IMPLEMENTATION OF RABIN CRYPTOSYSTEM WITH ECB CIPHERING MODE

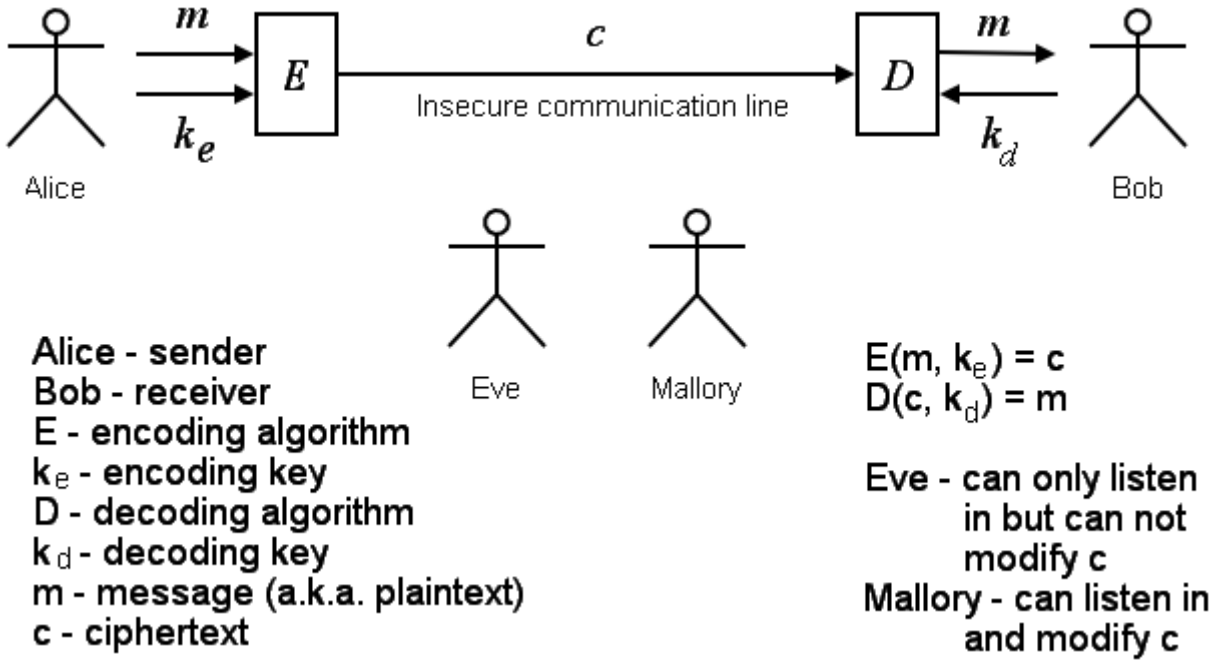
- by Jayant Nehra

Index

Abstract

Modern cryptography abandons the assumption that the Adversary has available infinite computing resources, and assumes instead that the adversary's computation is resource bounded in some reasonable way. In particular, in these notes we will assume that the adversary is a probabilistic algorithm who runs in polynomial time. Similarly, the encryption and decryption algorithms designed are probabilistic and run in polynomial time.

Overview



Public-key cryptography, or asymmetric cryptography, is any cryptographic system that uses pairs of keys: *public keys* that may be disseminated widely paired with *private keys* which are known only to the owner. There are two functions that can be achieved: using a public key to authenticate that a message originated with a holder of the paired private key; or encrypting a message with a public key to ensure that only the holder of the paired private key can decrypt it.

In a public-key encryption system, any person can encrypt a message using the public key of the receiver, but such a message can be decrypted only with the receiver's private key. For this to work it must be computationally easy for a user to generate a public and private key-pair to be used for encryption and decryption. The strength of a public-key cryptography system relies on the degree of difficulty (computational impracticality) for a properly generated private key to be determined from its corresponding public key. Security then depends only on keeping the private key private, and the public key may be published without compromising security.

Introduction

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

The Rabin cryptosystem is an asymmetric cryptographic technique, whose security, like that of RSA, is related to the difficulty of factorization. However the Rabin cryptosystem has the advantage that the problem on which it relies has been proved to be as hard as integer factorization, which is not currently known to be true of the RSA problem. 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 cipher-text, extra complexity is required on decryption to identify which of the four possible inputs was the true plain-text.

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. For the encryption, only the public key n is used, thus producing a cipher-text out of the plain-text. To decode the cipher-text, the private keys are necessary.

Key generation

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.

The precise key-generation process follows:

- Choose two large distinct primes p and q. One may choose
$$p \equiv q \equiv 3 \pmod 4$$
to simplify the computation of square roots modulo p and q (see below). But the scheme works with any primes.
- Let n= p.q Then n is the public key. The primes p and q are the private key.

To encrypt a message only the public key n is needed. To decrypt a cipher-text the factors p and q of n are necessary.

As a (non-real-world) example, if p=7 and q=11, then n=77 The public key, 77, would be released, and the message encoded using this key. And, in order to decode the message, the private keys, 7 and 11, would have to be known (of course, this would be a poor choice of keys, as the factorization of 77 is trivial; in reality much larger numbers would be used).

Encryption

For the encryption, only the public key n is used, thus producing a cipher-text out of the plain-text. The process follows:

$$P = \{0, \ldots, n-1\}$$

Let p be the plain-text space (consisting of numbers) and m be the

$$m \in P$$

plain-text. Now the cipher-text is determined by

$$c = m^2 \mod n$$

That is, c is the quadratic remainder of the square of the plain-text, modulo the key-number n.

$$P = \{0, \ldots, 76\}$$

In our simple example, p is our plain-text space. We will take m as our plain-text.

$$m = 20$$

The cipher-text is thus

$$c = m^2 \mod n = 400 \mod 77 = 15$$

For exactly four different values of *m*, the cipher-text 15 is produced, i.e. for

$$m \in \{13, 20, 57, 64\}$$

. This is true for most cipher-texts produced by the Rabin algorithm, i.e. it is a four-to-one function.

Decryption

To decode the cipher-text, the private keys are necessary. The process follows:

If *c* and *n* are known, the plain-text is then

$$m \in \{0, \ldots, n-1\}$$

with

$$m^2 \equiv c \pmod{n}$$

. For a composite *n* (that is, like the Rabin algorithm's) there is no efficient method known for the finding of *m*. If, however is prime (or *p* and *q* are, as in the Rabin algorithm), the Chinese remainder theorem can be applied to solve for *m*.

Thus the square roots

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

and

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

must be calculated.

In our example we get

$$m_p = 1 \text{ and } m_q = 9$$

By applying the extended Euclidean algorithm, we wish to find and such that $y_p \cdot p + y_q \cdot q = 1$. In our example, we have $y_p = -3$ and $y_q = 2$

Now, by invocation of the Chinese remainder theorem, the four square roots +r,-r,+s,-s of $c + n\mathbb{Z} \in \mathbb{Z}/n\mathbb{Z}$ are calculated (here stands for the ring of congruence classes modulo *n*). The four square roots are in the set :

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

One of these square roots is the original plain-text *m*. In our example, .

$$m \in \{64, \mathbf{20}, 13, 57\}$$

Rabin pointed out in his paper, that if someone is able to compute both, and , then he is also able to find the factorization of because:

$$\gcd(|r - s|, n) = p \quad \text{or} \quad \gcd(|r - s|, n) = q$$

, where gcd means Greatest common divisor.

Since the Greatest common divisor can be calculated efficiently you are able to find the factorization of n efficiently if you know r and s. In our example (picking 57 and 13 as r and s):

$$\gcd(57-13,77)=\gcd(44,77)=11=q$$

Computing square roots

The decryption requires to compute square roots of the cipher-text c modulo the primes p and q. Choosing

$$p \equiv q \equiv 3 \pmod 4$$

allows to compute square roots more easily by

$$m_p = c^{\frac{1}{4}(p+1)} \bmod p$$

and

$$m_q = c^{\frac{1}{4}(q+1)} \bmod q$$

We can show that this method works for p as follows. First

$$p \equiv 3 \pmod 4$$

implies that $(p+1)/4$ is an integer. The assumption is trivial for $c \equiv 0 \pmod p$. Thus we may assume that p does not divide c. Then

$$m_p^2 \equiv c^{\frac{1}{2}(p+1)} \equiv c \cdot c^{\frac{1}{2}(p-1)} \equiv c \cdot \left(\frac{c}{p}\right) \pmod p,$$

where

$$\left(\frac{c}{p}\right)$$

is a Legendre symbol.

From .

$$c \equiv m^2 \pmod{pq}$$

follows that . Thus c is a quadratic residue modulo p. Hence .

$$\left(\frac{c}{p}\right) = 1$$

and therefore

$$m_p^2 \equiv c \pmod{p}.$$

The relation

$$p \equiv 3 \pmod{4}$$

is not a requirement because square roots modulo other primes can be computed too. E.g., Rabin proposes to find the square roots modulo primes by using a special case of Berlekamp's algorithm.

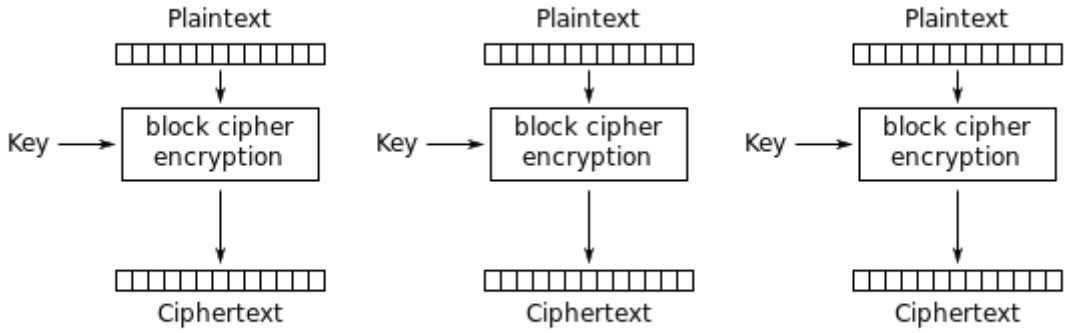
Block Cipher mode of operation

In cryptography, a mode of operation is an algorithm that uses a block cipher to encrypt messages of arbitrary length in a way that provides confidentiality or authenticity.[1] A block cipher by itself is only suitable for the secure cryptographic transformation (encryption or decryption) of one fixed-length group of bits called a block. A mode of operation describes how to repeatedly apply a cipher's single-block operation to securely transform amounts of data larger than a block.

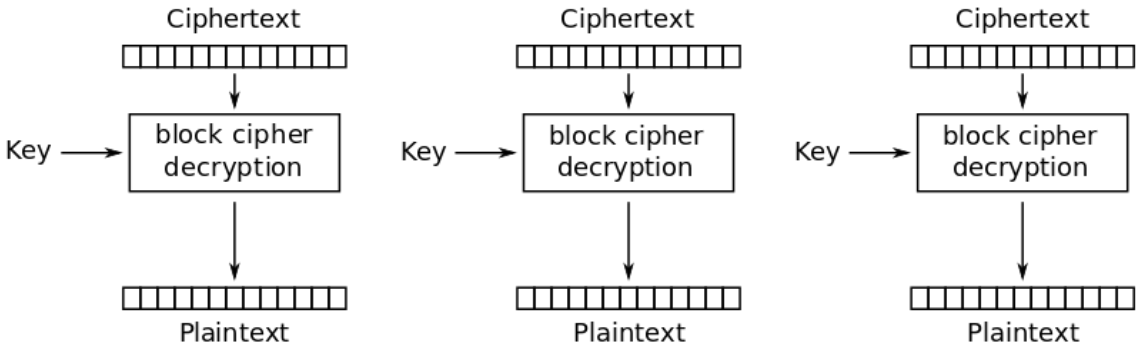
Most all modes require a unique binary sequence, often called an initialization vector (IV), for each encryption operation. The IV has to be non-repeating and, for some modes, random as well. The initialization vector is used to ensure distinct cipher-texts are produced even when the same plain-text is encrypted multiple times independently with the same key. Block ciphers have one or more block size(s), but during transformation the block size is always fixed. Block cipher modes operate on whole blocks and require that the last part of the data be padded to a full block if it is smaller than the current block size. There are, however, modes that do not require padding because they effectively use a block

cipher as a stream cipher; such ciphers are capable of encrypting arbitrarily long sequences of bytes or bits.

The simplest of the encryption modes is the **Electronic Codebook** (ECB) mode. The message is divided into blocks, and each block is encrypted separately.



Electronic Codebook (ECB) mode encryption



Electronic Codebook (ECB) mode decryption

The disadvantage of this method is that identical plain-text blocks are encrypted into identical cipher-text blocks; thus, it does not hide data patterns well. In some senses, it doesn't provide serious message confidentiality, and it is not recommended for use in cryptographic protocols at all.

A striking example of the degree to which ECB can leave plain-text data patterns in the cipher-text can be seen when ECB mode is used to encrypt a bitmap image which uses large areas of uniform colour. While the colour of each individual pixel is encrypted,

the overall image may still be discerned as the pattern of identically colored pixels in the original remains in the encrypted version.

ECB mode can also make protocols without integrity protection even more susceptible to replay attacks, since each block gets decrypted in exactly the same way.

Evaluation of the algorithm

Effectiveness

Decoding produces three false results in addition to the correct one, so that the correct result must be guessed. This is the major disadvantage of the Rabin cryptosystem and one of the factors which have prevented it from finding widespread practical use.

If the plain-text is intended to represent a text message, guessing is not difficult; however, if the plain-text 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 plain-texts 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. (Unless the convention of setting e=3 in the public key is used) 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 plain-text can be recovered entirely from the cipher-text only if the codebreaker is capable of efficiently factoring the public key

n.Note that this is a very weak level of security. Extensions of the Rabin cryptosystem achieve stronger notions of security.

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 plain-text was not created with a specific structure to ease decoding.)

Since the solution to the factorization problem is being sought on many different fronts, any solution (outside classified research organizations such as NSA) would rapidly become available to the whole scientific community. However, a solution has been long in coming, and the factorization problem has been, thus far, practically insoluble. Without such an advance, an attacker would have no chance today of breaking the code. This cryptosystem is provably secure (in a strong sense) against chosen plain-text attacks.

However, it has been proven an active attacker can break the system using a chosen cipher-text attack. By adding redundancies, for example, the repetition of the last 64 bits, the system can be made to produce a single root. This thwarts the chosen cipher-text attack, since the decoding algorithm then only produces the root that the attacker already knows. If this technique is applied, the proof of the equivalence with the factorization problem fails, so it is uncertain as of 2004 if this variant is secure. The Handbook of Applied Cryptography by Menezes, Oorschot and Vanstone considers this equivalence probable, however, as long as the finding of the roots remains a two-part process (1. roots and and 2. application of the Chinese remainder theorem).

Since in the encoding process, only the modulo remainders of perfect squares are used (in our example with , this is only 23 of the 76 possible values), other attacks on the process are possible.