# TD3 :: Attacks on the RSA

Université Sorbonne Paris Nord, M1. Sergey Dovgal
dovgal@lipn.fr

## 1 Prerequisites

**Chinese Remainder Theorem.** Let $n_1, \dots, n_k$ be relatively prime numbers. Then, the system of modular equations

$$\begin{cases} x \equiv a_1 \pmod{n_1}, \\ \dots \\ x \equiv a_k \pmod{n_k} \end{cases}$$

has a unique solution $x \pmod{n_1 \dots n_k}$.

**Exercise 1.** Prove Chinese Remainder Theorem.
*Hint.* Start with the case of two equations. Use generalised Euclid's algorithm.

**Exercise 2.** Find an $x \pmod{5 \cdot 7 \cdot 11}$ such that

$$\begin{cases} x \equiv 1 \pmod{5}, \\ x \equiv 2 \pmod{7}, \\ x \equiv 3 \pmod{11} \end{cases}$$

## 2 Elementary attacks

**Exercise 3.** Suppose that in the RSA scheme with modulus $n$ the value of $\phi(n)$ is known. Find $p$ and $q$.

**Exercise 4.** Show that if $e_1$ and $e_2$ are two different exponents, and the same message $m$ is encrypted with $e_1$ and $e_2$ then it is possible to recover $m$.

**Exercise 5 (Blinding)**. Suppose that we know only public parameters of the cryptosystem, and don't know $(p, q, d)$. We have a message $m \pmod{n}$. The user who has the secret key, refuses to sign the message $m$ because he disagrees with its content. However, if we provide any another innocent-looking message $m'$, which looks like a random noise, he doesn't mind to

sign it. Given $m$, design a message $m' \neq m$ such that from the signature of $m'$ you can recover the signature of $m$.

## 3 Large integer factoring*

**Exercise 6.*** Show that if $e$ and $d$ are both known, then it is possible to efficiently recover the factorisation of $n = pq$.

*Hint.* The solution to this exercise is given in the paper "Twenty Years of Attacks on the RSA Cryptosystem" by Dan Boneh, page 3. There are exactly four solutions to the equation $x^2 \equiv 1 \pmod{p}q$, given by $x = \pm 1 \pmod{p}$, $x = \pm 1 \pmod{q}$. If an element $g$ comprime with $n$ is chosen uniformly at random, one of the elements of the sequence $g^{(de-1)/2^k} \pmod{n}$ is a square root of unity that reveals factorisation of $N$.

## 4 Various possible situations

**Exercise 7.** Let the same message be sent multiple times with different moduli $n_i$, $i = 1, \ldots, k$ and the same exponent $e$. If $m^e < \prod_{i=1}^{k} n_i$ then it is possible to efficiently determine $n$.

*Hint.* Use Chinese Remainder Theorem.

**Exercise 8 (Fermat).** Let $n = pq$ and $|p - q| < cn^{1/4}$. Then, the number $n$ can be efficiently factorised in time $Poly(c)$.

*Hint.* Denote $p = \dfrac{x+y}{2}$, $q = \dfrac{x-y}{2}$. Find $k$ such that $\lfloor 2\sqrt{n} \rfloor + k$ is a perfect square. Show that $k < c^2/2 = 1$.

**Exercise 9.** Factorise `N = 14027259740375753058659671823 29` using the above technique.

**Exercise 10 (Howgrave-Graham's attack).*** RSA is not secure when $e = 3$ or when $e$ is very small.

**Exercise 11*** (Coppersmith).** If $\lfloor \log n/4 \rfloor$ least significant bits of $p$ are known, it is possible to efficiently factor $N$.

**Exercise 12*** (Coppersmith's Short Pad attack).** Padding is a principle of splitting the message into smaller parts, where each part is encrypted with the same RSA parameters, i.e. the same modulo $n$ and the same public exponent $e$. Show that RSA with padding is not secure.

*The solutions to the last three exercises are given in "Twenty Years of Attacks on the RSA Cryptosystem" by Dan Boneh.*