9. Public Key Encryption (ctd)

9.1 A Side Channel attack against RSA

Recall RSA public-key encryption:

$n = p \cdot q$, $p \neq q$ prime

$d \in \mathbb{Z}^*_{\varphi(n)}$, i.e., $\gcd(d, \varphi(n) = (p-1)(q-1)) = 1$

Public key $(e = d^{-1}, n)$, private key: d, p, q

Encryption: $c = m^e \bmod n$

Decryption: $m = c^d \bmod n$

(Most) expensive computations in RSA are exponentiations.

Use the Chinese Remainder Theorem (CRT) to speed up decryption.

(i) Compute $m_1 = c^d \bmod p$

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

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

(iii) Determine m such that

$$m \equiv m_1 \bmod p$$

$$m \equiv m_2 \bmod q$$

Solution: By the extended Euclidean Algorithm (EEA) compute a, b with $a p + b \cdot q = 1$, i.e., $a \equiv p^{-1} \pmod{q}$, $b \equiv q^{-1} \pmod{p}$

(This computation is necessary only once.)

$m = (a \cdot q \cdot m_1 + b \cdot p \cdot m_2) \bmod n$

By CRT $m = c^d \bmod n$ is unique

This method is approximately 4 times faster as direct computation, since number are of approx. half bit length, half the number of squarings are needed for SQM (square-and-multiply, with complexity $O(n^2)$ for squaring / multiplication, and it is done twice.

Attack against smartcards with this implementation by random hardware faults causing incorrect values.
By, e.g., high temperature, irregular clock frequency or voltage, radiation, magnetism, power drain: cause exactly one false computation.

Available are

m correct deciphering with m_1, m_2 from (i), (ii)
$\hat{m}$ faulty decryption with error in $\hat{m}_1$ of (i)

$$m - \hat{m} = aq(m_1 - \hat{m}_1) + bp(m_2 - \hat{m}_2)$$

Assumption $m_1 \not\equiv \hat{m}_1 \mod p$ (with high probability)
$m_2 \equiv \hat{m}_2 \mod q$

Hence $m \not\equiv \hat{m} \mod p$, but $m \equiv \hat{m} \mod q$

$p \nmid m - \hat{m}$ $\quad q \mid m - \hat{m}$

$\Rightarrow m - \hat{m} = \ell q$ $\quad \ell \in \mathbb{Z}$ $\quad p \nmid \ell$

$n = p \cdot q$

$\gcd(m - \hat{m}, n) = q$

9.2 Rabin Cryptosystem (Repetition)

Like RSA with $e = 2$, however $\exists\, d : e \cdot d \equiv 1 \pmod{\varphi(n)}$ as $\gcd(2, \underbrace{\varphi(n)}_{=(p-1)(q-1)}) = 2$

$\Rightarrow$ Deciphering means to calculate square roots $\pmod n$

But computing square roots is no easier than factoring ($\rightarrow$ Chapter 8)

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

$$\Rightarrow \gcd(x+1, n) \in \{p, q\}$$

Computing square roots mod p, q is easy

Def 9.1 c is called quadratic residue (QR) mod n if

$$\exists\, x : x^2 \equiv c \pmod n$$

Prop 9.2 Euler's criterion

$p > 2$, prime : c is QR mod $p \Leftrightarrow c^{(p-1)/2} \equiv 1 \pmod p$

No indications on how to get x.

Prop 9.3 p prime, $p \equiv 3 \pmod 4$, i.e., $p = 4k - 1$, c QR mod p

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

Remark: $p \equiv 1 \bmod 4$: no deterministic alg, but polynomial time prob. alg. known

Rabin's cryptosystem

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

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

(iii) Encryption $c = m^2 \bmod n$ for some msg $m \in \{1, \dots, n-1\}$

(iv) Decryption:

Determine
$$x : x^2 \equiv c \bmod p$$
$$y : y^2 \equiv c \bmod q$$
[2 solutions by Prop 9.3

Determine
$$f \equiv x \pmod p$$
$$f \equiv y \pmod q$$

by CRT (see chapter 9.1)

But there are 4 solutions!

Note: $m \gg \sqrt{n}$, otherwise compute square roots over the reals

Remark 9.5: 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 "tails". Alice says "Sorry, it was heads." She wins. There are multiple opportunities for cheating. Hence, a secure and fair protocol is sought.

Protocol

A		B
chooses large primes $p, q \equiv 3 \pmod 4$		
$n = p \cdot q$	$\longrightarrow$	n
		Chooses random x
y	$\longleftarrow$	$y = x^2 \bmod n$
Computes 4 square roots of $y \bmod n$		
$\pm a, \pm b$		
Selects randomly a or b (coin flipping)		
Say, b	$\longrightarrow$	b
A wins	$\longleftarrow$	If $b \equiv \pm x$
		If $b \not\equiv \pm x$, B factors n
p, q B wins	$\longleftarrow$	by $\gcd(x-b, n)$

Security of Flipping Coin 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 $\Rightarrow$ A lowers her chances to win, but B could check that by checking for primality, e.g. by MRPT.

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

c) A does not choose primes $p \neq q \equiv 3 \pmod 4$
$\Rightarrow$ She can't calculate square roots or it is more difficult ($\rightarrow$ e)

d) B sends a random y
  i) y is no square : A does not find a root $\Rightarrow$ stop : B cheats
  ii) y is 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 $z^2 \equiv y \pmod n$, if not $\rightarrow$ stop, A cheats ($\rightarrow$ c))

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

Example

A : $p = 19$, $q = 23$, $\underline{n = 437}$ $\rightarrow$ B

B : random $x = 112$

$Y = 112^2 \bmod 437 = 308 \rightarrow A$

A : $a = 112$, $-a = 325$

$b = 135$, $-b = 302$

A : Selects b or -b

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

or $\gcd(-b - x = 302 - 112, 437) = 19$

$\rightarrow$ B wins

Selects a or -a

B cannot factor.