A multifactor RSA-like scheme

Nadir Murru

joint work with Emanuele Bellini

Università di Trento, Dipartimento di Matematica

Lo schema RSA

Generazione delle chiavi

- si scelgono due numeri primi (grandi) p, q e si calcola $N = pq$;
- si sceglie un intero e tale che $gcd(e, (p-1)(q-1) = 1$.
  La coppia (N, e) è la *chiave pubblica* o di *criptazione*;
- si calcola $d = e^{-1} \pmod{(p-1)(q-1)}$.
  La tripla (p, q, d) è la *chiave privata* o di *decriptazione*.

Criptazione

Possiamo criptare un messaggio in chiaro $m \in \mathbb{Z}_N^*$. Il messaggio cifrato è $c = m^e \pmod{N}$.

Decriptazione

Si recupera il messaggio in chiaro calcolando $c^d \pmod{N}$.

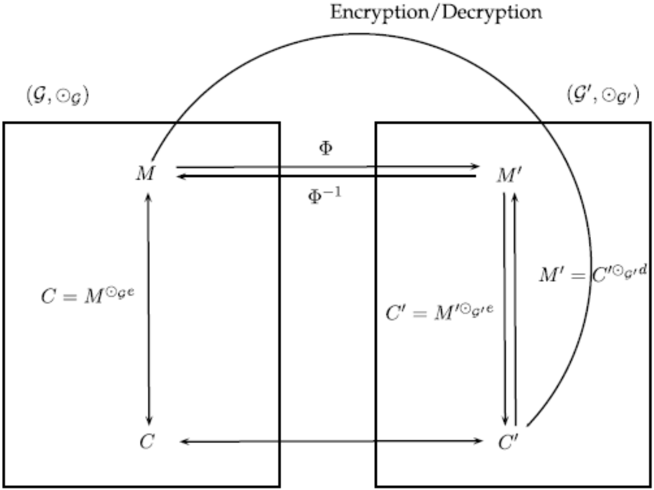
Lo schema RSA

- Fattorizzare N
- Calcolo della radice discreta
- Attacchi che sfruttano alcune debolezze di RSA e della sua implementazione
- Ottimizzare i tempi di cifratura e decifratura

Extension to multifactor modulus

- There exists variants of RSA scheme which exploit a modulus with more than 2 factors to achieve a faster decryption algorithm.
- This variants are sometimes called Multifactor RSA or Multiprime RSA.
- The first proposal exploiting a modulus of the form $N = p_1 p_2 p_3$ has been patented by Compaq in 1997.
- About at the same time Takagi (1998) proposed an even faster solution using the modulus $N = p^r q$, for which the exponentiation modulo p^r is computed using the Hensel lifting method.
- Later, this solution has been generalized to the modulus $N = p^r q^s$

RSA–like cryptosystems



The Pell equation

The Pell equation is

$$x^2 - Dy^2 = 1$$

for D a non–square integer and we wanto to find integer solutions. It arises from the Archimede's cattle problem

"Compute, O friend, the number of the cattle of the sun which once grazed upon the plains of Sicily, divided according to color into four herds, one milk-white, one black, one dappled and one yellow. The number of bulls is greater than the number of cows, and the relations between them are as follows: etc..."

The Brahamagupta product:

$$(x_1, y_1) \otimes (x_2, y_2) = (x_1 x_2 + D y_1 y_2, x_1 y_2 + x_2 y_1).$$

RSA–like cryptosystems

- RSA protocol based on the Pell equation, Lemmermeyer 2006
- RSA–like scheme based on isomorphism between conics and $\mathbb{Z}_N^*$, Padhye et al. 2006–2013
- RSA–like scheme based on Brahamagupta–Bhaskara equation, Thomas et al. 2011–2013
- RSA type cryptosystem based on cubic curves, Koyama et al. 1995–2017

The Pell equation from an algebraic point of view

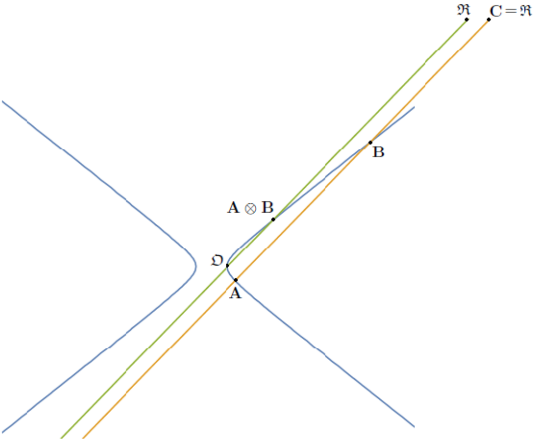
If we consider $\mathbb{Q}[\sqrt{D}] \simeq \mathbb{Q}[t]/(t^2 - D)$, the Brahmagupta product is the product of this **quadratic field**:

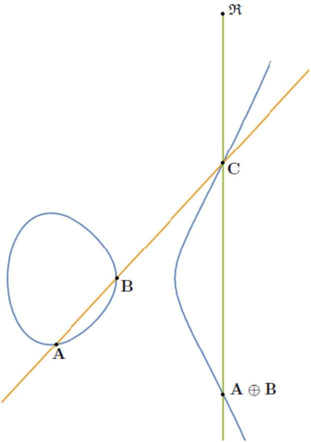
$$(a + bt)(c + dt) = ac + bdt^2 + (ad + bc)t = ac + bdD + (ad + bc)t.$$

The **norm** of an element $x + yt$ is

$$N(x + yt) = (x + yt)(x - yt) = x^2 - Dy^2.$$

The Pell conic





A construction of the group of the parameters

We can get a group $(P, \odot)$ using the following parametrization for the Pell conic

$$y = \frac{1}{m}(x+1)$$

which yields isomorphisms Φ and Φ^{-1} between $(\mathcal{C}, \otimes)$ and $(P, \odot)$

Remark

The above parametrization can be also obtained in an algebraic way considering $\mathbb{A} = \mathbb{F}[x]/(x^2 - D)$ and then $P = \mathbb{A}^*/\mathbb{F}^*$

A construction of the group of the parameters

This construction allows us to define the set $P = \mathbb{F} \cup \{\alpha\}$, with α not in $\mathbb{F}$, equipped with the following product:

$$\begin{cases} a \odot b = \dfrac{D + ab}{a + b}, \quad a + b \neq 0 \\ a \odot b = \alpha, \quad a + b = 0 \end{cases}.$$

We have that $(P, \odot)$ is a commutative group with identity α and the inverse of an element a is the element b such that $a + b = 0$.

Proposition

If $\mathbb{F} = \mathbb{Z}_p$, then $\mathbb{A} = GF(p^2)$ and $B = \mathbb{A}^*/\mathbb{F}^*$ has order $p + 1$. Thus, an analogous of the Fermat's little theorem holds in P:

$$z^{\odot(p+2)} \equiv z \pmod{p}, \quad \forall z \in P.$$

Generalization

Conic	Parameter	Product
$x^2 - Dy^2 = \ell,\ \ell = u^2$	$m = \dfrac{x+u}{y}$	$m_A \odot m_B = \dfrac{m_A m_B + D}{m_A + m_B}$
$x^2 - Dy^2 = \ell,\ \ell \neq u^2$	$m = \dfrac{y-\beta}{x-\alpha}$	$m_A \odot m_B = \dfrac{(Dm_A m_B + 1)\alpha - (m_A + m_B)\beta D}{(-(Dm_A m_B + 1)\beta + (m_A + m_B)\alpha)D}$
$y = ex^2 + k$	$m = (x+\alpha)e$	$m_A \odot m_B = -2\alpha e + m_A + m_B$

Rédei rational functions

The powers in P can be efficiently computed by means of the Rédei rational functions. They arise from the development of

$$(z+\sqrt{d})^n = N_n(d,z) + D_n(d,z)\sqrt{d},$$

for any integer $z \neq 0$, d non–square integer. The Rédei rational functions are defined as

$$Q_n(d,z) = \frac{N_n(d,z)}{D_n(d,z)}, \quad \forall n \geq 1.$$

Remark

The Rédei rational functions can be evaluated by means of an algorithm of complexity $O(\log_2(n))$ with respect to addition, subtraction and multiplication over rings, More 1995.

Rédei rational functions

Proposition

We have

$$Q_{n+m}(D,z) = Q_n(D,z) \odot Q_m(D,z).$$

Corollary

Let $z^{\odot n} = \underbrace{z \odot \cdots \odot z}_{n}$ *be the* n*–th power of* z *with respect to the product* $\odot$*. Then*

$$z^{\odot n} = Q_n(d,z)\ .$$

Algorithms

Direct(m, n)

if $m = 0$ **return** ∞

Set L, c_j s.t. $n = \sum_{j=1}^{L} c_j 2^{j-1}$

/ Pre-computation:

$x_1 = m$

for $j = 2, \ldots, L$

$x_j = x_{j-1}^{\odot 2}$

/ Exponentiation:

$y_1 = \infty$

for $j = 1, \ldots, L$

if $c_j = 1$ $y_{j+1} = y_j \odot x_j$

else $y_{j+1} = y_j$

return y_{L+1}

More(m, n)

if $m = 0$ or $n = 0$ **return** ∞

Set L, c_j s.t. $n = \sum_{j=1}^{L} c_j 2^{j-1}$

$R_1 = m$

for $j = 1, \ldots, L-1$

$$R_{j+1} = \frac{R_j^2 + b}{2R_j + a}$$

if $c_{L-j} = 1$

$$R_{j+1} = \frac{mR_{j+1} + b}{R_{j+1} + m + a}$$

return R_{L+1}

Modified_More(m, n)

if $m = 0$ or $n = 0$ **return** ∞

Set L, c_j s.t. $n = \sum_{j=1}^{L} c_j 2^{j-1}$

$A_1 = m, B_1 = 1$

for $j = 1, \ldots, L-1$

$A_{j+1} = A_j^2 + bB_j$

$B_{j+1} = 2A_jB_j + aB_j^2$

if $c_{L-j} = 1$

$A' = A_{j+1}, B' = B_{j+1}$

$A_{j+1} = mA' + bB'$

$B_{j+1} = A' + (m + a)B'$

return A_{L+1}/B_{L+1}

More			Modified More		
P	A	I	P	A	I
$2(L+w-2)$	$3(L-1)+2(w-1)$	$L+w-2$	$5(L-1)+3(w-1)$	$3(L-1)+2(w-1)$	1

Pell hyperbola over rings

Theorem

The Pell equation $x^2 - Dy^2 = 1$ has $p^{r-1}(p+1)$ solutions in $\mathbb{Z}_{p^r}$ for $D \in \mathbb{Z}^*_{p^r}$ quadratic non–residue in $\mathbb{Z}_p$.

Theorem

Let p, q be prime numbers and $N = p^r q^s$, then for all $(x, y) \in \mathcal{C}$ we have

$$(x, y)^{\otimes p^{r-1}(p+1)q^{s-1}(s+1)} \equiv (1, 0) \pmod N$$

for $D \in \mathbb{Z}^*_N$ quadratic non–residue in $\mathbb{Z}_p$ and $\mathbb{Z}_q$.

Pell hyperbola over rings

Corollary

Let $p_1, \dots, p_r$ be primes and $N = p_1^{e_1} \cdot \ldots \cdot p_r^{e_r}$, then for all $(x, y) \in \mathcal{H}_{\mathbb{Z}_{p^r}}$ we have

$$(x, y)^{\otimes \Psi(N)} = (1, 0) \pmod{N},$$

where

$$\Psi(N) = p_1^{e_1 - 1}(p_1 + 1) \cdot \ldots \cdot p_r^{e_r - 1}(p_r + 1),$$

for $D \in \mathbb{Z}_N^*$ quadratic non–residue in $\mathbb{Z}_{p_i}$, for $i = 1, \dots, r$.

As a consequence, we have an analogous of the Euler theorem also for the product $\odot$, i.e., for all $m \in \mathbb{Z}_N^*$ the following holds

$$m^{\odot \Psi(N)} = \alpha \pmod{N},$$

Key generation

- choose r prime numbers $p_1, \ldots, p_r$, r odd integers $e_1, \ldots, e_r$ and compute $N = \prod_{i=1}^{r} p_i^{e_i}$;
- choose an integer e such that $\gcd(e, \Psi(N)) = 1$;
- evaluate $d = e^{-1} \pmod{\Psi(N)}$.

The public or encryption key is given by (N, e) and the secret or decryption key is given by $(p_1, \ldots, p_r, d)$.

A scheme with multifactor modulus

Encryption

We can encrypt pair of messages $(M_x, M_y) \in \mathbb{Z}_N^* \times \mathbb{Z}_N^*$.

- compute $D = \dfrac{M_x^2 - 1}{M_y^2} \pmod{N}$;
- compute $M = \Phi(M_x, M_y) = \dfrac{M_x + 1}{M_y} \pmod{N}$;
- compute the ciphertext $C = M^{\odot e} \pmod{N} = Q_e(D, M) \pmod{N}$

Notice that not only C, but the pair (C, D) must be sent through the insecure channel.

A scheme with multifactor modulus

Decryption

- compute $C^{\odot d} \pmod{N} = Q_d(D, C) \pmod{N} = M$;
- compute $\Phi^{-1}(M) = \left(\frac{M^2 + D}{M^2 - D}, \frac{2M}{M^2 - D} \right) \pmod{N}$ for retrieving the messages (M_x, M_y).

A scheme with multifactor modulus

Thus, our scheme can be also exploited when $N = p_1^{e_1} \cdot \ldots \cdot p_r^{e_r}$. It can be attacked by solving one of the following problems:

1. factorizing the modulus $N = p_1^{e_1} \cdot \ldots \cdot p_r^{e_r}$;
2. computing $\Psi(N) = p_1^{e_1 - 1}(p_1 + 1) \cdot \ldots \cdot p_r^{e_r - 1}(p_r + 1)$, or finding the number of solutions of the equation $x^2 - Dy^2 \equiv 1 \mod N$, i.e. the curve order, which divides $\Psi(N)$;
3. computing Discrete Logarithm problem either in $(\mathcal{C}, \otimes)$ or in $(P, \odot)$;
4. finding the unknown d in the equation $ed \equiv 1 \mod \Psi(N)$;
5. finding an impossible group operation in P;
6. computing M_x, M_y from D.

Efficiency

- The appropriate number of primes to be chosen in order to resist state-of-the-art factorization algorithms depends from the modulus size, and, precisely, it can be: up to 3 primes for 1024, 1536, 2048, 2560, 3072, and 3584 bit modulus, up to 4 for 4096, and up to 5 for 8192.
- When $r = 2$ our scheme is two times faster than RSA, as it has already been shown. If $r = 3$ our scheme is 4.5 time faster, with $r = 4$ is 8 times faster, and with $r = 5$ is 12.5 times faster.

Thank you for the attention!