MEM204-Number Theory

11th virtual lecture

Giorgos Kapetanakis

Spring semester 2019-20 - 08/05/2020

University of Crete

The RSA cryptosystem

Introduction

The RSA cryptosystem was introduced R. Rivest, A. Shamir and L. Adleman in 1977 and it is probably the most widespread public-key cryptosystem used today.

Roughly speaking, its security relies on conjectures (that we will see later) and that the mathematical and computer science communities believe to be true.

The setting

In our setting, we have three characters: Alice, Bob and Chuck. Each has his/her purpose:

1. Bob wants to send a message to Alice.
2. Chuck has access their full conversation and to their past conversations.
3. Alice wants to read Bob's message, but without Chuck knowing the content of Bob's message.

Remark

This is the generic setting assumed by public-key cryptosystems. The private-key cryptosystems allow some prior private conversation between Alice and Bob.

The three phases

The RSA cryptosystem consists of three phases:

1. Key generation, performed by Alice.
2. Encryption, performed by Bob.
3. Decryption, performed by Alice.

Key generation

Alice generates her key as follows:

1. She picks two large distinct primes p and q and computes $n = pq$.
2. She computes $\varphi(n) = (p-1)(q-1)$.
3. She randomly chooses some $1 < e < \varphi(n)$, such that $(e, \varphi(n)) = 1$.
4. She finds (i.e., with the euclidead algorithm) $d \equiv e^{-1} \pmod{\varphi(n)}$.

Her public key is (n, e) and her private key is $(d, p, q, \varphi(n))$. She sends her public key to Bob (hence to Chuck as well).

Encryption

Bob wants to send his message m to Alice; here $(m, n) = 1$ and $m < n$. He has just received her public key (n, e) and he computes

$$c \equiv m^e \pmod{n}.$$

The encrypted message is c and he sends it to Alice (hence to Chuck as well).

Decryption

Alice has just received the encrypted message c and wants to read the original message m. Also, in order to perform the decryption she has the private key $(d, p, q, \varphi(n))$ in her disposal. So, she computes

$$M \equiv c^d \pmod{n}.$$

Correctness

Proposition

With the above notation, $M \equiv m \pmod{n}$.

Proof.

By construction, $ed \equiv 1 \pmod{\varphi(n)}$, hence, $ed = 1 + k\varphi(n)$. Thus,

$$M \equiv c^d \equiv (m^e)^d \equiv m^{ed} \equiv m \cdot (m^{\varphi(n)})^k \equiv m \pmod{n},$$

from Euler's theorem. □

Security

In order to study the security of the system, we have to check whether Chuck can retrieve m, from the information he has at his disposal.

Chuck knows n, e and c and wants to compute m, knowing that $m^e \equiv c \pmod{n}$. This computation is known as the RSA problem and about this the following conjecture is believed to be true.

Conjecture

The RSA problem is (computationally) hard.

Security

Perhaps the most obvious way for Chuck to retrieve m, is to first factor n. So lets assume that Chuck (somehow) factors $n = pq$ (so he also knows p and q). Then he can compute:

1. $\varphi(n) = (p-1)(q-1)$.
2. Using the euclidean algorithm (which is fast), he can compute $d \equiv e^{-1} \pmod{\varphi(n)}$.
3. Compute $m \equiv c^d \pmod{n}$.

Security

In the previous slide, we assumed that Chuck can factor n, in other words that he can easily solve the factorization problem, i.e., finding the prime factorization of a composite number. However, about this problem, the following is believed to be true:

Conjecture

The factorization problem is (computationally) hard.

Security

From the previous slides, we easily see that the factorization problem implies the RSA problem. In fact, the following conjecture has been stated.

Conjecture

The factorization problem and the RSA problem are equivalent.

It is worth mentioning however, that there is evidence suggesting that the above does not hold.

An exercise

Exercise

In Chuck's attack in Slide 10, in reality Chuck only needs to know $\varphi(n)$. Prove that knowing $\varphi(n)$ is equivalent to factoring $n = pq$. (Prove that if one knows n and $\varphi(n)$ and knows that $n = pq$ for some primes p and q, then he/she can compute p and q without performing any hard computation, such as the factorization of n.)

Stay home, stay safe!