Toteutusdokumentti: RSA salauksen toteutus Javalla.

Ohjelman yleisrakenne

Sovelluksella pystyy luomaan RSA-salaukseen vaadittavat yksityiset ja julkiset avaimet, sekä salaamaan ja purkamaan RSA-salauksen.

Se toimii tekstikäyttöliittymällä, jota ohjataan antamalla sille ohjeen mukaisia käskyjä. Käskyistä löytyy toiminnot avainten luontiin, salaamiseen, purkamiseen. Avaimet ja viestit voi tallettaa ja lukea tiedostosta.

Sovelluksessani on myös kyvykkyyttä vertailla avaimen koon ja erinäisten muiden muuttujien vaikutusta toimintojen aikavaativuuteen. Toimenpiteet kertovat kauan ne kuluttavat aikaa ja tämän lisäksi on erillinen ”*comparison*” käsky, jolla voi ajaa kahta erilaista eräajoa, jotka laskevat keskimääräisesti kulunutta aikaa eri työvaiheissa.

Työn puutteet ja parannusehdotukset

BigIntegerin toteuttamisen olisin halunnut saada aikaiseksi itse. Käytinkin siihen paljon aikaa, mutta en saanut ongelmia ratkaistuksi. Oletin kykeneväni toteuttamaan sen kohtuullisessa ajassa, mutta noin 10 tunnin jälkeen aloin hyväksyä, etten saa sitä ajoissa valmiiksi. Ehkä jatkan sen selvittelyä vielä kurssin ulkopuolella, sillä harmittaa, että se jäi kesken.

Toteutukseni tuskin on yhteensopiva olemassa olevien RSA-salauksen toteutusten kanssa, joten siinä olisi yksi kehittämisen kohde.

Lisäyksenä sovellukseen voisi olla käytännöllisyyden vuoksi hyödyllistä yhdistää jokin tapa salata isompia määriä tietoa. RSA itsessään rajoittuu salaamaan suhteellisen pieniä määriä dataa. Sen rajoittavana tekijänä on, että se voi salata ja purkaa onnistuneesti vain bittimääräisesti käytössä olevaa avainta pienemmän määrän tietoa. Tämä yleisesti ottaen kierretään salaamalla RSA-algoritmillä jonkin symmetrisen salausmenetelmän avain, jota on ensin käytetty salaamaan muu tieto.

Lähteet

Wikipedia: RSA (cryptosystem) (http://en.wikipedia.org/wiki/RSA)

Tietoliikenteen salaaminen Java-sovelluksen ja tietokannan välillä, Miika Päivinen, 2005 (ftp://cs.joensuu.fi/pub/Theses/2005_MSc_Paivinen_Miika.pdf)

Introduction to Cryptography and RSA, Leonid Grinberg (http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-045j-automata-computability-and-complexity-spring-2011/lecture-notes/MIT6_045JS11_rsa.pdf)

Public Key Cryptography: RSA Encryption Algorithm, Art of the Problem (https://www.youtube.com/watch?v=wXB-V_Keiu8)

Source for java.math.BigInteger, Warren Levy (http://developer.classpath.org/doc/java/math/BigInteger-source.html)