Attacks against RSA

- Factoring
  - Equivalence between factoring and breaking RSA?
- Mathematical attacks
  - Attacks against plain RSA encryption and signature
  - Heuristic countermeasures
  - Low private / public exponent attacks
  - Provably secure constructions
- Implementation attacks
  - Timing attacks, power attacks and fault attacks
  - Countermeasures
Key generation:
- Generate two large distinct primes $p$ and $q$ of same bit-size.
- Compute $n = p \cdot q$ and $\phi = (p - 1)(q - 1)$.
- Select a random integer $e$, $1 < e < \phi$ such that $\gcd(e, \phi) = 1$
- Compute the unique integer $d$ such that
  \[
e \cdot d \equiv 1 \mod \phi
\]
  using the extended Euclidean algorithm.
- The public key is $(n, e)$. The private key is $d$. 

RSA encryption

- **Encryption**
  - Given a message \( m \in [0, n - 1] \) and the recipient’s public-key \((n, e)\), compute the ciphertext:

  \[
  c = m^e \mod n
  \]

- **Decryption**
  - Given a ciphertext \( c \), to recover \( m \), compute:

  \[
  m = c^d \mod n
  \]
Low private exponent attacks

- To reduce decryption time, one could use a small $d$
  - Wiener’s attack: recover $d$ if $d < N^{0.25}$
- Boneh and Durfee’s attack (1999)
  - Recover $d$ if $d < N^{0.29}$
  - Based on lattice reduction and Coppersmith’s technique
  - Open problem: extend to $d < N^{0.5}$
- Conclusion: devastating attack
  - Use a full-size $d$
To reduce encryption time, one can use a small $e$
  
  For example $e = 3$ or $e = 2^{16} + 1$

Coppersmith’s theorem:
  
  Let $N$ be an integer and $f$ be a polynomial of degree $\delta$. Given $N$ and $f$, one can recover in polynomial time all $x_0$ such that $f(x_0) = 0 \mod N$ and $x_0 < N^{1/\delta}$.

Application: partially known message attack:
  
  If $c = (B||m)^3 \mod N$, one can recover $m$ if $|m| < |N|/3$
  
  Define $f(x) = (B \cdot 2^k + x)^3 - c \mod N$.
  
  Then $f(m) = 0 \mod N$ and apply Coppersmith's theorem to recover $m$. 
Coppersmith’s short pad attack

- Let \( c_1 = (m\|r_1)^3 \mod N \) and \( c_2 = (m\|r_2)^3 \mod N \)
- One can recover \( m \) if \( r_1, r_2 < N^{1/9} \)
- Let \( g_1(x, y) = x^3 - c_1 \) and \( g_2(x, y) = (x + y)^3 - c_2 \).
- \( g_1 \) and \( g_2 \) have a common root \( (m\|r_1, r_2 - r_1) \) modulo \( N \).
- \( h(y) = \text{Res}_x(g_1, g_2) \) has a root \( \Delta = r_2 - r_1 \), with \( \text{deg} \ h = 9 \).
- To recover \( m\|r_1 \), take \( \text{gcd} \) of \( g_1(x, \Delta) \) and \( g_2(x, \Delta) \).

Conclusion:

- Attack only works for particular encryption schemes.
- Low public exponent is secure when provably secure construction is used. One often takes \( e = 2^{16} + 1 \).
Solving $p(x) = 0 \mod N$ when $N$ is of unknown factorization: hard problem.

- For $p(x) = x^2 - a$, equivalent to factoring $N$.
- For $p(x) = x^e - a$, equivalent to inverting RSA.

Coppersmith showed (E96) that finding small roots is easy.

- When $\deg p = \delta$, finds in polynomial time all integer $x_0$ such that $p(x_0) = 0 \mod N$ and $|x_0| \leq N^{1/\delta}$.
- Based the LLL lattice reduction algorithm.

Can be heuristically extended to more variables.
Coppersmith’s algorithm has numerous applications in cryptanalysis:

- Cryptanalysis of plain RSA when some part of the message is known:
  - If \( c = (B + x_0)^3 \mod N \), let \( p(x) = (B + x)^3 - c \) and recover \( x_0 \) if \( x_0 < N^{1/3} \).

- Factoring \( n = p^r q \) for large \( r \) (Boneh and al., C99).

Applications in provable security:

- Improved security proof for RSA-OAEP with low-exponent (Shoup, C01).
Solving \( p(x) = 0 \mod N \)

- Find a small linear integer combination \( h(x) \) of the polynomials:
  - \( q_{ik}(x) = x^i \cdot N^{\ell-k}p^k(x) \mod N^\ell \)
  - For some \( \ell \) and \( 0 \leq i < \delta \) and \( 0 \leq k \leq \ell \).
  - \( p(x_0) = 0 \mod N \Rightarrow p^k(x_0) = 0 \mod N^k \Rightarrow q_{ik}(x_0) = 0 \mod N^\ell \).
  - Then \( h(x_0) = 0 \mod N^\ell \).
- If the coefficients of \( h(x) \) are small enough:
  - Then \( h(x_0) = 0 \) holds over \( \mathbb{Z} \).
  - \( x_0 \) can be found using any standard root-finding algorithm.
Illustration with a polynomial of degree 2:
- Let $p(x) = x^2 + ax + b \mod N$.
- We must find $x_0$ such that $p(x_0) = 0 \mod N$ and $|x_0| \leq X$.

We are interested in finding a small linear integer combination of the polynomials:
- $p(x)$, $Nx$ and $N$.
- Then $h(x_0) = 0 \mod N$.

If the coefficients of $h(x)$ are small enough:
- Then $h(x_0) = 0$ also holds over $\mathbb{Z}$,
- which enables to recover $x_0$. 

Jean-Sébastien Coron
Cryptography
Howgrave-Graham lemma

Given \( h(x) = \sum h_i x^i \), let \( \| h \|^2 = \sum h_i^2 \).

Howgrave-Graham lemma:

Let \( h \in \mathbb{Z}[x] \) be a sum of at most \( \omega \) monomials. If \( h(x_0) = 0 \mod N \) with \( |x_0| \leq X \) and \( \| h(xX) \| < N/\sqrt{\omega} \), then \( h(x_0) = 0 \) holds over \( \mathbb{Z} \).

Proof:

\[
|h(x_0)| = \left| \sum h_i x_0^i \right| = \left| \sum h_i X^i \left( \frac{x_0}{X} \right)^i \right|
\]

\[
\leq \sum \left| h_i X^i \left( \frac{x_0}{X} \right)^i \right| \leq \sum \left| h_i X^i \right|
\]

\[
\leq \sqrt{\omega} \| h(xX) \| < N
\]

Since \( h(x_0) = 0 \mod N \), this gives \( h(x_0) = 0 \).
Illustration of HG lemma
The coefficients of $h(xX)$ must be small:

- $h(xX)$ is a linear integer combination of the polynomials

\[
\begin{align*}
  p(xX) &= X^2 \cdot x^2 + aX \cdot x + b \\
  q_1(xX) &= NX \cdot x \\
  q_2(xX) &= N
\end{align*}
\]

We must find a small integer linear combination of the vectors:

- $[X^2, aX, b], [0, NX, 0]$ and $[0, 0, N]$

Tool: LLL algorithm.
Lattice and lattice reduction

We must find a small linear integer combination \( h(xX) \) of the polynomials \( p(xX), xXN \) and \( N \).

Let \( L \) be the corresponding lattice, with a basis of row vectors:

\[
\begin{bmatrix}
   X^2 & aX & b \\
   NX &   & N
\end{bmatrix}
\]

Using LLL, one can find a lattice vector \( b \) of norm:

\[
\|b\| \leq 2(\det L)^{1/3} \leq 2N^{2/3} X
\]

Then if \( X < N^{1/3}/4 \), then \( \|h(xX)\| = \|b\| < N/2 \)

Howgrave-Graham lemma applies and \( h(x_0) = 0 \).
Lattice

Definition:
Let \( u_1, \ldots, u_\omega \in \mathbb{Z}^n \) be linearly independent vectors with \( \omega \leq n \). The lattice \( L \) spanned by the \( u_i \)'s is
\[
L = \left\{ \sum_{i=1}^{\omega} n_i \cdot u_i \mid n_i \in \mathbb{Z} \right\}
\]
If \( L \) is full rank \((\omega = n)\), then \( \det L = |\det M| \), where \( M \) is the matrix whose rows are the basis vectors \( u_1, \ldots, u_\omega \).

The LLL algorithm:
The LLL algorithm, given \((u_1, \ldots, u_\omega)\), finds in polynomial time a vector \( b_1 \) such that:
\[
\|b_1\| \leq 2^{(\omega-1)/4} \det(L)^{1/\omega}
\]
The previous bound gives $|x_0| \leq N^{1/3}/4$.

But Coppersmith's bound gives $|x_0| \leq N^{1/2}$.

**Technique**: work modulo $N^k$ instead of $N$.

- Let $q(x) = (p(x))^2$. Then $q(x_0) = 0 \mod N^2$.
- $q(x) = x^4 + a'x^3 + b'x^2 + c'x + d'$.
- Find a small linear combination $h(x)$ of the polynomials $q(x)$, $Nxp(x)$, $Np(x)$, $N^2x$ and $N^2$.
- Then $h(x_0) = 0 \mod N^2$.
- If the coefficients of $h(x)$ are small enough, then $h(x_0) = 0$. 

Jean-Sébastien Coron  Cryptography
Details when working modulo $N^2$

- **Lattice basis**: 
  
  \[
  \begin{bmatrix}
  X^4 & a' X^3 & b' X^2 & c' X & d' \\
  NX^3 & NaX^2 & NbX & & \\
  NX^2 & NaX & Nb & & \\
  N^2 X & & & & \\
  N^2 & & & & 
  \end{bmatrix}
  \]

- Using LLL, one gets:
  
  \[
  \|h(xX)\| \leq 2 \cdot (\det L)^{1/5} \leq 2X^2 N^{6/5}
  \]

  If $X \leq N^{2/5}/6$, then $\|h(xX)\| \leq N^2/3$ and $h(x_0) = 0$. 