MATEMATISKA INSTITUTIONEN
STOCKHOLMS UNIVERSITET
Samuel Lundqvist
9 December 2012

RSA

Alice väljer två primtal p och q och beräknar

- $n = pq$ (genom vanlig multiplikation)
- $\phi(n)$ genom att utnyttja att p och q är relativt prima, vilket betyder att $\phi(n) = \phi(pq) = \phi(p) \cdot \phi(q) = (p-1)(q-1)$.
- e så att $SGD(e, \phi(n)) = 1$. Vanligen väljer man e till ett mycket litet primtal och testar, t.ex. med Euklides algoritm, att e och $\phi(n)$ är relativt prima. Om så inte skulle vara fallet provar man ett annat litet primtal, osv.
- $d = e^{-1}$ i $\mathbb{Z}_{\phi(n)}$. Att $d = e^{-1}$ i $\mathbb{Z}_{\phi(n)}$ är ekvivalent med att $de = 1$ i $\mathbb{Z}_{\phi(n)}$, dvs att det existerar a så att $de + a \cdot \phi(n) = 1$. Detta är en diofantisk ekvation med d och a som obekanta! Därför kan d och a beräknas genom att utföra Euklides algoritm för att bestämma $SGD(e, \phi(n))$ och sedan gå baklänges i uträkningarna, se Exempel 1.

Alice publika nyckel är n och e. Alice privata nyckel är d (ibland säger man att Alice privata nyckel är (d, n)). Bob ska skicka meddelandet M till Alice, som han gör om till ett heltal m med en metod som Alice och Bob har kommunicerat öppet om. Han beräknar

$$c = m^e \pmod{n}$$

och skickar c till Alice.
Alice bestämmer m genom att beräkna $c^d \pmod{n}$. Resultatet blir precis m Eftersom

$$c^d = (m^e)^d = m^{ed} = m^{1-a\cdot\phi(n)} = m^1 \cdot m^{-a\cdot\phi(n))}$$

$$= m \cdot (m^{\phi(n)})^{-a} = m \cdot 1^{-a} = m \pmod{n}.$$

Själva beräkningen $c^d \pmod{n}$ är snabb om man utnyttjar potenslagarna effektivt.
Varför kan inte Eve dekryptera meddelandet? Eve har tillgång till n, e och c.
Det skulle räcka för henne att beräkna d för att knäcka meddelandet. Att beräkna $d(= e^{-1})$ är lätt om man känner till $\phi(n)$ (som Alice gör). Men det är beräkningstungt att bestämma $\phi(n)$ om man inte känner till primtalsfaktoriseringen av n.
Och primtalsfaktorisering är, till skillnad från primtalstestning, svårt. Att primtalsfaktorisering är svårt är inte något som är bevisat, och om någon skulle finna en snabb faktoriseringsalgoritm skulle RSA knäckas.
För fortsatt läsning om RSA, se engelskspråkiga Wikipedia. Där finns även längre exempel som illustrerar metoden. Vi avslutar genom att visa ett exempel där det framgår hur man kan göra för att bestäma den multiplikativa inversen med hjälp av Euklides algoritm.

Exempel 1. *Beräkna* $7^{-1} \pmod{\phi(77)}$.

Eftersom $77 = 7 \cdot 11$ *så är* $\phi(77) = \phi(7) \cdot \phi(11) = 6 \cdot 10 = 60$.
Euklides algoritm för att beräkna SGD(60,7) ger oss

$$60 = 7 \cdot 8 + 4$$

$$7 = 4 \cdot 1 + 3$$

$$4 = 3 \cdot 1 + 1$$

vilket vi använder för att lösa ekvationen $1 = 7 \cdot a + 60 \cdot b$*:*
$1 = 4 - 3 \cdot 1$, $1 = 4 - (7 - 4 \cdot 1) \cdot 1$, $1 = -7 + 4 \cdot 2$, $1 = -7 + (60 - 7 \cdot 8) \cdot 2$, $1 = 7 \cdot (-17) + 60 \cdot 2$.

Eftersom $-17 \equiv_{60} 43$ *så är* $7^{-1} = 43$ *i* $\mathbb{Z}_{60}$*. Detta kan givetvis kontrolleras genom att beräkna* $7 \cdot 43 = 301$ *vilket är kongruent med* 1 *modulo* 60*.*

Följande exempel är taget från Wikipedia.

Exempel 2. *Alice väljer* $p = 61$*,* $q = 53$ *och beräknar* $n = 61 \cdot 53 = 3233, \phi(n) = 60 \cdot 52 = 3120$*. Alice väljer sedan* $e = 17$ *(som är relativt prima 3120). Hon beräknar* $d = e^{-1} = 2753$ *i* Z_{3120}*.*
Alice publika nyckel är (3233,17) och Alices privata nyckel är (3120, 2753).
Bob vill skicka meddelandet 65 till Alice och beräknar $65^{17} = 2790$ *i* Z_{3233} *som han sänder öppet till Alice.*
Alice dekrypterar meddelandet till 65 *genom att beräkna* 2790^{2753} *i* Z_{3233}*.*