ja q.

GeraCap

Aritmetiikan peruslause

- *Aritmetiikalla* tarkoitetaan luvuilla laskemista.
- Alkulukujen merkitys aritmetiikassa perustuu *aritmetiikan peruslauseeseen*, jonka mukaan jokainen ykköstä suurempi kokonaisluku voidaan tekijöiden järjestystä vaille yksikäsitteisesti esittää alkulukujen tulona.
- Alkulukuja, joilla jokin luku on tasan jaollinen, sanotaan ko. luvun *alkutekijöiksi*.
- Esim. $360=2^3 3^2 5$
- Jos luvun alkutekijät tunnetaan, ne voidaan helposti kertoa keskenään.
- Suuren kokonaisluvun alkutekijöiden etsiminen, luvun *tekijöihinjako* eli *faktorointi*, on yleisesti laskennallisesti vaativa tehtävä.
- Tähän yksisuuntaisuuteen perustuu RSA-järjestelmän tarjoama tietoturva.

GeraCap

Eulerin funktio

- *Eulerin funktiolla* positiivisesta kokonaisluvusta n tarkoitetaan niiden positiivisten kokonaislukujen, jotka ovat pienempiä kuin n ja joilla ei ole yhteisiä tekijöitä luvun n kanssa, lukumäärää.

- Siis $\phi(n) = \#\{k \mid 0<k<n \text{ ja } syt(k,n)=1\}$.

- Jos p on alkuluku ja k positiivinen kokonaisluku, niin $\phi(p^k) = p^{k-1}(p-1)$. Erityisesti $\phi(p) = p-1$.

- Jos $syt(m,n) = 1$, niin $\phi(mn) = \phi(m)\phi(n)$.

- Erityisesti, jos p ja q ovat erisuuria alkulukuja, niin $\phi(pq) = \phi(p)\,\phi(q) = (p-1)(q-1)$. Tätä tulosta hyödynnetään RSA-menetelmässä.

- Eulerin funktion arvon laskeminen on laskennallisesti yhtä vaikea tehtävä kuin ko. luvun tekijöihinjako.

GeraCap

Jakojäännösaritmetiikka

- *Jakojäännösaritmetiikalla* l. *modulaariaritmetiikalla* tarkoitetaan tietyn kokonaisluvun, *modulin*, jakojäännöksillä laskemista.
- Jakojäännösaritmetiikan keskeinen työkalu on *kongruenssirelaatio.*
- Kokonaislukujen a ja b sanotaan olevan *kongruentteja modulo* n, jos niiden jakojäännökset luvulla n jaettaessa ovat yhtäsuuret. Tällöin merkitään $a \equiv b \pmod{n}$.
- Kongruenssirelaation käyttö muistuttaa hyvin pitkälti yhtäsuuruusmerkin = käyttöä matematiikassa.
- Kongruensseilla laskettaessa kokonaisluku voidaan tyypillisesti korvata sen kanssa kongruentilla toisella kokonaisluvulla.
- Jakojäännösaritmetiikan peruslaskusäännöt ovat yksinkertaisia, mutta niiden sujuva käyttö vaatii rutinoitumista.

GeraCap

Jakojäännösaritmetiikan käänteisluku

- Olkoon n jokin positiivinen kokonaisluku. Jos kokonaisluvut a ja b toteuttavat ehdon $ab \equiv 1 \pmod{n}$, niin sanomme, että luvut a ja b ovat toistensa *käänteislukuja modulo* n.
- Lukujen a ja b tulo antaa siis luvulla n jaettaessa jakojäännökseksi luvun 1.
- Esim. luvut 3 ja 7 ovat toistensa käänteislukuja modulo 10, koska $3*7=21 \equiv 1 \pmod{10}$.
- Jakojäännösaritmetiikan käänteisluku voidaan laskea:
- Fermat'n pientä lausetta ja potenssiinkorotusalgoritmia käyttäen, jos n on alkuluku.
- Eulerin lausetta ja potenssiinkorotusalgoritmia käyttäen, jos luvun n tekijöihinjako tunnetaan.
- Yleisesti laajennettua Eukleideen algoritmia käyttäen.
- Viimeksi mainittu menetelmä on tehokkain, sillä se ei edellytä alkulukutodistusten ja tekijöihinjakomenetelmien käyttöä.

GeraCap

RSA-järjestelmän avainten valinta

- Valitaan kaksi isoa alkulukua, merk. p ja q
- Kerrotaan luvut keskenään, merk. n = pq
- Lasketaan *Eulerin funktion* arvo luvusta n: $\phi(n) = (p-1)(q-1)$
- Valitaan *julkinen salakirjoitusavain (kryptausavain)* e väliltä $1 \ldots \phi(n)-1$
- Lasketaan luvun e käänteisluku modulo $\phi(n)$, merk. d
- Nyt $ed \equiv 1 \pmod{\phi(n)}$
- Luku d on *salainen purkuavain l. dekryptausavain*

GeraCap

Järjestelmän käyttö

- Olkoon m välillä 0 … n−1 oleva kokonaisluku (salattava viesti).
- Viestin lähettäjä laskee luvun m^e jakojäännöksen luvun n suhteen.
- Näin saatu luku, merk. c muodostaa kryptatun viestin.
- Vastaanottaja laskee luvun c^d jakojäännöksen luvun n suhteen.
- Nyt $c^d \equiv (m^e)^d = m^{ed} \equiv m^1 = m \pmod{n}$.

GeraCap

Potenssiinkorotusalgoritmi

- Potenssiinkorotusalgoritmi on yksi keskeisimmistä RSA-menetelmän käyttämistä algoritmeista.
- Sitä käytetään sekä viestin salaamiseen (kryptaukseen), että sen avaamiseen (dekryptaukseen).
- Potenssiinkorotusalgoritmi perustuu eksponentin toistuvaan puolittamiseen ja neliöönkorotukseen (kertolaskuun).
- Esim. Laskettava $y = a^b$, missä a ja b ovat kokonaislukuja.
- Jos b on parillinen, niin $y = (a^{b/2})^2$
- Jos b on pariton, niin $y = a(a^{(b-1)/2})^2$

GeraCap

Laajennettu Eukleideen algoritmi

- Laajennetun Eukleideen algoritmin (extended euclidean algorithm) avulla voidaan löytää sellaiset kokonaisluvut a ja b, että $am + bn = syt(m,n)$
- Luvut m ja n ovat ns. *keskenään jaottomia*, jos $syt(m,n) = 1$.
- Ts. luvut m ja n ovat keskenään jaottomia, jos niillä ei ole yhteisiä alkulukutekijöitä.
- Laajennetun Eukleideen algoritmin tärkein käytännön sovellus on jakojäännösaritmetiikan käänteislukujen laskeminen.
- Jos nimittäin $am+bn = 1$, niin $am \equiv 1 \pmod n$ ts. Luku a on luvun m käänteisluku modulo n.
- Laajennettu Eukleideen algoritmi on tehokas. Sen avulla voidaan käsitellä lukuja, joissa on jopa kymmeniä tuhansia numeroita.

GeraCap

Fermat'n pieni lause

- Tehokkaimmat nykyisin tunnetut alkulukutestit perustuvat tavalla tai toisella Fermat'n pieneen lauseeseen.
- Fermat'n pieni lause sanoo, että jos a on jokin kokonaisluku ja p on sellainen alkuluku, että p ei ole a:n tekijä, niin $a^{p-1} \equiv 1 \pmod{p}$.
- Kääntäen, jos n on jokin nollaa suurempi kokonaisluku ja a sellainen kokonaisluku, että syt(a,n)=1 ja luvun a^{n-1} jakojäännös luvulla n jaettaessa on jokin muu kuin 1, niin voimme olla varmoja siitä, että n ei ole alkuluku.
- Luvun toteamiseen alkuluvuksi Fermat'n pieni lause ei suoraan sellaisenaan sovellu. On nimittäin olemassa sellaisia yhdistettyjä lukuja, jotka toteuttavat Fermat'n pienen lauseen kaikilla luvun a valinnoilla.
- Näitä lukuja sanotaan Carmichaelin luvuiksi.

GeraCap

Fermat'n pieni lause (jatk.)

- Lukuja n, jotka toteuttavat Fermat'n pienen lauseen kantaluvulla a (ts. $a^{n-1} \equiv 1 \pmod{n}$), sanotaan *kannan a pseudoalkuluvuiksi*.
- Lukuja, jotka toteuttavat Fermat'n pienen lauseen useilla eri kantaluvun a arvoilla, sanotaan *todennäköisiksi alkuluvuiksi* ja vastaavaa testiä *heikoksi alkulukutestiksi.*
- Heikko alkulukutesti ei siis takaa sataprosenttisella varmuudella sitä, että tutkittu luku on alkuluku.
- Ns. *vahvalla alkulukutestillä* voidaan määrätä sataprosenttisella varmuudella se, onko tutkittu luku alkuluku vai ei.

GeraCap

Eulerin lause

- Eulerin lause on yksi tärkeimmistä RSA:n pohjalla olevista matemaattisista perustuloksista.
- Jos n on jokin positiivinen kokonaisluku ja a sellainen kokonaisluku, että syt(a,n)=1, niin Eulerin lauseen mukaan $a^{\phi(n)} \equiv 1 \pmod{n}$.
- Eulerin lause on Fermat'n pienen lauseen yleistys.

GeraCap

Alkulukutestien polynomiaikaisuus

- Algoritmia sanotaan polynomiaikaiseksi, mikäli sen suoritusajan voidaan todeta olevan verrannollinen syötteen pituuden johonkin kokonaislukupotenssiin.
- Alkulukutesti on polynomiaikainen, jos on olemassa sellainen positiivinen kokonaisluku k, että testin suoritusaika t toteuttaa ehdon $t = O((\log(n))^k)$ jokaisella testattavalla kokonaisluvulla n. (Huom. luku log(n) kuvaa tässä syötteen pituutta.)
- Avoin kysymys oli vielä äskettäin, onko polynomiaikaista vahvaa alkulukutestiä mahdollista muodostaa.
- Intialaiset tutkijat ovat äskettäin esittäneet ns. AKS-testin (Agrawal, Kayal, Saxena), jonka on todettu olevan polynomiaikainen.
- Sellaisenaan testi on kuitenkin epäkäytännöllinen, sillä sen suoritusaikaa rajoittavan polynomin aste on luokkaa 8.

GeraCap

Tekijöihinjakomenetelmät

- Systeemin varsinainen käyttäjä ei yleensä tarvitse tekijöihinjakomenetelmiä.
- Tekijöihinjakomenetelmien käyttäminen liittyy lähinnä systeemin murtoyrityksiin.
- Järjestelmän turvallisuus perustuu luvun n alkutekijöiden p ja q salassapitoon.
- Henkilö, joka tuntee luvun n alkutekijät, saa systeemin vaivattomasti murrettua.
- Nykyisillä tekijöihinjakomenetelmillä voidaan rutiininomaisesti jakaa tekijöihin lähes satanumeroisia lukuja.

GeraCap

RSA Challenge

- Turva-alan yritys RSA SECURITY pitää Internetissä listaa yhdistetyistä luvuista, joiden faktoroinnista yritys lupaa erisuuruisia palkkioita.
- Yrityksen tarkoituksena on luonnollisesti seurata (siviilipuolella käytettävien) tekijöihinjakomenetelmien kehitystasoa ja testata siten omien tuotteidensa turvallisuutta.
- RSA:n haasteluvut löytyvät internetistä osoitteesta http://www.rsasecurity.com/rsalabs/challenges/factoring/numbers.html
- Tällä hetkellä näistä luvuista pienin on 174 numeroinen luku
  188198812920607963838697239461650439807163563379417382700763356422988859715234665485319060606504743045317388011303396716199692321205734031879550656996 22130516875930765025705 9
- Palkkio tämän luvun tekijöihin jakajalle on 10000 dollaria.

GeraCap

RSA-160

- Date: Tue, 1 Apr 2003 Subject: RSA-160

- We have factored RSA160 by gnfs. The prime factors are: p=45427892858481394071686190649738831\
  65613714577846979325095998470925000 4157335359
  q=473880906038320161966338323037889 51\ 97326892292104095794474135464881202849390936 7

- The prime factors of p-1 are 2 37 41 43 61 541 13951723 726865585068607252226214637712149456933 4513 and 1040469870918042412 91 .

- The prime factors of p+1 are 2^8 5 3 3 13 98104939 25019146414499357 3837489523921 and 12881789233737946101473657780153835884 3 .

- The prime factors of q-1 are 2 9973 165833 113565073373690071091376382935 61 3694569081502991 81 and 3414553020359960488907 .

- The prime factors of q+1 are 2^3 3 3 13 82811 31715129 7996901997270235141 and 24105551744955147858438633224726891765307591 97.

- The computations for the factorization of RSA160 took place at the Bundesamt für Sicherheit in der Informationstechnik (BSI) in Bonn.

- Lattice sieving took place between Dec. 20, 2002 and Jan. 6, 2003, using 32 R12000 and 72 Alpha EV67. The total yield of lattice sieving was 323778082. Uniqueness checks reduced the number of sieve reports to 289145711. After the filtering step, we obtained an almost square matrix of size with 5037191 columns. Block Lanczos for this matrix took 148 hours on 25 R12000 CPUs. The square root steps took an average of 1.5 hours on a 1.8 GHz P4 CPU, giving the factors of RSA160 after processing the 6-th lanczos solution.

- F. Bahr J. Franke T. Kleinjung M. Lochter M. Böhm

Fermat'n tekijöihinjakomenetelmä

- Fermat'n tekijöihinjakomenetelmässä etsitään sellaiset kokonaisluvut x ja y, että $x^2 \equiv y^2 \pmod{n}$.
- Nyt $y^2 - x^2 \equiv 0 \pmod{n}$ ts. $(y-x)(y+x) \equiv 0 \pmod{n}$ ts. luku n jakaa tulon $(y-x)(y+x)$.
- Jos n=pq, missä p ja q ovat erisuuria alkulukuja, kuten RSA:ssa, niin on mahdollista, että luku p jakaa luvun y-x ja q jakaa luvun y+x (tai päinvastoin).
- Jos p jakaa luvun y-x, mutta q ei jaa sitä, niin syt(n, y-x) = p.
- Toisin sanoen luvun n ei-triviaali tekijä p löydetään Eukleideen algoritmilla tehokkaasti.
- Vaikeutena Fermat'n menetelmässä on lukujen x ja y löytäminen.
- Fermat'n menetelmä on kuitenkin perustana monille muille tehokkaammille menetelmille (Esim. Neliöseulamenetelmä).

GeraCap

Muita tekijöihinjakomenetelmiä

- Pollard p−1 −menetelmä
- Pollardin rho menetelmä
- Luokkaryhmämenetelmä (Class Group Factorization Method)
- Ketjumurtolukumenetelmä (Continued Fraction Factorization Algorithm),
- Elliptisten käyrien menetelmä (Elliptic Curve Factorization Method)
- Eulerin tekijöihinjakomenetelmä
- Lukukuntaseula (Number Field Sieve)
- Neliöseulamenetelmä (Quadratic Sieve)
- Williamsin p+1 menetelmä

GeraCap

Protokollat

- *Protokollalla* tarkoitetaan säännöstöä, jota noudattamalla käyttäjä voi varmistua järjestelmän turvallisuudesta käyttösovellusten yhteydessä.
- Protokolla kuvaa yksityiskohtaisesti, miten avainten käsittelyssä, viestin kryptauksessa ja dekryptauksessa jne. menetellään.
- Protokolla laaditaan teoreettisen tietämyksen perusteella.
- Sen noudattaminen ei kuitenkaan välttämättä vaadi yksityiskohtaista taustalla olevien kryptografisten primitiivien teoreettista tuntemusta.
- Protokollasta poikkeaminen (inhimillinen tekijä) muodostaa yhden merkittävimmistä tietoturvariskeistä.
- Julkisen avaimen infrastruktuurin (PKI) järjestelmät pyritäänkin suunnittelemaan siten, että protokollasta poikkeaminen ei ole mahdollista.

GeraCap

Esimerkki protokollavirheestä

- Järjestelmän laatija käyttää samaa alkulukua p kahden eri kryptosysteemin muodostamiseen.
- Siis $n_1 = pq_1$ ja $n_2 = pq_2$, missä p, q_1 ja q_2 ovat erisuuria alkulukuja.
- Tunkeutuja (intruder) laskee julkisten moduulilukujen n_1 ja n_2 suurimman yhteisen tekijän:
  $syt(n_1,n_2) = syt(pq_1,pq_2) = p*syt(q_1,q_2) = p*1 = p$.
- Näin hän saa molemmat moduuliluvut jaettua tekijöihin, jonka seurauksena molemmat järjestelmät murretaan.

GeraCap

Valitun salakielitekstin hyökkäys

- Oheinen valitun salakielitekstin hyökkäys on kuvattu Bruce Schneierin kirjassa Applied Cryptography s. 471. (Ks. viitteet).
- A, joka kuuntelee B:n tietoliikennettä, onnistuu sieppaamaan B:n julkisella avaimella salatun viestin c.
- A valitsee satunnaisluvun r, joka on pienempi kuin n.
- A laskee luvut $x \equiv r^e \pmod{n}$, $y \equiv xc \pmod{n}$ ja $t \equiv r^{-1} \pmod{n}$, missä e on B:n julkinen avain.
- Koska $x \equiv r^e \pmod{n}$, niin $x^d = (r^e)^d = r^{ed} \equiv r^1 = r \pmod{n}$.
- A pyytää B:tä allekirjoittamaan viestin y salaisella avaimellaan.
- B lähettää A:lle viestin $u \equiv y^d \pmod{n}$.
- A laskee $tu \equiv ty^d \equiv tx^dc^d \equiv trc^d \equiv c^d \equiv m \pmod{n}$.

GeraCap

Lähteitä

- W. Diffie, M. Hellman: New directions in cryptography, IEEE Transactions of Information Theory IT-22, 6 (Nov. 1976), 644-654.
- R. L. Rivest, A. Shamir, L. Adleman: A Method for Obtaining Digital Signatures and Public Key Cryptosystems, Communications of the ACM, Vol 21, No 2, 1978, 120-126.
- Bruce Schneier: Applied Cryptography - Protocols, Algorithms and Source Code in C, 2nd ed., John Wiley & Sons, 1996.
- Matti K. Sinisalo: Suurten kokonaislukujen tekijöihinjaosta ja alkulukutesteistä, Lisensiaatintyö, Oulun Yliopisto, 1994.

GeraCap