The security of RSA

Reminder: The public parameters are N, e coprime to (the non-public) $\phi(N)$. The private parameters are p, q such that $N = pq,\ d = e^{-1} \mod \phi(N) = (p-1)(q-1)$.

$$Enc(m, Pk) = m^e \mod N = c$$
$$Dec(c, Sk) = c^d \mod N = m^{ed} \mod N = m$$

In this lecture, we show one attack on RSA (that works for a bad choice of private parameters), and then we discuss the security model that is relevant to public-key encryption (chosen ciphertext attack).

1 The "Low exponent attack"

We will show that if d is chosen too low ($< \frac{1}{3}N^{1/4}$) then there is an efficient algorithm to recover it.

Proposition 1. *Suppose $q < p < 2q$ (a standard assumption) and suppose that $d < \frac{1}{3}N^{1/4}$ (i.e., d is "small"). Then there is an efficient algorithm to compute d.*

Proof. Let h such that $ed = 1 + h(p-1)(q-1)$. We will show that $\frac{e}{N}$ is very close to $\frac{h}{d}$. Then we use the continued fraction expansion of e/N to recover h/d. Finally since $h = d = 1$ (because $ed - h(p-1)(q-1) = 1$) this yields d.

We proceed by bounding $hN - ed$ from above.

$$hN - ed = hN - h\phi(N) - 1 < hN - h\phi(N) = h(N - \phi(N)$$

$$\begin{aligned} N - \phi(N) &= pq - (p-1)(q-1) = pq - (pq - p - q + 1) \\ &= p + q - 1 < 3q < 3N^{1/2} \end{aligned}$$

Moreover $\phi(N)h = ed - 1 < ed < \frac{1}{3}N^{1/4}\phi(N)$. (Since $e < \phi(N),\ d < \frac{1}{3}N^{1/4}$). So $h < \frac{1}{3}N^{1/4}$ and $hN - ed < \frac{1}{3}N^{1/4}3N^{1/2} = N^{3/4}$. We divide by Nd.

$$\frac{h}{d} - \frac{e}{N} < \frac{1}{dN^{1/4}}$$

and since $d < \frac{1}{3}N^{1/4}$,

$$\frac{1}{dN^{1/4}} < \frac{1}{3d} < \frac{1}{3d^2} < \frac{1}{2d^2}$$

So we have

$$\left| \frac{e}{N} - \frac{h}{d} \right| < \frac{1}{2d^2}$$

.

According to a well-known result, the continued fraction expansion of e/N contains h/d. But what is the continued fraction expansion of $n \in \mathbb{R}$? It is a process which goes like this:
1st step: $a_0 = \lfloor n \rfloor,\ n = a_0 + \epsilon_0,\ 0 \leq \epsilon_0 < 1$ So $n \sim a_0 \in \mathbb{Z},\ a_0 = p_0/q_0$

2nd step: $1/\epsilon_0 = a_1 + \epsilon_1$ where $a_1 = \lfloor 1/\epsilon_0 \rfloor,\ 0 \leq \epsilon_1 < 1$ so $n = a_0 + \frac{1}{a_1+\epsilon_1} \sim a_0 + 1/a_1 = p_1/q_1$.

3rd step: $1/\epsilon_1 = a_2 + \epsilon_2$ where $a_2 = \lfloor 1/\epsilon_1 \rfloor,\ 0 \leq \epsilon_1 < 1$ so

$$n = a_0 + \frac{1}{a_1 + \frac{1}{a_2+\epsilon_2}} \sim a_0 + \frac{1}{a_1 + \frac{1}{a_2}} = \frac{p_2}{q_2}$$

etc... The sequence $p_0/q_0,\ p_1/q_1,\ p_2/q_2, ...$ is the continued fraction expansion of n. It may be infinite. If $n = e/N$, there are at most $\log(N)$ different p_i/q_i,and if $|n - h/d| < \frac{1}{2d^2}$, one of them has to be h/d. □

So the attack is the following:

1. Compute the continued fraction expansion $p_0/q_0, p_1/q_1, ...$ of e/N.
2. For each p_i/q_i, hope that $\frac{p_i}{q_i} = \frac{A}{B}$ where (hopefully) $A = h, B = d$. Let $C = \frac{eB-1}{A}$ be a candidate for $\phi(N)$. If C is not an integer, go back to step 1; otherwise move on to step 3.
3. We want to calculate the secret p, q. They are the roots of $(x-p)(x-q)$. If $C = \phi(N)$, then $x^2 - (N - C + 1)X + N = (x-p)(x-q)$ and therefore its roots are the secret divisors of N. If not, go back to step 1.

2 Chosen Plaintext Attacks (and why it is not enough)

The security game we have used so far to modelize the adversary makes the assumption that they are passive:

Challenger	Adversary
encrypts m_0, m	chooses m_0, m. Decides which message was encrypted.

This situation is known as the Chosen Plaintext Attack (CPA), it is the standard test for secret-key encryption (in the one-time key context). It is not enough to assess the security of a public key encryption scheme. In many cases, it makes sense to give the adversary access to a decryption oracle.

Example 1 (Situation where the adversary has the decryption of a chosen ciphertext)**.** *Bob encrypts* $m = to : Alice@gmail.com \mid body \longrightarrow$ *gmail decrypts* m *reads recipient* $\longrightarrow$ *sends body to Alice.*

The adversary sees $c = c_1 \mid c_2$ *where* $c_1 = Enc(to:\ Alice@gamil.com)$*,* c_2 *is the encryption of the body.*

If the adversary wants the decryption of c_2 *he can compute* $c_1' = Enc(to : Adversary@gmail.com)$*. produce* $c' = c_1' \mid c_2$*.*

Send c' *to gmail, and he will receive* $Dec(c_2, Sk)$*.*

3 Chosen Ciphertext Attacks

To account for the possibility that an adversary could have access to a decryption algorithm, we add decryption queries before the adversary chooses m_0, m_1 (phase 1) and after (phase 2).

Challenger		Adversary
		chooses $c_1, \cdots, c_q \in \mathcal{C}$
	$\longleftarrow$	Asks for decryption of the c_i
Sends $\mathrm{Dec}(c_i, k)$	$\longrightarrow$	
	(Phase 1)	
		Chooses m_0, m_1
	$\longleftarrow$	Sends m_0, m_1
Chooses $i \in \{0, 1\}$		
encrypts $c = Enc(m_i)$		
Sends c	$\longrightarrow$	
	(Challenge)	
		Chooses $c'_1, \cdots, c'_q \neq c$
	$\longleftarrow$	Asks for decryption of the c'_i
Sends $Dec(c'_i, k)$	$\longrightarrow$	
	(Phase 2)	
		Decides which m_i was encrypted

Definition 1 (IND-CCA 1 secure)**.** *If the scheme is secure with only Phase 1 (but no Phase 2), we say it is Indistinguishable under the non-adaptive Chosen Ciphertext Attack, and we denote it by IND-CCA 1.*

Definition 2 (IND-CCA 2 secure)**.** *If the scheme is secure with Phase 1 and 2, it is Indistinguishable under the adaptive Chosen Ciphertext Attack, and we denote it by IND-CCA 2.*

Example 2. *Textbook RSA is malleable. It means that without the random padding,* $Enc(m_1, Pk) \cdot Enc(m_2, Pk) = Enc(m_1 m_2, Pk)$*. This is why it cannot achieve IND-CCA 2 with textbook RSA:*

Challenger		*Adversary*
Decrypts c_1	$\longleftarrow$	*chooses* $c_1 \in \mathcal{C}$
$m_1 = Dec(c_1, Sk)$	$\longrightarrow$	*chooses* $m_0 \in \mathcal{M}$
Encrypts	$\longleftarrow$	*sends* m_0, m_1
$c_i = Enc(m_i, Pk)$	$\longrightarrow$	*Compute* $c = c_1 c_i$
Decrypts c	$\longleftarrow$	*Asks for the decryption of* c
$Dec(c, Sk) = m_i m_1$	$\longrightarrow$	*computes* $m_i = \dfrac{m_1 m_i}{m_1}$ *Solves the Problem*