Rabin's Cryptosystem

COMPUTATIONAL TOOLS FOR PROBLEM SOLVING

Universitat de Lleida

EMIR PAJIĆ

Rabin's Cryptosystem

- The **Rabin cryptosystem** is an asymmetric cryptographic technique, whose security, like that of RSA, is related to the difficulty of factorization

- It has the disadvantage that each output of the Rabin function can be generated by any of four possible inputs

- If each output is a ciphertext, extra complexity is required on decryption to identify which of the four possible inputs was the true plaintext.

History

- The process was published in January 1979 by Michael O. Rabin.
- The Rabin cryptosystem was the first asymmetric cryptosystem where recovering the entire plaintext from the ciphertext could be proven to be as hard as factoring.

Michael Oser Rabin

born September 1, 1931

is an Israeli mathematician

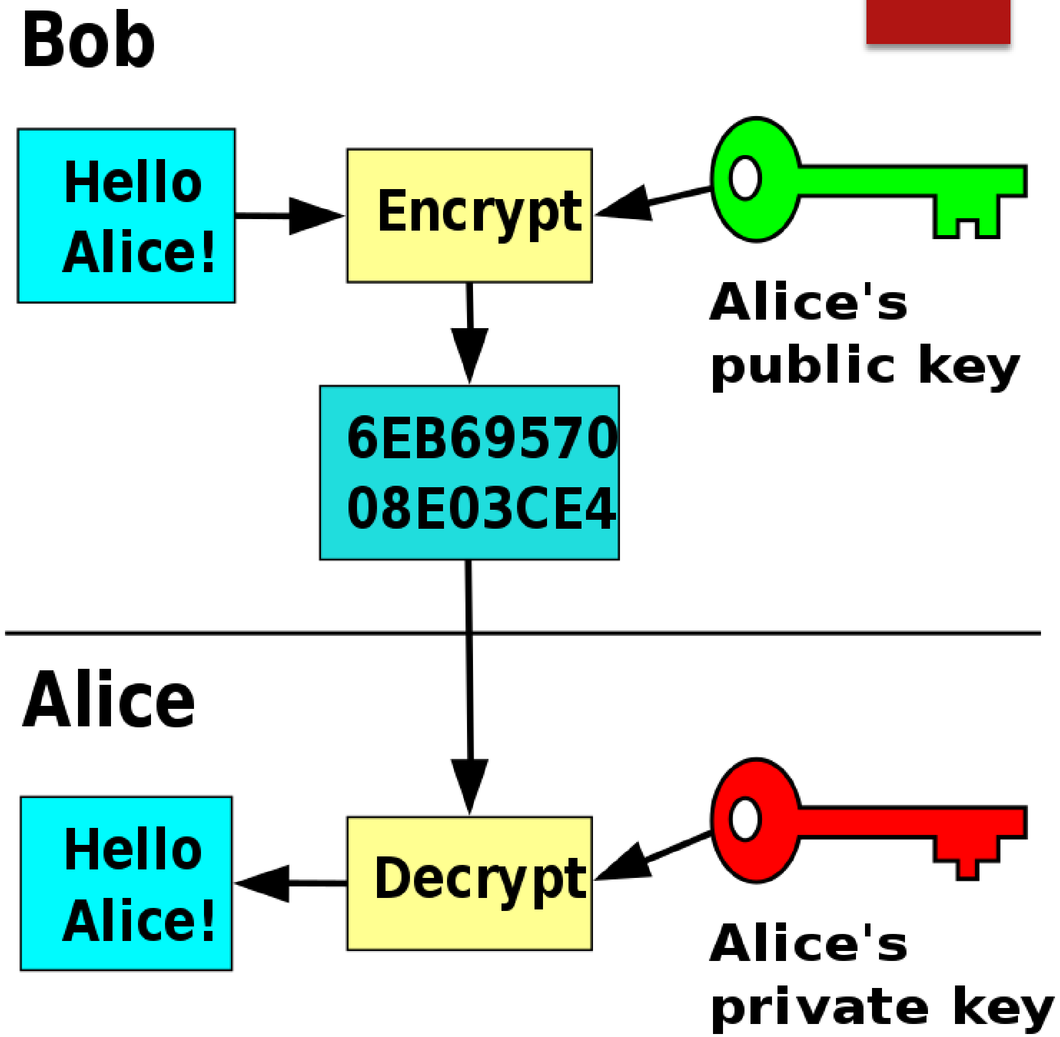
and computer scientist

The **Hebrew University of Jerusalem**

Princeton University, New Jersey

Algorithm

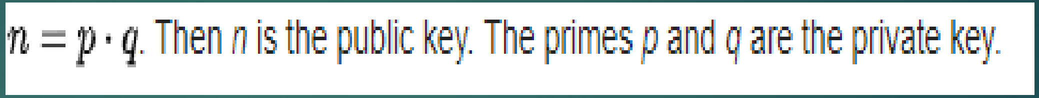
As with all asymmetric cryptosystems, the Rabin system uses both a **public** and a **private** key. The public key is necessary for later encryption and can be published, while the private key must be possessed only by the recipient of the message.



In an asymmetric key encryption scheme, anyone can encrypt messages using the public key, but only the holder of the paired private key can decrypt.

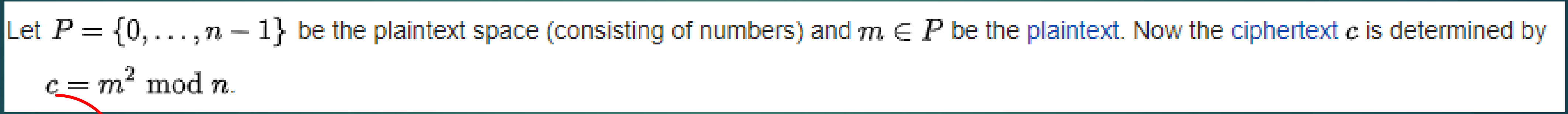
KeyGen & Encryption

- Choose two primes *p* and



$n = p \cdot q$. Then *n* is the public key. The primes *p* and *q* are the private key.

- For the encryption, only the public key *n* is used, thus producing a ciphertext out of the plaintext.



Let $P = \{0, \ldots, n-1\}$ be the plaintext space (consisting of numbers) and $m \in P$ be the plaintext. Now the ciphertext c is determined by

$$c = m^2 \bmod n.$$

- That is, c is the quadratic remainder of the square of the plaintext, modulo the key-number n.
- In cryptography, plaintext or cleartext is unencrypted information
- Ciphertext or cyphertext is the result of encryption performed on plaintext using an algorithm
- For exactly four different values of m, the ciphertext is produced

Decryption

- To efficiently decode the ciphertext, the private keys are necessary.

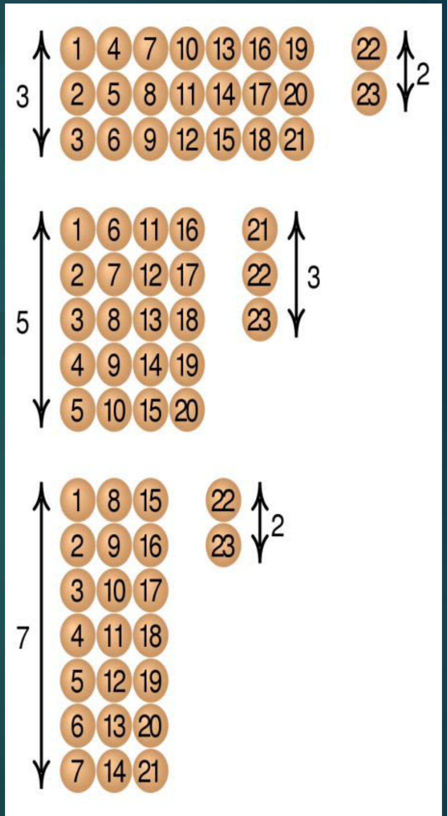
If c and n are known, the plaintext is then $m \in \{0, \dots, n-1\}$ with $m^2 \equiv c \bmod n$.

- For a composite n $n = p \cdot q$
- There is no efficient method known for the finding of m.
- If, however n is prime (or p and q are) the ***Chinese remainder theorem*** can be applied to solve for m.
- The decryption requires to compute square roots of the ciphertext c modulo the primes p and q.

$$m_p = \sqrt{c} \bmod p$$

and

$$m_q = \sqrt{c} \bmod q$$



Sunzi's original formulation

https://en.wikipedia.org/wiki/Chinese_remainder_theorem

Decryption

- By applying the extended Euclidean algorithm, we wish to find y_p and y_q such that $y_p \cdot p + y_q \cdot q = 1.$

- Now, by invocation of the Chinese remainder theorem, the four square roots +r, -r, +s, -s
- The four square roots are in the set {0, … n-1}
- We calculate roots according to this formula

$$\begin{aligned} r &= (y_p \cdot p \cdot m_q + y_q \cdot q \cdot m_p) \bmod n \\ -r &= n - r \\ s &= (y_p \cdot p \cdot m_q - y_q \cdot q \cdot m_p) \bmod n \\ -s &= n - s \end{aligned}$$

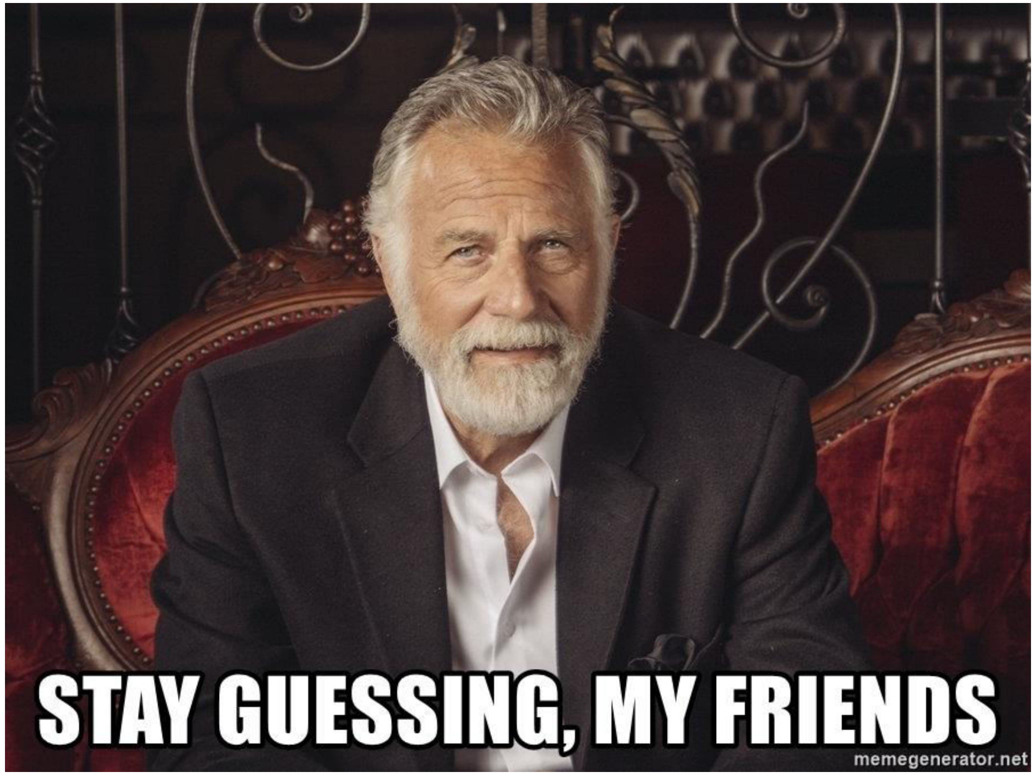
One of these square roots **mod n** is the original plaintext m.

Finding the factorization of **n** is possible, as Rabin pointed out in his paper, if both, **r** and s can be computed as gcd(|r-s|,n) is either p or q

Since the greatest common divisor can be calculated efficiently, the factorization of n can be found efficiently if r and s are known

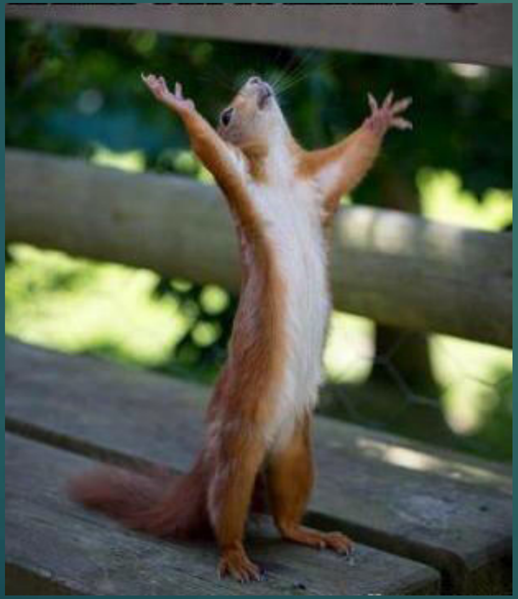
Effectiveness

- Decoding produces three false results in addition to the correct one, so that the correct result must be guessed (r, -r, s, -s).



Effectiveness

- Already mentioned guessing is the major disadvantage of the Rabin cryptosystem and one of the factors which have prevented it from finding widespread practical use.
- If the plaintext is intended to represent a text message, guessing is not difficult; however, if the plaintext is intended to represent a numerical value, this issue becomes a problem that must be resolved by some kind of disambiguation scheme.



It is possible to choose plaintexts with special structures, or to add padding, to eliminate this problem

A way of removing the ambiguity of inversion was suggested by Blum and Williams: the two primes used are restricted to primes congruent to 3 modulo 4 and the domain of the squaring is restricted to the set of quadratic residues. These restrictions make the squaring function into a trapdoor permutation, eliminating the ambiguity.

Efficiency

- For encryption, a square modulo n must be calculated. This is more efficient than RSA, which requires the calculation of at least a cube.

- For decryption, the Chinese remainder theorem is applied, along with two modular exponentiations. Here the efficiency is comparable to RSA.

- Disambiguation introduces additional computational costs, and is what has prevented the Rabin cryptosystem from finding widespread practical use

Security

- The great advantage of the Rabin cryptosystem is that a random plaintext can be recovered entirely from the ciphertext only if the codebreaker is capable of efficiently factoring the public key n.

- It has been proven that decoding the Rabin cryptosystem is equivalent to the integer factorization problem, something that has not been proven for RSA. Thus the Rabin system is 'more secure' in this sense than is RSA, and will remain so until a general solution for the factorization problem is discovered, or until the RSA problem is discovered to be equivalent to factorization. (This assumes that the plaintext was not created with a specific structure to ease decoding.)

Attacks

However, this cryptosystem does not provide indistinguishability against chosen plaintext attacks since the process of encryption is deterministic. An adversary, given a ciphertext and a candidate message, can easily determine whether or not the ciphertext encodes the candidate message (by simply checking whether encrypting the candidate message yields the given ciphertext).

Furthermore, it has been proven an active attacker can break the system using a chosen ciphertext attack (even when challenge messages are chosen uniformly at random from the message space). By adding redundancies, for example, the repetition of the last 64 bits, the system can be made to produce a single root.

Example

Key generation

n = p * q we choose p = 7 and q = 11 (p,q are private keys)

n = 7 * 11

n = 77 – public key

Encryption

We choose our message m = 20 – plaintext

To encrypt our message:

c = m^2 mod n = > c = 400 mod 77 = 15 - ciphertext

Example

Decryption

First we find the square roots m_p and m_q

$m_p = \sqrt{c} \bmod p$ => $m_{p = 1}$

$m_q = \sqrt{c} \bmod q$ => $m_{q= 9}$

Now using extended Eucledian algorithm to finy y_p and $y_{q:}$

$y_p * p + y_q * q = 1$

we get that $y_p = -3$ and $y_q = 2$

Example

- Now to find the four square roots which one of them is correct one:
- We use 4 formulas:

$r = (y_p * p * m_q + y_q * q * m_p\ ) \bmod n$

$r = \ (-3 * 7 * 9 + 2*11*1) \bmod 77$

$r = (-189 + 22) \bmod 77$

$r = 64$

$-r = n-r$

$-r = 77 – 64$

$-r = 13$

$s = (\ y_p * p * m_q – y_q * q * m_p) \bmod n$

$s = (-3 * 7 * 9 – 2* 11 *1) \bmod 77$

$s = (-189-22) \bmod 77$

$s = 20$

$-s = n-s$

$-s = 77-20$

$-s = 57$

NOW WE HAVE 4 SQUARE ROOTS {64,13,20, 57}

So indeed we got our plaintext which was 20 at the beginning

Example - finding the factorization of n

Like mentioned in one of the previous slides we use greatest common divisor

Since we know r and s we use

$\gcd(|r-s|,n) => \gcd(|64-20|,77) => \gcd(44,77)$ and the value will be either p or q

Since $\gcd(44,77) = 11$ we found out that it was q

THANK YOU
FOR YOUR ATTENTION