Lattice Reduction Techniques To Attack RSA

David Wong

University of Bordeaux

March 2015

RSA ?

(e, N) is the **public key**, (d, N) is the **private key**.

(e, N) is the **public key**, (d, N) is the **private key**.

To **encrypt** a message m, with $m < N$ we just do :

$$c = m^e \pmod N$$

(e, N) is the **public key**, (d, N) is the **private key**.

To **encrypt** a message m, with $m < N$ we just do :

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

And to **decrypt** :

$$m = c^d \pmod{N}$$

To generate these keys, we first generate **two primes** p and q such that :

$$N = p \times q$$

To generate these keys, we first generate **two primes** p and q such that :

$$N = p \times q$$

Use p and q to generate the pair **private key/public key** (d, e).

ATTACKS ?

On the implementation or on the mathematics.

On the implementation or on the mathematics.

Model :

- Recover the plaintext $m^e = c \pmod{N}$
- Recover the private key d

On the implementation or on the mathematics.

Model :

- Recover the plaintext $m^e = c \pmod{N}$
- Recover the private key d

Relaxed model :

On the implementation or on the mathematics.

Model :

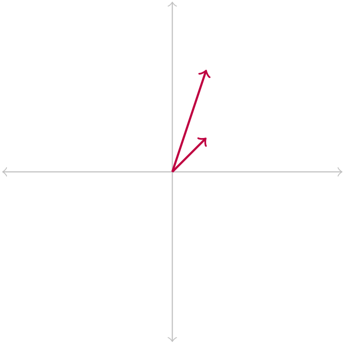
- Recover the plaintext $m^e = c \pmod{N}$
- Recover the private key d

Relaxed model :

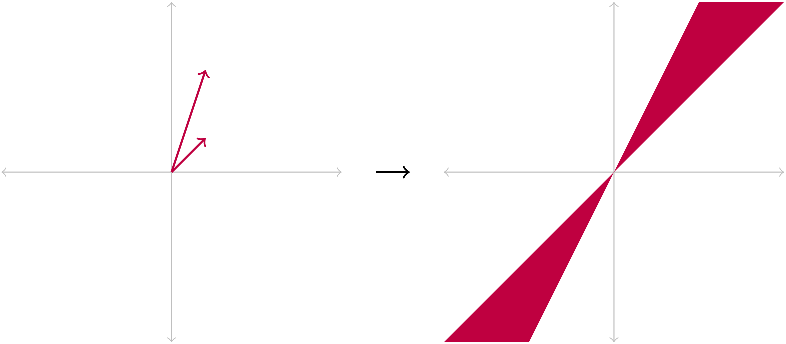
- We know a part of the message
- We know an approximation of one of the prime
- The private exponent is too small

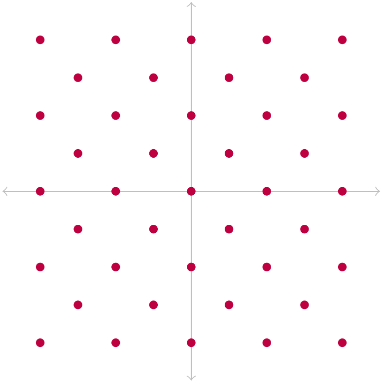
LATTICE ?

A bit like a **vector space**.



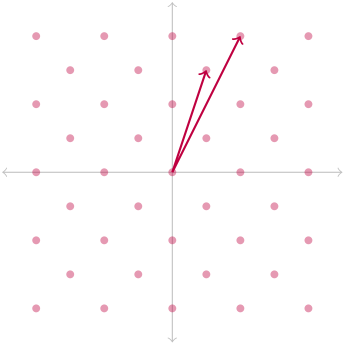
A bit like a **vector space**.



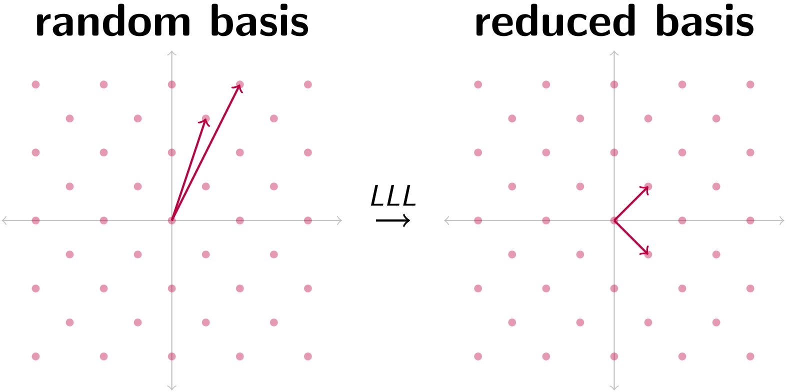


LLL, a lattice basis reduction algorithm

random basis



LLL, a lattice basis reduction algorithm



$$B = \begin{pmatrix} \vec{b_1} \\ \vdots \\ \vec{b_n} \end{pmatrix} \xrightarrow{\textbf{LLL}} B' = \begin{pmatrix} \vec{b'_1} \\ \vdots \\ \vec{b'_n} \end{pmatrix}$$

$$B = \begin{pmatrix} \vec{b_1} \\ \vdots \\ \vec{b_n} \end{pmatrix} \xrightarrow{\quad\textbf{LLL}\quad} B' = \begin{pmatrix} \vec{b'_1} \\ \vdots \\ \vec{b'_n} \end{pmatrix}$$

$$\|b'_1\| \leq \|b'_2\| \leq \ldots \leq \|b'_i\| \leq 2^{\frac{n(n-1)}{4(n+1-i)}} \cdot det(L)^{\frac{1}{n+1-i}}$$

COPPERSMITH ?

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

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

$$m = m_0 + x_0$$

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

$$m = m_0 + x_0$$

“le mot de pass du jour est : cupcake”

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

$$m = m_0 + x_0$$

“le mot de pass du jour est : cupcake”

$$c = (m_0 + x)^e \pmod{N}$$

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

$$m = m_0 + x_0$$

"le mot de pass du jour est : cupcake"

$$c = (m_0 + x)^e \pmod{N}$$

$$f(x) = c - (m_0 + x)^e \pmod{N}$$

$$f(x) = 0 \pmod{N} \text{ with } |x| < X$$

$$\downarrow$$

$$g(x) = 0 \text{ over } \mathbb{Z}$$

Howgrave-Graham :
Let $g(x)$ be an univariate polynomial with n monomials. Further, let m be a positive integer. Suppose that

Howgrave-Graham :
Let $g(x)$ be an univariate polynomial with n monomials. Further, let m be a positive integer. Suppose that

$$g(x_0) = 0 \pmod{N} \quad \text{where} \quad |x_0| \leq X \tag{1}$$

Howgrave-Graham :
Let $g(x)$ be an univariate polynomial with n monomials. Further, let m be a positive integer. Suppose that

$$g(x_0) = 0 \pmod{N} \quad \text{where} \quad |x_0| \le X \tag{1}$$

$$\|g(xX)\| < \frac{N}{\sqrt{n}} \tag{2}$$

Howgrave-Graham :
Let $g(x)$ be an univariate polynomial with n monomials. Further, let m be a positive integer. Suppose that

$$g(x_0) = 0 \pmod{N} \quad \text{where} \quad |x_0| \le X \tag{1}$$
$$\|g(xX)\| < \frac{N}{\sqrt{n}} \tag{2}$$

Then $g(x_0) = 0$ holds over the integers.

Howgrave-Graham :
Let $g(x)$ be an univariate polynomial with n monomials. Further, let m be a positive integer. Suppose that

$$g(x_0) = 0 \pmod{N^m} \quad \text{where} \quad |x_0| \le X \tag{1}$$

$$\|g(xX)\| < \frac{N^m}{\sqrt{n}} \tag{2}$$

Then $g(x_0) = 0$ holds over the integers.

$$f(x_0) = 0 \pmod{N} \text{ with } |x_0| < X$$

$$g(x_0) = 0 \pmod{N^m}$$
$$\|g(xX)\| < \frac{N^m}{\sqrt{n}}$$

$$g(x_0) = 0 \text{ over } \mathbb{Z}$$

LLL reduction :

- It only does **integer linear operations** on the basis vectors
- The **shortest vector of the output basis is bound** (as seen in **Property 1**)

$$B = \begin{pmatrix} \vec{b_1} \\ \vdots \\ \vec{b_n} \end{pmatrix} \xrightarrow{\quad\textbf{LLL}\quad} B' = \begin{pmatrix} \vec{b'_1} \\ \vdots \\ \vec{b'_n} \end{pmatrix}$$

$$\|b'_1\| \leq \|b'_2\| \leq \ldots \leq \|b'_i\| \leq 2^{\frac{n(n-1)}{4(n+1-i)}} \cdot det(L)^{\frac{1}{n+1-i}}$$

$$B = \begin{pmatrix} \vec{b_1} \\ \vdots \\ \vec{b_n} \end{pmatrix} \xrightarrow{\quad\textbf{LLL}\quad} B' = \begin{pmatrix} \vec{b_1'} \\ \vdots \\ \vec{b_n'} \end{pmatrix}$$

$$\|b_1'\| \leq \|b_2'\| \leq \ldots \leq \|b_i'\| \leq 2^{\frac{n(n-1)}{4(n+1-i)}} \cdot det(L)^{\frac{1}{n+1-i}}$$

$$\|b_1'\| \leq 2^{\frac{n(n-1)}{4(n)}} \cdot det(L)^{\frac{1}{n}}$$

$$
\begin{aligned}
g_{i,j}(x) &= x^j \cdot N^i \cdot f^{m-i}(x) \\
&\quad \text{for } i = 0, \ldots, m-1,\; j = 0, \ldots, \delta - 1 \\
h_i(x) &= x^i \cdot f^m(x) \\
&\quad \text{for } i = 0, \ldots, t-1
\end{aligned}
$$

$$
\begin{aligned}
g_{i,j}(x) &= x^j \cdot N^i \cdot f^{m-i}(x) \\
&\quad \text{for } i = 0, \ldots, m-1,\ j = 0, \ldots, \delta - 1 \\
h_i(x) &= x^i \cdot f^m(x) \\
&\quad \text{for } i = 0, \ldots, t-1
\end{aligned}
$$

Those polynomials achieve two things :

- they have the **same root** x_0 but modulo N^m

$$
\begin{aligned}
g_{i,j}(x) &= x^j \cdot N^i \cdot f^{m-i}(x) \\
&\quad \text{for } i = 0, \ldots, m-1,\; j = 0, \ldots, \delta - 1 \\
h_i(x) &= x^i \cdot f^m(x) \\
&\quad \text{for } i = 0, \ldots, t-1
\end{aligned}
$$

Those polynomials achieve two things :

- they have the **same root** x_0 but modulo N^m
- each iteration introduce a new polynomial

$$f(x_0) = 0 \pmod{N} \text{ with } |x_0| < X$$

$$g(x_0) = 0 \pmod{N^m}$$

$$\|g(xX)\| < \frac{N^m}{\sqrt{n}}$$

$$g(x_0) = 0 \text{ over } \mathbb{Z}$$

$$
\begin{tikzcd}
f(x_0) = 0 \pmod{N} \text{ with } |x_0| < X \arrow[dr] \arrow[dddd] & \\
& \text{generate } f_i \text{ s.t. } f_i(x_0) = 0 \pmod{N^m} \arrow[d] \\
& B = \begin{pmatrix} f_i(xX) \\ \vdots \\ f_n(xX) \end{pmatrix} \arrow[d, "\text{LLL}"] \\
& B' = \begin{pmatrix} b_1 = g(xX) \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \arrow[d] \\
g(x_0) = 0 \text{ over } \mathbb{Z} & g(x_0) = 0 \pmod{N^m} \text{ and } \|g(xX)\| < \frac{N^m}{\sqrt{n}} \arrow[l]
\end{tikzcd}
$$

Let $g(x)$ be an univariate polynomial with n monomials. Further, let m be a positive integer. Suppose that

$$g(x_0) = 0 \pmod{b^m} \quad \text{where} \quad |x_0| \le X \tag{1}$$

$$\|g(xX)\| < \frac{b^m}{\sqrt{n}} \tag{2}$$

Then $g(x_0) = 0$ holds over the integers.

$$|\tilde{p} - p| < N^{\frac{1}{4}}$$

Now we have an equation with one unknown, modulo another unknown :

$$\tilde{p} = x_0 \pmod{p}$$

Coppersmith Theorem

Let N be an integer of unknown factorization, which has a divisor $b \geq N^{\beta}$, $0 < \beta \leq 1$. Let $f(x)$ be a univariate monic polynomial of degree δ and let $c \geq 1$.

Then we can find in time $\mathcal{O}(c\delta^5 log^9(N))$ all solutions x_0 of the equation

$$f(x) = 0 \pmod{b} \quad \text{with} \quad |x_0| \leq c \cdot N^{\frac{\beta^2}{\delta}}$$

BONEH-DURFEE ?

Recall how RSA works :

$$\begin{aligned} & e \cdot d = 1 \pmod{\varphi(N)} \\ \Longrightarrow\ & e \cdot d = k \cdot \varphi(N) + 1 \\ \Longrightarrow\ & k \cdot \varphi(N) + 1 = 0 \pmod{e} \\ \Longrightarrow\ & k \cdot (N + 1 - p - q) + 1 = 0 \pmod{e} \end{aligned}$$

Howgrave-Graham :
Let $g(x)$ be an bivariate polynomial with at most n monomials. Further, let m be a positive integer. Suppose that

$$g(x_0, y_0) = 0 \pmod{e^m} \quad |x_0| \leq X \text{ and } |y_0| \leq Y \tag{1}$$

$$\|g(xX, yY)\| < \frac{e^m}{\sqrt{n}} \tag{2}$$

Then $g(x_0, y_0) = 0$ holds over the integers.

Boneh and Durfee proposed a construction of the f_i polynomials as follow :

for $k = 0, \ldots, m$:

$$g_{i,k}(x) = x^i \cdot f^k(x,y) \cdot e^{m-k} \text{ for } i = 0, \ldots, m -$$

$$h_{j,k}(x) = y^j \cdot f^k(x,y) \cdot e^{m-k} \text{ for } j = 0, \ldots, t$$

$$
\begin{array}{c}
 \\
e^2 \\ xe^2 \\ fe \\ x^2e^2 \\ xfe \\ f^2 \\ ye^2 \\ yfe \\ yf^2
\end{array}
\begin{array}{c}
\begin{array}{ccccccc}
1 & x & xy & x^2 & x^2y & x^2y^2 & y
\end{array} \\
\begin{pmatrix}
e^2 & & & & & & \\
 & e^2X & & & & & \\
e & eAX & eXY & & & & \\
 & & & e^2X^2 & & & \\
 & eX & & eAX^2 & eX^2Y & & \\
1 & 2AX & 2XY & A^2X^2 & 2AX^2Y & X^2Y^2 & \\
 & & & & & e^2Y & \\
 & & eAXY & & & & eY \\
 & & 2AXY & & A^2X^2Y & 2AX^2Y^2 & Y
\end{pmatrix}
\end{array}
$$

Boneh-Durfee basis matrix for $m = 2$, $t = 1$

$$
\begin{array}{c}
 \\
e^2 \\ xe^2 \\ fe \\ x^2e^2 \\ xfe
\end{array}
\begin{array}{c}
\begin{array}{ccccccc}
1 & x & xy & x^2 & x^2y & x^2y^2 & y
\end{array} \\
\begin{pmatrix}
e^2 & & & & & & \\
 & e^2X & & & & & \\
e & eAX & eXY & & & & \\
 & & & e^2X^2 & & & \\
 & eX & & eAX^2 & eX^2Y & &
\end{pmatrix}
\end{array}
$$

$$f(x,y) = \underbrace{1 + xy}_{u} + Ax \pmod{e}$$

$$
\begin{array}{cc}
 & \begin{matrix} 1 & x & u & x^2 & ux & u^2 & u^2y \end{matrix} \\
\begin{matrix} e^2 \\ xe^2 \\ \bar{f}e \\ x^2e^2 \\ x\bar{f}e \\ \bar{f}^2 \\ y\bar{f}^2 \end{matrix} &
\begin{pmatrix}
e^2 & & & & & & \\
 & e^2X & & & & & \\
 & eAX & eU & & & & \\
 & & & e^2X^2 & & & \\
 & & & eAX^2 & eUX & & \\
 & & & A^2X^2 & 2AUX & U^2 & \\
 & -A^2X & -2AU & & A^2UX & 2AU^2 & U^2Y
\end{pmatrix}
\end{array}
$$

if $d < N^{0.292}$ we can find.

Conclusions