Introduction to Public Key Cryptography

Ryan C. Daileda

Trinity University

Number Theory

Introduction

Today we will discuss the essential elements of *cryptography*, which is the study and practice of secure communication in the presence of eavesdroppers.

We will begin by introducing basic terminology and discussing ways to represent messages as integers.

We will then look at an elementary encryption scheme known as the *Caesar cipher*, which turns out to be far from secure.

Finally we will discuss the RSA cryptosystem, which is widely used today.

The Basics

Goal. Communicate securely in the presence of eavesdroppers.

Idea. Transmit encoded messages that only the intended recipient can decode.

Terminology.

- *plaintext:* the message to be sent
- *ciphertext:* the encoded version of the message
- *alphabet:* the symbols used to write the plaintext and ciphertext
- *letter:* member of the alphabet
- *encryption/enciphering:* the process of converting plaintext to ciphertext
- *decryption/deciphering:* the process of converting ciphertext to plaintext

Message Units

Plaintext/ciphertext are usually broken into blocks called *message units* before being enciphered/deciphered.

Examples of message units include single letters, pairs of letters (*digraphs*), strings of 100 letters, etc.

We let

$$\mathcal{P} = \{\text{plaintext message units}\},$$
$$\mathcal{C} = \{\text{ciphertext message units}\}.$$

Cryptosystems

An *enciphering transformation* is a bijection $f : \mathcal{P} \to \mathcal{C}$. Its inverse is the *deciphering transformation*.

A *cryptosystem* is a quadruple $(\mathcal{P}, \mathcal{C}, f, f^{-1})$. Schematically:

$$\mathcal{P} \xrightarrow{f} \mathcal{C} \xrightarrow{f^{-1}} \mathcal{P}.$$

If $m_1, m_2, \ldots, m_r \in \mathcal{P}$ and our plaintext is $M = m_1 m_2 \cdots m_r$, the the resulting ciphertext is

$$M' = f(m_1)f(m_2)\cdots f(m_r).$$

M' is decrypted by applying f^{-1} to each of its blocks.

Numerical Equivalents

It is convenient to represent messages using numbers rather than letters. There are various ways to do this.

If our alphabet has N letters we can easily biject it with $\mathbb{Z}/N\mathbb{Z}$.

For instance, if our alphabet is A–Z with a space:

(space)	A	B	C	$\cdots$	Y	Z
0	1	2	3	$\cdots$	25	26

Message units can then be encoded using base N representations.

Suppose our message units are strings of length ℓ:

$$m_1 m_2 \cdots m_\ell,$$

with each letter m_i converted to a member of $\mathbb{Z}/N\mathbb{Z}$.

This defines a single integer in base N:

$$M = m_\ell N^{\ell-1} + m_{\ell-1} N^{\ell-2} + \cdots + m_2 N + m_1 < N^\ell.$$

Conversely, any $0 \leq M < N^\ell$ has a unique base N representation, whose "digits" yield a length ℓ message unit.

Warning. With this scheme *the order of letters and "digits" will be reversed.*

Examples

Example 1

Using the 27 letter alphabet discussed above, express the message HIDE as an integer.

Solution. Since H $= 8$, I $= 9$, D $= 4$, E $= 5$, we have

$$M = 5 \cdot 27^3 + 4 \cdot 27^2 + 9 \cdot 27 + 8 = \boxed{101582}.$$

□

Example 2

Express the six-letter message unit $M = 359707244$ in our 27 letter alphabet.

Solution. We need to compute the base 27 expansion of M.

We repeatedly divide by 27 to obtain the "digits" of M.

$$\begin{aligned}
359707244 &= 13322490 \cdot 27 + \boxed{14}, \\
13322490 &= 493425 \cdot 27 + \boxed{15}, \\
493425 &= 18275 \cdot 27 + \boxed{0}, \\
18275 &= 676 \cdot 27 + \boxed{23}, \\
676 &= 25 \cdot 27 + \boxed{1}, \\
25 &= 0 \cdot 27 + \boxed{25}.
\end{aligned}$$

These are already in the correct order, so we simply convert back to letters, obtaining the message

$$\boxed{\text{NO WAY}}.$$

The Caesar Cipher

Suppose our alphabet is $\mathbb{Z}/N\mathbb{Z}$ and we using single letter message units, i.e. we set $\mathcal{P} = \mathcal{C} = \mathbb{Z}/N\mathbb{Z}$.

For any $b \in \mathbb{Z}/N\mathbb{Z}$ we define the *shift transformation* by

$$C = f(P) \equiv P + b \pmod{N}.$$

The deciphering transformation is another shift, given by

$$P = f^{-1}(C) \equiv C - b \equiv C + (N - b) \pmod{N}.$$

This cryptosystem is called the *Caesar cipher*. The letter b is called its *key*.

Knowledge of the (common) key is necessary for both encryption and decryption.

Examples

Example 3

Encrypt the message ATTACK AT DAWN using the Caesar cipher on our 27 letter alphabet with key $b = R$.

Solution. We convert the message to its numerical equivalents and shift each by R = 18 modulo 27.

This yields

P :	1	20	20	1	3	11	0	1	20	0	4	1	23	14
C :	19	11	11	19	21	2	18	19	11	17	22	19	14	5

Converted back to letters this becomes

$$\boxed{\text{SKKSUBRSKRVSNE}}.$$

Example 4

Decrypt the message ROVZJWO, which was encrypted using a Caesar cipher with key $b = \mathrm{J}$.

Solution. To decrypt the message we must convert it to its numerical equivalents and subtract $\mathrm{J} = 10$ modulo 27.

We have:

C :	18	15	22	26	10	23	15
P :	8	5	12	16	0	13	5

This translates to

$$\boxed{\text{HELP ME}}.$$

Remarks

- In the modern day, the Caesar cipher is susceptible to *frequency analysis*: analyzing how often each letter occurs in the ciphertext.

  In English, the most commonly occurring letter likely corresponds to E (if the text is long enough).

  Knowing this is enough to crack the cipher, since if the most commonly occurring letter is x, then we must have $f(5) = 5 + b \equiv x \pmod{27}$, so that $b \equiv x - 5 \pmod{27}$.

- Perhaps even more realistically, a modern computer can simply try every possible key (there are only 27 in our alphabet) until a readable message is obtained.

Public Key Cryptography

Definition

Let $(\mathcal{P}, \mathcal{C}, f, f^{-1})$ be a cryptosystem. The information needed to compute f is called the *encryption key* and is denoted K_E. The information needed to compute f^{-1} is called the *decryption key* and is denoted K_D.

So, for the Caesar cipher we have

$$K_E = K_D = b.$$

In particular,

$$\text{knowledge of } K_E = \text{knowledge of } K_D.$$

This means that:

- If you can encrypt a message, you can decrypt any message.
- Both sender and recipient must know K_E and keep it secret.

The goal of *public key cryptography* is to break this symmetry.

Let S denote the message *sender* and R the (intended) *recipient*.

We seek cryptosystems for which

knowledge of K_E gives no (practical) information about K_D.

Using such a cryptosystem, if R keeps K_D secret, we can then make K_E *public* information, so that S can be *anyone*.

That is, anyone can securely communicate a message to R, and only R has to keep K_D secret.

This is advantageous since, as Benjamin Franklin observed, "Three may keep a secret, if two of them are dead."

Put another way, f is "easy" to compute (by anyone with K_E), but f^{-1} is "hard" to compute (without K_D).

Such an f is called a *trapdoor function* and such a cryptosystem is called a *public key cryptosystem*.

Given a public key cryptosystem and a group of individuals using it, each individual is assigned a key pair (K_E, K_D).

The decryption keys K_D are kept secret by each individual, while every K_E is published in a directory.

With this arrangement every member of the group can easily communicate securely with any other member.

RSA

Developed by Rivest, Shamir and Adleman (ca. 1977), the RSA public key cryptosystem works as follows.

Each individual in the group:

- chooses two prime numbers p and q;
- chooses a (random) e with $(e, (p-1)(q-1)) = 1$;
- sets $n = pq$ and publishes $K_E = (n, e)$;
- uses the EA to find d so that $de \equiv 1 \pmod{(p-1)(q-1)}$, and keeps $K_D = (n, d)$ secret.

We take $\mathcal{P} = \mathcal{C} = \mathbb{Z}/n\mathbb{Z}$ and for $P \in \mathcal{P}$ we define

$$C = f(P) \equiv P^e \pmod{n}.$$

Since

$$\varphi(n) = \varphi(pq) = (p-1)(q-1)$$

and $de \equiv 1 \pmod{(p-1)(q-1)}$, Euler's Theorem implies

$$C^d = (P^e)^d = P^{ed} \equiv P \pmod{n}.$$

Thus for $C \in \mathcal{C}$,

$$f^{-1}(C) \equiv C^d \pmod{n}.$$

Example

If $p = 17$ and $q = 31$, then $n = pq = 527$ and $(p-1)(q-1) = 16 \cdot 30 = 480$.

Suppose we choose $e = 377$. Then $d \equiv e^{-1} \equiv 233 \pmod{480}$.

Suppose our plaintext block is $P = 300$. Then our ciphertext is

$$C \equiv P^e \equiv 300^{377} \equiv 210 \pmod{572}.$$

In the other direction, suppose we receive the cipertext $C = 432$. The the corresponding plaintext is

$$P \equiv C^d \equiv 432^{233} \equiv 333 \pmod{572}.$$

The Trapdoor

I practice, RSA uses extremely large primes (100 digits or more). Why does this make RSA secure?

Since $K_E = (n, e)$ is public knowledge, *anyone* can encrypt a message and send it.

However, in order to decrypt a message, one needs $K_D = d$, which is the multiplicative inverse of e in $\mathbb{Z}/(p-1)(q-1)\mathbb{Z}$.

Claim. The function $f(P) \equiv P^e \pmod{n}$ is a trapdoor function.

Proof. Repeated squaring provides an efficient procedure for computing P^e, even for large (hundreds of digits) n.

So f is "easy" to compute.

It is also easy to compute d via the EA, provided we are *given the modulus*

$$\varphi(n) = (p-1)(q-1).$$

But knowledge of $\varphi(n) = (p-1)(q-1)$ and $n = pq$ is equivalent to knowledge of q and q.

To see this first observe that if we know p and q, then it is trivial to compute pq and $(p-1)(q-1)$.

So suppose we know $n = pq$ and $\varphi(n) = (p-1)(q-1)$.

Note that $\varphi(n) = pq - (p+q) + 1 = n - (p+q) + 1$, so that $p + q = n - \varphi(n) + 1$.

It follows that

$$(X-p)(X-q) = X^2 - (p+q)X + pq = X^2 - (n-\varphi(n)+1)X + n,$$

so that p and q can be computed using n, $\varphi(n)$ and the quadratic formula.

So knowledge of n and $\varphi(n)$ is equivalent to knowledge of the factorization $n = pq$.

However, factorization of integers is a notoriously difficult problem!

We conclude that K_D, and hence f^{-1}, is "hard" to compute. □

Our proof shows that an eavesdropper's difficulty in cracking a given RSA cipher (i.e. determining K_D from K_E) rests in the difficulty of factoring n.

Therefore if p and q are large enough to make the factorization of $n = pq$ prohibitive, the enciphering transformation f should be secure.

Remark. Strictly speaking, the application of Euler's Theorem in RSA requires $P \in (\mathbb{Z}/n\mathbb{Z})^\times$. However:

- One can show that $P^{ed} \equiv P \pmod{n}$ even if P has factors in common with n.
- If $(P, n) \neq 1$, then by computing (P, n) we can factor n. Given that factoring n is "hard," such a plaintext block is not likely to be encountered.