Recent Developments in Cryptography: What is "breaking RSA"?

J.H. Davenport

University of Bath

jhd@maths.bath.ac.uk

Conventions

- Use classical arithmetic, e.g. multiplying n-bit integers takes $O(n^2)$.

- Vector spaces generally have dimension d.

- $||\mathbf{a}|| = \sqrt{\sum_{i=1}^{d} a_i^2}$; $||\sum_{i=0}^{d} a_i x^i|| = \sqrt{\sum_{i=0}^{d} a_i^2}$.

- "choose" means "chooses carefully, avoiding many well-known weaknesses".

RSA (as encryption)

B chooses p, q primes, and e coprime to $(p-1)(q-1)$; computes $N = pq$ and D such that $De \equiv 1 \pmod{(p-1)(q-1)}$; publishes n-bit N and e.

$$\begin{array}{ccc} A & \longrightarrow & B \\ c \equiv m^e \pmod{N} & c & m \equiv c^D \pmod{N} \\ & \downarrow & \\ & E & \end{array}$$

A and B have tasks $O(n^3)$.
What about E?

E's options

$$
\begin{array}{cl}
\text{factoring } N & \\
\Updownarrow & O(n^2) \\
\phi(N) = (p-1)(q-1) & \\
\Downarrow & O(n^2) \\
\text{computing } D & \\
\Downarrow & O(n^3) \\
\text{computing } m \text{ from } c &
\end{array}
$$

But there *might* be other ways, e.g. an $O(n^6)$ means of computing roots directly.

“breaking RSA” = “factoring”?

- Define “completely breaking RSA” as having an algorithm which, given N, e and c, computes m.

- Then this is no harder than factoring: it may in fact be easier.

- “Breaking how system X uses RSA” depends very much on system X: if system X always chooses $e = 1$, it is trivial to break.

- Focus on case of “small’ e, e.g. $e = 3$

\+ used when A has less compute power than B.

One scenario (Coppersmith)
almost known-plaintext

- The message is “withdraw £L from this account”, and I know everything (e.g. from a discarded receipt) except the PIN z.

- Assume (purely for notational simplicity) the PIN is at the end of the message.

- Then I know m_0 (message if $z = 0$) and have to solve $(m_0 + z)^e \equiv c \pmod{N}$.

Lattices (simply)

- A lattice L is a discrete additive subgroup of $\mathbf{R}^d$.

- Assume $L \subseteq \mathbf{Z}^d$ and $\dim L = d$.

- L is generated by a **basis** $(b_1, \ldots, b_d)$: $b_i \in \mathbf{Z}^d$.

- Many bases generate the same L, but

  $$\det(L) = |\det(b_1, \ldots, b_d)|$$

  is an invariant.
  Example: $(2, 5)$ and $(3, 8)$ generate a lattice of determinant 1, but this lattice can also be generated by $(0, 1)$ and $(1, 0)$.

Short vectors (theoretically)

Define $\lambda_1(L)$ to be $\min_{\substack{v\neq 0\\ v\in L}} ||v||$.

Define $\lambda_2(L) = \displaystyle\min_{\substack{\dim\langle v_1,v_2\rangle=2\\ v_1,v_2\in L}} \max(||v_1||, ||v_2||)$, and so on.

Minkowski: $\lambda_1(L) \leq \sqrt{d}(\det(L))^{1/d}$ and, more generally,

$$\prod_{i=1}^{r} \lambda_i(L) \leq d^{r/2}(\det(L))^{r/d}.$$

However, we know no efficient ways to compute these.

Gram-Schmidt orthogonalisation
of basis $(b_1, \ldots, b_d)$.

Define (recursively)

- $b_i^* = b_i - \sum_{j=1}^{i-1} \mu_{i,j} b_j^*$;

- $\mu_{i,j} = (b_i \cdot b_j^*)/||b_j^*||^2$.

The (b_i^*) form an orthogonal basis for the same lattice, but not, normally, over $\mathbf{Z}$.

Lenstra–Lenstra–Lovács reduction

The basis $(b_1, \ldots, b_d)$ is LLL-reduced if (were one to apply Gram-Schmidt to it!)

1. $|\mu_{i,j}| \leq \frac{1}{2}$, $1 \leq j < i \leq d$
   (else reduce b_i by an integral multiple of b_j)

2. $||b_i^* + \mu_{i,i-1} b_{i-1}^*||^2 \geq \frac{3}{4} ||b_{i-1}^*||^2$
   (else swap b_i and b_{i-1})

This process terminates in polynomial time:

$$O(d^6 \log^3 \max |b_{i,j}|).$$

What properties?

- Certainly not canonical!
- $||b_1|| \le 2^{(d-1)/4} \det(L)^{1/d}$.
- $||b_i|| \le 2^{(d-1)/2} \lambda_i(L)$.
- $\prod_{i=1}^{d} ||b_i|| \le 2^{d(d-1)/4} \det(L)$.
- Often we do much better than this.
- Many variants/improvements.
- Still a black art.

Small modular roots (Coppersmith; Howgrave-Graham)

Given a monic $p(x)$ of degree k, with a root x_0 mod N, with $|x_0| \leq X$.
For i in $1 \ldots hk$, define $v = \lfloor (i-1)/k \rfloor$ and $u = (i-1) - kv$, and

$$q_i(x) = q_{u,v}(x) = N^{h-1-v} x^u (p(x))^v.$$

Note $q_i(x_0) \equiv 0 \pmod{N^{h-1}}$.
Treat the coefficients of each q_i (multiplying the coefficent of x^j by X^j) as entries of a (lower-triangular) $(hk)^2$ matrix M.

An example

$$x^2 + 14x + 19 \equiv 0 \pmod{35}.$$

Take $X = 2$ and $h = 3$.

$$\begin{pmatrix} N^2 & xN^2 & Np & Nxp & p^2 & xp^2 \end{pmatrix}^T =$$

$$\begin{pmatrix} 35^2 & & & & & \\ 0 & 35^2 \cdot 2 & & & & \\ 35 \cdot 19 & 35 \cdot 14 \cdot 2 & 35 \cdot 2^2 & & & \\ 0 & 35 \cdot 19 \cdot 2 & 35 \cdot 14 \cdot 2^2 & 35 \cdot 2^3 & & \\ 361 & 532 \cdot 2 & 234 \cdot 2^2 & 28 \cdot 2^3 & 2^4 & \\ 0 & 361 \cdot 2 & 532 \cdot 2^2 & 234 \cdot 2^3 & 28 \cdot 2^4 & 2^5 \end{pmatrix}$$

Theory (continued)

Since M is lower triangular,

$$\det(M) = X^{hk(hk-1)/2} N^{hk(h-1)/2}.$$

If b is the first vector in LLL-reduced M, then

$$||b|| \le 2^{(hk-1)/4} X^{(hk-1)/2} N^{(h-1)/2}.$$

Dividing b_j by X^{j-1} (which we can do) gives a polynomial $r(x)$ over $\mathbf{Z}$, an integral multiple of the q_i, so $r(x_0) \equiv 0 \pmod{N^{h-1}}$.
For $|x| \le X$, $|r(x)| \le \sum |b_i| \le \sqrt{hk}||b||$.
So $|r(x)| \le \left(2^{(hk-1)/4}\sqrt{hk}\right) X^{(hk-1)/2} N^{(h-1)/2}$.
Choosing

$$X = \left\lceil \left(2^{-1/2}(hk)^{-1/(hk-1)}\right) N^{(h-1)/(hk-1)} \right\rceil - 1$$

gives $|r(x)| < N^{h-1}$. Hence $r(x_0) = 0$.

Example (continued)

This matrix reduces to

$$\begin{pmatrix} 3 & 8\cdot 2 & -24\cdot 2^2 & -8\cdot 2^3 & -1\cdot 2^4 & 2\cdot 2^5 \\ 49 & 50\cdot 2 & 0 & 20\cdot 2^3 & 0 & 2\cdot 2^5 \\ 115 & -83\cdot 2 & 4\cdot 2^2 & 13\cdot 2^3 & 6\cdot 2^4 & 2\cdot 2^5 \\ 61 & 16\cdot 2 & 36\cdot 2^2 & -16\cdot 2^3 & 3\cdot 2^4 & 4\cdot 2^5 \\ 21 & -37\cdot 2 & -14\cdot 2^2 & 2\cdot 2^3 & 14\cdot 2^4 & -4\cdot 2^5 \\ -201 & 4\cdot 2 & 33\cdot 2^2 & -4\cdot 2^3 & -3\cdot 2^4 & 1\cdot 2^5 \end{pmatrix}$$

so $r(x) = 2x^5 - x^4 - 8x^3 - 24x^2 + 8x + 3$, and $x_0 = 3$.

$x_0 > X$, but LLL sometimes over-performs.

Observations

- Can remove first row and column from M.

- Finds solutions up to $O(N^{(h-1)/(hk-1)}) \to N^{1/k}$.

- Time $O(h^9 k^6 \log^3 N)$.

- For $k = 3$ we get

$$\begin{array}{lllll|l} h=2 & h=3 & h=4 & h=5 & h=6 & h=67 \\ N^{0.2} & N^{0.25} & N^{0.27} & N^{0.286} & N^{0.294} & N^{0.33} \end{array}$$

- Since one normally uses with small $d = hk$, the sub-optimality of LLL w.r.t d is not a major problem.

RSA — Mostly Known Messages

Suppose e is small (say $e = 3$) and the message m is mostly known, say $m = m_0 + m_1$ with $|m_1| < N^{1/e}$.
Then solving $(m_0 + m_1)^e - c \equiv 0 \pmod{N}$ gives m_1.
Works if m_1 is not at end of m, say $m = m_0 + 2^s m_1$.
Note that this does not "break N" in the sense of finding the decrypting exponent D.
Some heuristic extensions for case where unknown part is in more than one piece.

RSA — Mostly Similar Messages

Suppose e is small (say $e = 3$) and the message m_2 is mostly similar to message m_1. So $m_1^e = c_1$ and $(m_1 + \delta)^e = c_2$.
Using resultants (for $e = 3$)

$$\delta^9 + 3(c_1 - c_2)\delta^6 + 3(c_1^2 + 7c_1c_2 + c_2^2)\delta^3 + (c_1 - c_2)^3$$

$$\equiv 0 \pmod{N}$$

Note this is an equation in δ^3, and should be solved as such.
Finds $|\delta| < N^{1/e^2}$.

$$\gcd_z(z^3 - c_1, (z + \delta)^3 - c_2) = z - m_1.$$

Note that increasing N increases δ findable.

IP packets encrypted with RSA

Assume (denial of service attack) that we can force A's higher-level protocols to replay. This replay will differ in IP identification (16 bits) and in the effect of that on the checksum). Have to solve

$$m^e = c_1; (m + 2^{48}\delta - \delta - \pm 65535)^e = c_2$$

(where the ± 65535 factor allows for the possibility of wrap-round in the checksum): again an e-degree equation in m^e.

Difficulty of solving

$e = 3$ 512-bit RSA trivial: < 1 second.

1024-bit RSA: < 10 seconds.

e=5 512-bit RSA needs $h = 4$: 20×20 lattice with 2k-bit entries.

δ **in 11 bits** $h = 2$, 10×10 lattice with 1k-bit entries.

1024-bit RSA: 10×10 lattice with 2k-bit entries.

Factoring $N = p^r q$
Boneh, Durfee, Howgrave-Graham

Choose (enumerate) P close to p, and solve for $(P + x)^r \equiv 0 \pmod{p^r}$.
p^r is unknown, but divides N.
Time $\exp\left(\frac{c+1}{r+c} \log p\right) r^{15} \log^3 N$,
where $c = \log q / \log p$.
Similar to ECM and NFS when $r = \sqrt{\log p}$.