9. Public Key Encryption (RSA)

9.1 A Side Channel attack against RSA

Recall RSA public key encryption:

\[ n = p \cdot q, \quad p \neq q \text{ prime} \]
\[ d \in \mathbb{Z}^\times \mathbb{F}(n), \quad \text{i.e., } \gcd(d, (p-1)(q-1)) = 1 \]

Public key \((e = d^{-1} \pmod n)\), private key \((d \mid p, q)\)

Encryption: \(c = m^e \pmod n\)

Decryption: \(m = c^d \pmod n\)

(Most) expensive computations in RSA are exponentiation. Use the Chinese Remainder Theorem (CRT) to speed up decryption.

(i) Compute \(m_1 = c^d \pmod p\)

Since \(c^{p-1} \equiv 1 \pmod p\), compute \(m_1 = c^d \pmod{(p-1)} \pmod p\)

(ii) Compute \(m_2 = c^d \pmod{(q-1)} \pmod q\) (analogously)

(iii) Determine \(m\) such that

\[ m \equiv m_1 \pmod p \]
\[ m \equiv m_2 \pmod q \]

Solution: By the extended Euclidean Algorithm (EEA) compute \(a, b\)

with \(a \cdot p + b \cdot q = 1\), i.e., \(a \equiv p^{-1} \pmod q\), \(b \equiv q^{-1} \pmod p\) (This computation is necessary only once.)

\[ m = (a \cdot q \cdot m_1 + b \cdot p \cdot m_2) \pmod n \]

By (RT) \(m = c^d \pmod n\) is unique.
This method is approximately 4 times faster as direct computation, since number size is of approx. half bit length, half the number of squarings are needed for $SQ M$ square-and-multiply, with complexity $O(n^2)$ for squaring (multiplication), and it is done twice.

Attacks against smartcards with this implementation by random hardware faults causing incorrect values.

By e.g., high temperature, irregular clock frequency or voltage, radiation, magnetic, power draw, cause exactly one false computation.

Available are

- correct deciphering with $m_1, m_2$ from (i), (ii)
- faulty deciphering with error in $\widehat{m}_n$ of (i)

\[ n = \widehat{m}_n = a \cdot q (m_1 - m_2) + b \cdot q (m_2 - m_1) \]

Assumed: $m_1 \neq m_2 \mod p$ (with high probability)

- $\widehat{m}_n \equiv m_2 \mod q$  
- Hence $m \neq \widehat{m} \mod p$, but $m \equiv \widehat{m} \mod q$

\[ \frac{m - \widehat{m}}{q} = \frac{m - \widehat{m}}{q} \]

\[ g = \gcd(m - \widehat{m}, n) = q \]
9.2 Relevance of Cryptography (Repetition)

Like RSA with \( e = 2 \), however \( \exists d \; e \cdot d \equiv 1 \pmod{\phi(n)} \) as \( \gcd(2, \phi(n)) = 2 \)

\( \equiv (p-1)(q-1) \)

\( = \) Deciphering means to calculate square roots \( \pmod{n} \)

But computing square roots is no easier than factoring \( \Rightarrow \) (hard)

Prop 8.3: \( n = p \cdot q \), \( x \) non-trivial solution of \( x^2 \equiv 1 \pmod{n} \)

\( = \) \( \gcd(1 + 1, n) \in \{p, q\} \)

Computing square roots \( \pmod{p}, q \) is easy

Def 9.1: \( c \) is called quadratic residue (QR) \( \pmod{n} \) if

\[ x \quad \text{such that} \quad x^2 \equiv c \pmod{n} \]

Prop 9.2: Euler's criterion

\[ p > 2, \text{prime} \quad c \equiv c \text{ QR } \pmod{p} \quad \Rightarrow c^{(p-1)/2} \equiv 1 \pmod{p} \]

No indication on how to get \( x \).

Prop 9.3: \( p \text{ prime} \quad p \equiv 3 \pmod{4} \quad \Rightarrow \quad p = 4k - 1 \quad \text{QR } \pmod{p} \)

\( = \) \( x^2 \equiv c \pmod{p} \) has the only solutions \( x_{1,2} = \pm c^k \pmod{p} \)

Remark: \( p \equiv 1 \pmod{4} \): no deterministic algo, but polynomial time

prob. algo known.
Railmi's cryptosystem

(i) \( p \neq q \) prime, \( p \cdot q \equiv 3 \pmod{4} \), \( n = p \cdot q \)

(ii) Public key \( n \) private key \((p, q)\)

(iii) Encryption \( c = m^2 \mod n \) for some \( m \in \{1, \ldots, m-1\} \)

(iv) Decryption:

\[
\begin{align*}
\text{Determine } & x^2 \equiv c \pmod{p} \\
& \gamma \equiv c \pmod{q} \\
\text{Determine } & t \equiv x \pmod{p} \\
& \Gamma \equiv \gamma \pmod{q} 
\end{align*}
\]

by CRT (see chapter 9.1)

But there are 4 solutions?  

Note: \( m > \sqrt{n} \), otherwise compute square roots over the reals

Remark 9.8: Need to identify correct solution out of 4

9.6: a) Breaking is the same factoring

b) Vulnerable against chosen-ciphertext
   (analogously to the RSA side-channel attack)

c) Broadcasting: CRT may be applied
   (also for RSA with small e)
9.3 Flipping Coins over the Telephone

Alice and Bob want to decide over the telephone who gets a device offered by a friend. Alice flips a coin. Bob chooses a tail. Alice says “Sorry, it was Heads.” She wins. There are multiple opportunities for cheating. Hence, a secure and fair protocol is sought.

Protocol

A
chooses large primes $p + q = 3 \mod(n)$

$n = p \cdot q$

B
chooses random $x$

$Y = x^2 \mod(n)$

computes 4 square roots of $Y \mod(n)$

$t = a_1 \pm b$

selects randomly $a$ or $b$ (coin flipping)

Say $b$

A wins

If $b = \pm x$

If $b \neq \pm x$, B factors $n$ by $\gcd(x - b, n)$

P & B wins

- 5 -
Security of the cipher Card Protocol

a) $A$ chooses $n$ as a product of more than two primes, $k$ primes.

- There are $2^k$ square roots. (for $2^k - 2$ numbers $B$ is able
to factor $n$)

- $A$ lowers his chances to win, but $B$ could check that by checking for primality, e.g. by M.P.R.

b) $A$ chooses a prime as $n$ =) $B$ could/should test $n$ for primality.

c) $A$ does not choose primes $p \neq q \equiv 3 \pmod{4}$

- She can't calculate square roots or it is more difficult (→ e)

d) $B$ sends a random $y$

i) $y$ is no square =) $A$ does not a find root =) stop: $B$ cheats

ii) $y \not\equiv QR$ =) $A$ finds a root but $B$ cannot factor. $A$ wins.

e) $A$ sends a random number $z$, not a root of $y$, $B$ cannot factor.

- $B$ checks $2^z \equiv y \pmod{n}$, if not =) stop, $A$ cheats (→ c)

f) If $B$ wants to lose, he can do so.
Example

\( A : p = 19, q = 23, n = 437 \rightarrow B \)

\( B : \text{random } x = 112 \)
\( y = 112^2 \mod 437 = 308 \rightarrow A \)

\( A : a = 112, -a = 325 \)
\( b = 135, -b = 302 \)

\( A : \text{Select } b \text{ or } -b \)

\( B : \gcd(b - x = 135 - 112, 437) = 23 \)
\( \gcd(-b - x = 302 - 112, 437) = 19 \)

\( \rightarrow B \text{ wins} \)