RSA

lehký úvod

V roce 1977 pánové Rivest, Shamir a Adleman vypracovali algoritmus RSA. Jedná se o dvouklíčový algoritmus (asymetrickou šifru), čili jsou nutné dva klíče - jeden pro šifrování a jeden pro dešifrování. Při znalosti pouze jednoho klíče tedy můžeme pouze šifrovat nebo pouze dešifrovat. Neprolomitelnost spočívá ve vysokých nárocích na čas potřebný k rozložení velkého čísla na jeho prvočinitele (proč tak vyplyne z následujícího textu).

předpokládané znalosti

V dalším popisu budu předpokládat základní znalosti *modulární aritmetiky*. Dále budu automaticky předpokládat znalost *převodu z decimální soustavy do soustavy binární*.

pár výpočetních postupů do začátku

hledání modula vysoké mocniny

Nejprve se musíme naučit efektivně počítat v matematice "modulo" (neboli *efektivně počítat zbytek po dělení*). Pro malá čísla to není nic obtížného, pohybujeme-li se však v číslech velkých, je tato operace velice zdlouhavá. Musíme se naučit rychle spočítat číslo

$$Y = X^s \bmod r \quad \text{kde} \quad X < r$$

Na to existuje následující algoritmus:

1) převedu číslo s do binární soustavy (dále, budu-li mluvit o číslu s, budu ho již považovat za převedené na binární)

2) vezmu číslo s_0 (cifru, jež se vyskytuje na !!*poslední*!! pozici v číslu s). Je-li $s_0 = 1$, pak číslo $A = X$, jinak $A = 1$

3) vypočítám číslo $Y_1 = X^2 \bmod r$

4) pokud číslo $s_1 = 1$ (s_1 je na předposlední pozici), pak položím $A = A * Y_1 \bmod r$, pokud $s_1 = 0$, pak nedělám nic

5) opakuju předchozí dva kroky (s tím, že v kroku 3 již nepočítám $Y_n = X^2 \bmod r$, ale $Y_n = Y_{n-1}^2 \bmod r$, a v kroku 4 již nepočítám $A = A * Y_1 \bmod r$, ale $A = A * Y_n \bmod r$) dokud neprojdu všechny cifry čísla s

6) v tuto chvíli mám v A hledanou hodnotu

příklad: číslo Y = 45^93 mod 107 (= X^s mod r)

1) převedu číslo s do binární soustavy: (odteď číslo s beru v binární soustavě) s = **1011101**

2) s0=1, pak A=45

3 + 4) Y1=X^2 mod r=45^2 mod 107=99; s1=0 => A=45

3 + 4) Y2=Y1^2 mod r=99^2 mod 107=64; s2=1 => A=A*Y2 mod r=45*64 mod 107=98

3 + 4) Y3=Y2^2 mod r=64^2 mod 107=30; s3=1 => A=A*Y3 mod r=98*30 mod 107=51

3 + 4) Y4=Y3^2 mod r=30^2 mod 107=44; s4=1 => A=A*Y4 mod r=51*44 mod 107=104

3 + 4) Y5=Y4^2 mod r=44^2 mod 107=10; s5=0 => A=104

3 + 4) Y6=Y5^2 mod r=10^2 mod 107=100; s6=1 => A=A*Y6 mod r=104*100 mod 107=21

5) prošli jsme tedy všechny cifry s a nyní máme v A uloženo číslo A=Y=45^93 mod 107=**21**

Eulerova funkce + další důležitá potřebná teorie

$$\varphi(x)=(p_1-1)*p_1^{k_1-1}*..*(p_n-1)*p_1^{k_n-1}=..$$

$$..=x*(1-1/p_1)*..*(1-1/p_n)$$

, kde $p_1^{k_1}*..*p_n^{k_n}$ je rozklad čísla **x** na prvočinitele

dále , když číslo **a** je nesoudělné s číslem **b**, tak platí $a^{\varphi(b)}=1\ mod\ b$

dále z Eulerovy věty platí $a^{-1}=a^{\varphi(b)-1}\ mod\ b$

princip šifrování RSA

1) měj textový řetězec, obsahující nějaké znaky nějaké abecedy. Tato abeceda nechť má své znaky ohodnoceny navzájem různými číselnými konstantami pro různé znaky.

2) Text k zašifrování převeď do číselné formy. Takovýto číselný řetězec rozděl do bloků stejné délky (poslední blok doplň nulami před první cifru do požadované délky).

3) Zvol číslo N, které je součinem dvou velkých prvočísel **p** a **q** (v praxi 500 a více místných), čili **N = p*q**. Dále **x** nechť vyjadřuje pořadí každého jednoho číselného bloku. Navíc ožadujeme, aby každé toto vyjádření splňovalo podmínku **1<=x<N**

4) Zvol tzv. šifrovací exponent **s**, nesoudělný s funkcí **f(N)**, definovanou jako **f(N) = (p-1)*(q-1)**, čili chceme aby jejich *největší společný dělitel* (těchto čísel **s** a **f(N)**) byl **1**)

5) Dále najde číslo **t** (tzv. *dešifrovací exponent*), takové, že **t*s = 1 mod f(N)**, čili **t*s = 1 mod (p-1)*(q-1)**. Protože největší společný dělitel **s** a **f(N)** je **1**, pak má tato pouze jediné řešení.

6) Každý účastník komunikace oznámí dvojici **(N,s)**, kde **s** naýváme v*eřejný klíč*

7) Nyní můžeme šifrovat pomocí funkce $y = x^s \ mod \ N$ (tím zašifruji nešifrovaný blok **x** do šifrovaného vyjádření bloku **x**, značeného **y**). a dešifrovat pomocí funkce $x = y^t \ mod \ N$ (tím dešifruji šifrované vyjádření bloku **x**, tedy **y** do původního tvaru, tedy do původního bloku **x**, a z těchto bloků již snadno pomocí tabulky ohodnocení odvodím původní text).

příklad: (pouze ilustrační, není vhodné papírově ilustrovat silnější šifrování, neboť jen samotné vypsání jednoho z prvočísel by zabralo několik řádků. Ukážu tedy příklad značně zjednodušený.)

1) Budu chtít zašifrovat řetězec "**Ahojda**".

Pro naše znaky v řetězci "Ahojda" to buď například následující

ZNAK	*OHODNOCENÍ*	*UPRAVENÉ OHODNOCENÍ*	
A	3	=>	03
h	56	=>	56
o	98	=>	98
j	9	=>	09
d	99	=>	99
a	11	=>	11

(v praxi je dobré číslovat například podle ASCII kódování)

ps.: je nutné *přečíslovat* si znaky tak, aby všechny obsahovaly *stejný počet cifer* (proč, to vyplyne při samotném šifrování a dešifrování - nyní to ponecháme stranou)

2) nyní mám číselný řetězec ve tvaru [**035698099911**] vyjadřující původní textový řetězec

zvolíme si *délku bloku*: nechť je například 3 (poslední blok doplním nulami nazačátek, pokud mi délka nevyjde) a daný řetězec převedeme do bloků
{x1=[035], x2=[698], x3=[099], x4=[911]}

3) zvolím číslo N - zvolím tedy například prvočísla p a q takto:
p = 37
q = 43
dopočítám **N**: **N = p*q = 37*43 = 1591**
spočítám **f(N)**: **f(N) = (p-1)*(q-1) = 36*42 = 1512**

4) najdu číslo **s** tak, aby nejmensi spolecny delitel **s** a **f(N)** byl **1** (tedy **s** a **f(N)** musí být *nesoudělná* - předpokládám znalost hledání nesoudělných čísel)
ve zkratce **f(N)=2*2*2*3*3*3*7** => **s** mohu zvolit například **17** (mohu ale *libovolně*).

5) nyní potřebuji najít dešifrovací exponent **t**. Musím najít řešení **t*s = 1 mod f(N)**, čili řešení kongruence **t*17 = 1 mod 1512**

pro vyřešení využiji Eulerovy věty a Eulerovy funkce

vypočítám
$$\varphi(f(N))=\varphi(1512)=1512+(1-1/2)*(1-1/3)*(1-1/7)=432$$

chceme řešení pro $t*17=1\,mod\,1512$, tedy $t=(17^{-1})\,mod\,1512$
(!!!! pohybujeme se v modulární aritmetice a je nutné s číslem 17^{-1} zacházet opatrně)

z Eulerovy věty spočítáme s^{-1} (čili 17^{-1} v modulární aritmetice)
$s^{-1}=s^{\varphi(f(N))-1}\,mod\,f(N)=..$ (podle našeho předem předpočítaného)
$..=s^{432-1}\,mod\,1512=s^{431}\,mod\,1512=..$ (podle v úvodu ukázaného způsobu počítání mocnin v modulo) = **89**

6) Strany si nyní zašlou své klíče.

7) nyní začnu šifrovat (zašifruji **{x1=[035], x2=[698], x3=[099], x4=[911]}**)
(bloky o méně, než 3 cifrách nyní musím ale doplnit na 3 ciferné: *takto oxx* apod.. - stejný princip budu praktikovat při dešifrování)
035^17 mod 1591 = 907 => 907 = y1
698^17 mod 1591 = 52 => 052 = y2
099^17 mod 1591 = 669 => 669 = y3
911^17 mod 1591 = 82 => 082 = y4

a nyní mám zašifrovaný text [**907052669082**]

nyní text opět dešifruji (dešifruji **y1 ... yn**) - nezapomenu opět doplnit nuly, kde je třeba

907^89 mod 1591 = 35 => 035 = x1
052^89 mod 1591 = 698 => 698 = x2
669^89 mod 1591 = 99 => 099 = x3
082^89 mod 1591 = 911 => 911 = x4

převedu zpět na číselný řetězec, z nějž již podle tabulky ohodnocení dané abecedy jednoduše vyhodnotím původní text

Problém šifrování pomocí RSA

Algoritmus není 100% správný. Není totiž znám algoritmus pro přesné vygenerování dostatečně velkého prvočísla v reálném čase. Řešení spočívá ve využití algoritmů typu "Monte Carlo" (dokáží vygenerovat nesmírně velké prvočíslo, ale pořád mají malou pravděpodobnost, že se zmílí). Pokud se nám ovšem podaří na počátku opravdu vybrat 2 velká prvočísla, je algoritmus korektní.

Rychlost algoritmu není efektivní. Pro přenášení velkého objemu šifrovaných dat je lepší si pomocí algoritmu vyměnit pouze klíče k jiným algoritmům (nejlépe symetrickým).

Proč je algoritmus považován za bezpečný?

Je nutno chránit pouze privátní (dešifrovací klíč) a veřejný klíč může být opravdu veřejný, protože pomocí veřejného klíče není možné zprávu dešifrovat, pouze zašifrovat. Pokud dostanu žádost o navázání komunikace obousměrně šifrované, pak se naskytnou dvě možnosti: buď navazující strana pošle svůj RSA veřejný klíč a já zašlu můj a komunikujeme pomocí dvou RSA kanálů. Nebo mi navazující strana pošle RSA klíč a já jí na oplátku pošlu klíč k nějaké jiné (symetrické šifře) pomocí jejího RSA klíče.

Z veřejného klíče je sice možné odvodit klíč privátní, ale ani nejlepší metody toho nejsou schopny dosáhnout v použitelném čase (myšleno dny, měsíce, roky a více). Nejlepším známým typem útoku je faktorizace, založená na principu Malé fermatovy věty. Ovšem dodnes se šifru RSA nepodařilo prolomit.