A Peek to Cryptography History

Jimmy Wang

March 18, 2020

Abstract

This paper is intended to introduce some basics of cryptography, covering the idea of secure communication, some of the classic cryptosystems, including Caesar cipher, Vigenère, and other traditional encoding ways, while introducing how the cryptosystem works or some features of them.

1 Introduction

Information transmitting happens all the time. For example, right now, you are reading this piece of text. Reading is a kind of information transmitting. You read my text, thus my words and ideas are transmitted to you. Letters, pictures, touching, or even face-to-face speaking, are all kinds of information transmitting.

Here is a question, how do you actually understand my words, instead of thinking this paper is just kind of gibberish? The answer is, I am writing in English. My ideas are *encoded* in a widely-known type of language, English, while only someone who understands English can read my work. English became the bridge connecting you and my ideas. However, if someone who never studied English before, would not understand this piece of text without a translator.

In some circumstances, however, transmitting plain text is not that good, because there are too many people can read that. For at least 2,000 years, people want their information shared with only a specific group of people. That means, people without a kind of permission by the creator should not read or understand the contents. This, is the idea of *Cryptography*.

Cryptography, in other words, is a study of applying some transformation to the plain texts to make it gibberish-like, helps to keep communication *secure*. It is like when I am sending a message to you, I put that message into a box with a lock on it, and only you,

who obtain a key which I gave to you a few weeks ago, can unlock that box. Anyone except you can never, or should never open that box. This kind of method might work, unless the box is intercepted by a person who knows how to pick a lock.

Nowadays, the 'lock' we put on the box has been upgraded by the computer. It is like a digital combination lock that can hardly be broken even when the world ends. Whereas, before computer appears, people have already developed a few ways to keep information secure. This paper will introduce some classical, traditional ways to encode messages. I'll explain from communication, to the most classic substitution system, and other traditional encoding methods.

2 Some Aspects of Communication and Security

Before we start our trip to some detailed encryptions, I would like to introduce some basics about *communication*.

The idea of transmitting, or technically it is called communication, is we "reproduce at one point exactly or approximately a message selected at another point" [7]. The father of informatics, Claude E. Shannon, introduced this famous model about communication system:

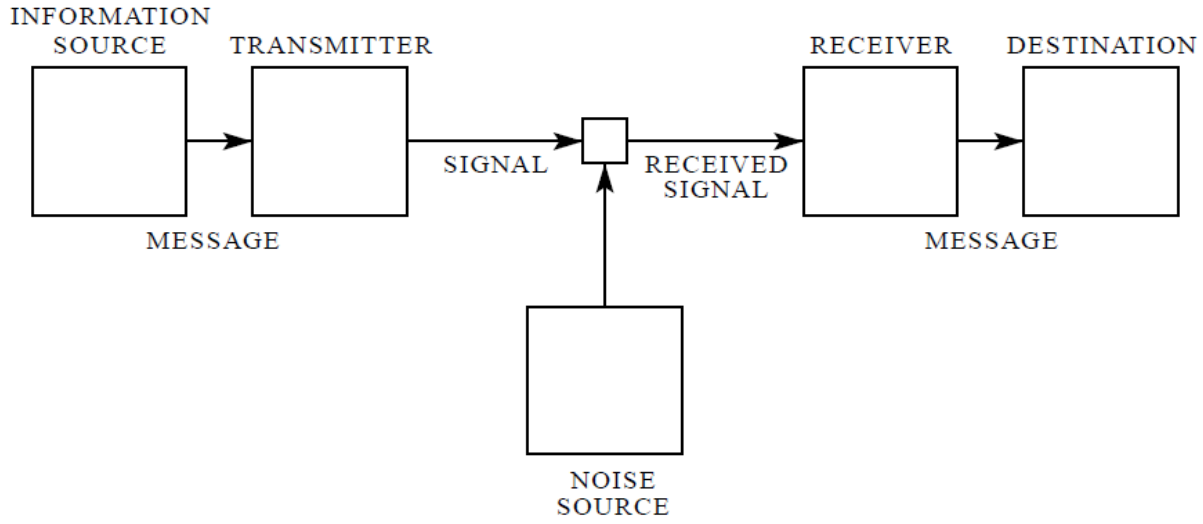

Figure 1: Schematic diagram of a general communication system [7]

The sender, or *information source*, would like to transmit a piece of message to the destination. The original message, which we give it a special name called *plaintext*, would first go through the *transmitter*, which transfers (encodes) the plain text into signals. Bypassing some medium, like paper, cable wires, fibers or electromagnetic waves, the *receiver* device near to the destination receives those signals, and transfers (decodes) them back to plaintext. The plaintext then presents at *destination*. There might be some noise source interfereing our signals during the trasmitting, but we would not consider that in the idea of traditional cryptography. We would assume that our signals are perfectly transmitted to the destination.

An obvious problem is, if we take a look at this figure, the information source cannot make sure that the signals are transmitted to the destination. Moreover, does the signal transmitted only to the designated destinations? For example, if signals are tranferring via some electromagnetic waves across the sky, what would happen if someone else is listening to the same channel? Others would receive the signals and therefore know what are we talking.

You may confirm the information is transferred successfully by receiving an acknoledgement or reply from the destination, but it turns out that you can hardly avoid someone else listening to the medium. Even you are whispering to your friend, maybe there is a third person nearby can hear your voice. So, the best solution is, do not transmit the plain text. To make things secure, we use some technique dealing with the text at the ends. That is, the source and destination.

Here is an example. I am sending messages to Bob. At the information source, I *encrpyt* the plain text into some code, using a protocol or an algorithm that Bob and I agree with. At the destination, Bob *decrpyts* the code into plain text. Since encryption and decryption only happen at the ends, it would be a hard time for anyone who monitors the signals and tries to decrypt the message. Because he/she doesn't know the protocol used by Bob and me, he/she would make a large number of guesses to figure out how to interpret the encoded texts. In other words, he/she needs to guess our protocol to decrypt the message.

It is true that if some snooper takes much effort, the snooper could went through all kinds of combinations and ultimately he can figure out the protocol. This process is called *breaking* a cipher. However, if our protocol is complicated enough, it would take a long time, say 100 years, for the snoopers to break it down. If that is the case, we could say the protocol is *relatively* secure.

In the following sections, I would introduce some of the classic protocols that human beings used in history. They may be beaten by the fantastic, well-developed computer today, but they worked for a while before the computer appears on the Earth.

3 Substitution Cipher System

3.1 Caesar Cipher

It was at the first century before christ, Gaius Julius Caesar, the emperor of the Roman Empire, applied a way to write down his instructions securely. When he was trying to write some messages, instead of writing it directly, he shift each letter three times forward in the alphabet, and put that shifted letter down. For example, in English, we would replace `A` with the letter `D`, and replace `B` with letter `E`. The rule keeps shifting when it reaches letter `W`, which we'll use `Z`, and we are out of alphabets. However, we start at letter `D`, the letter `A`, `B`, `C` were not used. So we would reuse them, replacing letter `X` with `A`, `Y` with `B`, and `Z` with `C`. If we use a table to represent this kind of relationship:

Plain Text	A	B	C	$\cdots$	W	X	Y	Z
Cipher Text	D	E	F	$\cdots$	Z	A	B	C

Table 1: Caesar Cipher, 3 forward

So in the English, we could encrypt the plain text

`HISTORY OF MATHEMATICS`

into

`KLVWRUB RI PDWKHPDWLFV`

where letter `H` becomes `K`, and `I` becomes `L`, etc.

Note that the previous cipher table is in English, not what Caesar really used in his language, Latin. But it should give you an intuition how the Caesar Cipher works.

Actually, the Caesar Cipher doesn't have to stick with 3 step forward. We could choose 2 step forward, so `A` replaced by `C`, or 4 step forward, `A` replaced by `E`, or even 25 step forward, `A` replaced by `Z`, although in this case it is better to call it as 1 step backward.

Caesar Cipher is, as you could imagine, easy to break. One could simply break it by trying all 25 different shifting ways. And see if it can form some meaningful texts. If one of the shifting way works, the snooper wins.

Caesar nevertheless earned a place in the history of cryptography, for the 'Julius Caesar' cipher, as it is still called, is an early example of an *encryption system* and is a special case of a *simple substitution cipher*. [2]

3.2 General Substitution Cipher System

Caesar Cipher is simple. It is based on the order of alphabet and simply just do the shifting, which as I shown above, is easy to break down. One can try all 25 different shifting.

We could make the Caesar more complicated. Suppose we just scramble all 26 alphabets in a random order. The rule is, the plain text alphabets and the cipher text alphabets have a *one-to-one correspondence*. Say I replace `A` with `G`, `B` with `K`, and `C` with `X`, and so on. Each letter in the cipher text alphabet appears once and only once.

Notice I used the word *one-to-one correspondence*. This is required in the substitution method, because ultimately we need to decipher it. If two plaintext letters are replaced by the same cipher letter, the receiver may have a hard time deciphering, since if that cipher letter appears, he/she needs to guess which plaintext letter it represents. Of course, by English wording rules, or when we put the words together into a sentence, we may figure it out, but that is not suitable for all cases.

As I mentioned, we have *one-to-one correspondence* in a substitution cipher system. Personally, I would like to use the function notation to represent this relationship. That is:

$$f : \Sigma \to \Sigma$$

where the Σ notation represents the alphabet, the set of all letters we use in the plaintext and also in the cipher.

Consider this "function" as a black box. We input a plaintext letter to the box, and the box provides an output of the *encrypted* cipher letter. To decipher it, we'll need an inverse of this box:

$$f^{-1} : \Sigma \to \Sigma$$

which does exactly the inverse of f, receiving a cipher letter and change it back to the original plaintext letter. This inverse function must exist and it is unique, by the property of one-to-one correspondence functions.

We can further define an extended version of our cipher function f so it could work on strings instead of individual letters, if we introduce the * notation, the Kleene Star.

Back to the substitution we extended from Caesar. How many different kinds of cipher functions f can we found, if the alphabet has 26 letters? The answer is $26! \approx 4.03 \times 10^{26}$, where ! represents factorial. If one can test a type of combination with 10 seconds, he would need 4.03×10^{27} seconds to try all combinations. This number is, however, about 10^{20} years, a billion times greater than the time passed since Earth was born. Seems like that much time is secure, right?

Unfortunately, breaking down a substitution cipher is easier than we expected. Even Achilles has a weak point at his heel.

3.3 Break Down the Substitution

The idea to systematically break a substitution cipher is called *Frequency Analysis*, first described by the Arabic philosopher and mathematician Al-Kindi, in the ninth century [1]. The word *frequency analysis* contains two parts: frequency, and analysis. The word 'frequency' stands for how often the various symbols occur in the ciphertext; based on the frequency that letters occur in some language, and possibly some ways to form words, we would deduce each symbol represents which letter, and that's *analysis*.

For example, in English, some letters are occured significantly more than other letters. The following figure shows the frequencies of each letter occur in English language.

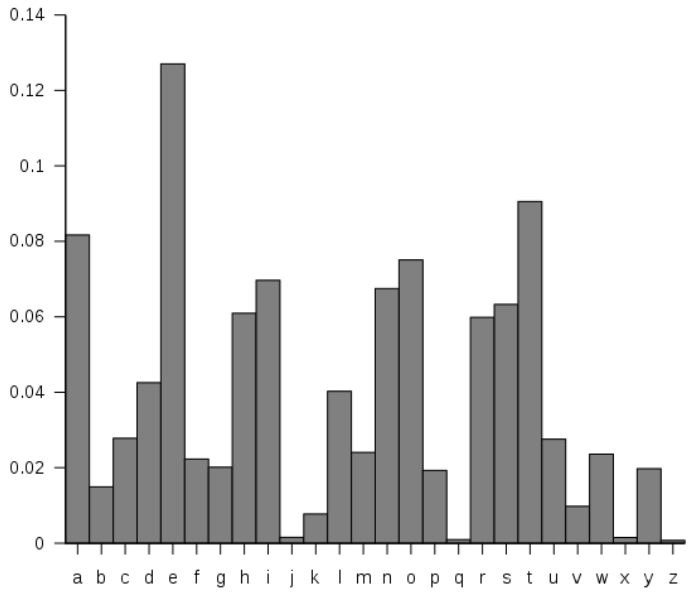

Figure 2: Letter Frequencies in English [5]

It appears that the letter `E` occurs most frequently, following the letter `T`; letters `A`, `O`, `I`, and `N` are also frequently occured, with quite the same frequency. So by counting all symbols appeared in a cipher text, if a symbol occurs most frequently in a substitution cipher text generated from English, we could say that symbol is more likely representing letter `E`.

After substituting some of the most frequently occured symbols with these letters, we could start guessing some words. Some frequently used words can be deduced, like `t*e` is most likely to be the word `the`, `t**t` usually stands for `that`, etc. A single letter word would be `A` or `I`, and some two-letter words like `is`, `it`, `to`, `as` may also easy to recognize.

I'll show an example breaking substitution. This text is generated from English plaintext and encrypted using some substitution.

```
MTS SR JUR RMWJWRA OM J WURJI YOVOD SJU IRGIOMW SHRIHRU
IHJI MJIOTM TU JML MJIOTM GT YTMYROVRA JMA GT ARAOYJIRA
YJM DTMW RMAPUR SR JUR QRI TM J WURJI CJIIDRNORDA TN IHJI
SJU SR HJVR YTQR IT ARAOYJIR J FTUIOTM TN IHJI NORDA
JG J NOMJD URGIOMW FDJYR NTU IHTGR SHT HRUR WJVR IHROU
DOVRG IHJI IHJI MJIOTM QOWHI DOVR OI OG JDITWRIHRU NOIIOMW
JMA FUTFRU IHJI SR GHTPDA AT IHOG
```

The first thing after we suggest this is a substitution method, is to count how many times does each symbol occur in this text.

A	14	B	0	C	1	D	11	E	0	F	4
G	10	H	17	**I**	**37**	J	33	K	0	L	1
M	22	N	7	O	24	P	2	Q	3	**R**	**39**
S	9	T	23	U	17	V	6	W	11	X	0
Y	8	Z	0								

Table 2: Frequency Table of the Example Text

The letter `R` and `I` occurs the most among all 26 symbols. If we assume this text is written in English, I'll substitute the letter `R` with `e`, the most frequently occured letter, and `I` with `t`, the second-most frequently occured letter. Here and the following examples, I'll use uppercase letters for cipher letters, and lowercase letters for temporary plaintext letters.

```
MTS Se JUe eMWJWeA OM J WUeJt YOVOD SJU teGtOMW SHetHeU
tHJt MJtOTM TU JML MJtOTM GT YTMYeOVeA JMA GT AeAOYJteA
YJM DTMW eMAPUe Se JUe Qet TM J WUeJt CJttDeNOeDA TN tHJt
SJU Se HJVe YTQe tT AeAOYJte J FTUtOTM TN tHJt NOeDA
JG J NOMJD UeGtOMW FDJYe NTU tHTGe SHT HeUe WJVe tHeOU
DOVeG tHJt tHJt MJtOTM QOWHt DOVe Ot OG JDtTWetHeU NOttOMW
JMA FUTFeU tHJt Se GHTPDA AT tHOG
```

This still looks a bit vague. We could try substitute a third letter with `A`, the third-most frequently appeared letter, or we could find some two-letter words and guess what it could be. In this case I choose the word `tT` in the 4^{th} line. Notice that the only 2-letter word in English with a `t` beginning is `to`. Therefore I substitute `T` with `o`, and get the following.

```
MoS Se JUe eMWJWeA OM J WUeJt YOVOD SJU teGtOMW SHetHeU
tHJt MJtOoM oU JML MJtOoM Go YoMYeOVeA JMA Go AeAOYJteA
YJM DoMW eMAPUe Se JUe Qet oM J WUeJt CJttDeNOeDA oN tHJt
SJU Se HJVe YoQe to AeAOYJte J FoUtOoM oN tHJt NOeDA
JG J NOMJD UeGtOMW FDJYe NoU tHoGe SHo HeUe WJVe tHeOU
DOVeG tHJt tHJt MJtOoM QOWHt DOVe Ot OG JDtoWetHeU NOttOMW
JMA FUoFeU tHJt Se GHoPDA Ao tHOG
```

We notice that a word `tHJt` appeared a lot in the text. A 4-letter word in English starts with `t` and ends with `t`, a good guess would be `that`. In addition, the letter `J` appears individually as a 1-letter word, which I mentioned could only be `a` or `i`, strengthens our guessing that `J` stands for `a`. I'll replace `H` with `h` together.

```
MoS Se aUe eMWaWeA OM a WUeat YOVOD SaU teGtOMW ShetheU
that MatOoM oU aML MatOoM Go YoMYeOVeA aMA Go AeAOYateA
YaM DoMW eMAPUe Se aUe Qet oM a WUeat CattDeNOeDA oN that
SaU Se haVe YoQe to AeAOYate a FoUtOoM oN that NOeDA
aG a NOMaD UeGtOMW FDaYe NoU thoGe Sho heUe WaVe theOU
DOVeG that that MatOoM QOWht DOVe Ot OG aDtoWetheU NOttOMW
aMA FUoFeU that Se GhoPDA Ao thOG
```

After substituting five letters, some words become more obvious. In line 4 we have `haVe`, which is simply just `have`, while in line 1 we have `ShetheU`, and the word `whether` is the best fit. Substitute `V` with `v`, `S` with `w`, and `U` with `r`.

```
Mow we are eMWaWeA OM a Wreat YOvOD war teGtOMW whether
that MatOoM or aML MatOoM Go YoMYeOveA aMA Go AeAOYateA
YaM DoMW eMAPre we are Qet oM a Wreat CattDeNOeDA oN that
war we have YoQe to AeAOYate a FortOoM oN that NOeDA
aG a NOMaD reGtOMW FDaYe Nor thoGe who here Wave theOr
DOveG that that MatOoM QOWht DOve Ot OG aDtoWether NOttOMW
aMA FroFer that we GhoPDA Ao thOG
```

On line 6 we have a 2-letter word `Ot`, which could be `it` or `at`. However, `a` has already being used, so guessing `O` as `i`. Another evidence is in line 5, the word `theOr`, which if we substitute `O` with `i`, is `their`.

```
Mow we are eMWaWeA iM a Wreat YiviD war teGtiMW whether
that MatioM or aML MatioM Go YoMYeiveA aMA Go AeAiYateA
YaM DoMW eMAPre we are Qet oM a Wreat CattDeNieDA oN that
war we have YoQe to AeAiYate a FortioM oN that NieDA
aG a NiMaD reGtiMW FDaYe Nor thoGe who here Wave their
DiveG that that MatioM QiWht Dive it iG aDtoWether NittiMW
aMA FroFer that we GhoPDA Ao thiG
```

The text looks better now, that we could guess a lot more words. The next word few words to figure out is `MatioM` in line 2, which can be replaced with `nation`; the word `FroFer` in line 7, which is clearly `proper`, and the word `aDtoWether` in line 6, could be `altogether`. Replace `M` with `n`, `F` with `p`, `D` with `l`, and `W` with `g`.

```
now we are engageA in a great Yivil war teGting whether
that nation or anL nation Go YonYeiveA anA Go AeAiYateA
Yan long enAPre we are Qet on a great CattleNielA oN that
war we have YoQe to AeAiYate a portion oN that NielA
aG a Ninal reGting plaYe Nor thoGe who here gave their
liveG that that nation Qight live it iG altogether Nitting
anA proper that we GhoPlA Ao thiG
```

At this moment, a lot of words are now already decrpyted. What's next is just repeat guessing the cipher letters. `engageA` with `engaged`, `Yivil` with `civil`, `teGting` with `testing`. So `A` with `d`, `Y` with `c`, `G` with `s`.

```
now we are engaged in a great civil war testing whether
that nation or anL nation so conceived and so dedicated
can long endPre we are Qet on a great CattleNield oN that
war we have coQe to dedicate a portion oN that Nield
as a Ninal resting place Nor those who here gave their
lives that that nation Qight live it is altogether Nitting
and proper that we shoPld do this
```

And finally we only got 5 cipher letters left. `anL` with `any`, `shoPld` with `should`, `Ninal` for `final`, `coQe` for `come`, and `CattleNield` for `battlefield`. Substituting the last few letters and we are done.

```
now we are engaged in a great civil war testing whether
that nation or any nation so conceived and so dedicated
can long endure we are met on a great battlefield of that
war we have come to dedicate a portion of that field
as a final resting place for those who here gave their
lives that that nation might live it is altogether fitting
and proper that we should do this
```

The excerpt is from the famous speech, *the Gettysburg Address*, by Abraham Lincoln [4].

To sum up, to break a general substitution cipher, we first use frequency analysis and substitute some of the most frequently appeared letters in cipher text with most frequently appeared letters in English. Then we focus on 1-letter, 2-letter or some easy to recognize words, like prepositions and conjunctions, and substituting the corresponding letters. After we substituted a few letters, we could figure out more words, and finally decipher the whole text.

This is general substitution. It may take some time to break, but it is still not secure.

3.4 Vigenère: The "Peak" of Substitution

How could we improve the substitution cipher system, such that it cannot be broken down by frequency analysis? In the 16th century, French diplomat Blaise de Vigenère came up with an idea that using multiple Caesar ciphers to encrypt a same text [1].

Say we shift the first letter by 1, the second by 2, the third by 3, and so on, our example

`HISTORY OF MATHEMATICS`

into

`IKVXTXF WO WLFUSBQKAVM`

where you see `S` in word `HISTORY` has been encrypted with letter `V`, but the last `S` in word `MATHEMATICS` has been encrypted with letter `M`.

A more complicated version, which is the original Vigenère Cipher, is we have a *key* to encrpyt. A *key* is a sequence of letters, or a *string*, that is describing each letter shall be applied with which Caesar Cipher.

For example, a key `MATH`. The first letter `M` means plaintext letter `A` shall be replaced by `M`, `B` with `N`, and so on; the second letter `A` means the plaintext letter shall not be transformed. The third letter `T`, similarly, means we shall apply some Caesar cipher that shifts `A` to `T`.

Plain Text	A	B	C	$\cdots$	W	X	Y	Z
Cipher M	M	N	O	$\cdots$	I	J	K	L
Cipher A	A	B	C	$\cdots$	W	X	Y	Z
Cipher T	T	U	V	$\cdots$	R	Q	R	S
Cipher H	H	I	J	$\cdots$	D	E	F	G

Table 3: Vigenère Cipher, Key = `MATH`

We encrypt the first plaintext letter to Cipher M, second plaintext to Cipher A, third to Cipher T, fourth to Cipher H, and we are out of keys. In this case, we will reuse Cipher M to encrypt the fifth plaintext letter, and Cipher A to the sixth, This reusing key method is called *repeat key*.

Using Vigenère and the key `MATH`, we could encrypt the example

`HISTORY OF MATHEMATICS`

into

`TILAARR VR MTATEFHFIVZ`

It is very difficult to break a Vigenère cipher. In fact, since it is created at 1500s, for a whole three centuries people didn't come up with a useful algorithm to break it down.

Until mid 1800s, the English mathematician Charles Babbage, who is now recognized as a founding figure in the field of computing, suggested that if we could guess or deduce the *length* of the key, and the key is repeatedly used in Vigenère cipher, what's next is just breaking a few simple substitutions. He used some extended frequency analysis to discover the length of the key, but he never published his algorithm to public. Some evidence says that the British Intelligence stopped him. However, in 1863, a Prussian Army officer, William Kasiski, independently figured out how to break the Vigenère code and published the method to public. Since then, the Vigenère was insecure [1].

4 Diverse Traditional Encryption

Vigenère was the peak of traditional substitution cipher system. However, besides the Vigenère, a few other types of cipher system are also remarkable to be mentioned. They may or may not based on the substitution, nevertheless, all of them gives us some new idea developing the traditional cyptography.

4.1 From Monograph to Digraph

The word *monograph*, a combination of Greek *mono* ('single') and *grapho* ('to write'), has a special meaning in cryptography. Like the substitution ciphers I introduced on the previous chapter, each individual letter is replaced by another symbol. This transformation working as

$$\Sigma \to \Sigma$$

is called a monograph cryptosystem. However, the symbols we are replacing are not restricted to be a single letter. In fact, there are some cryptosystems that use two, or even more letters to encode a single letter, that is:

$$\Sigma \to \Sigma^*$$

where the star could be any specified positive integer.

Some more complicated cryptosystems may even have different length of encoding. For example, the Huffman encoding. Each letter is encrypted to 0s and 1s of different length according to the frequency they occur in the plaintext. The letters that occur much more frequently may have shorter encodings than those rarely occur.

Here I am introducing two different encoding ways that use multiple letters to encode a single character.

4.1.1 Telephone Keypad Cipher

Telephone keypad was born in 1960s in United States, with some letters bear on the numbers. The letters come from a legacy system, *Telephone exchange name*.

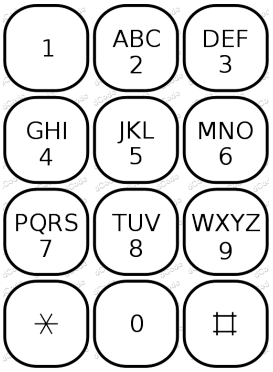

Figure 3: A cellphone keypad

Although people in United States don't dial letters anymore, these letters are preserved and thus provided a kind of transformation, which is later called *Telephone Keypad Cipher*.

The idea is to convert each letter into two digits. The first digit is its position on the keypad, i.e. which key contains that letter; the second digit is its order on that key. For example, `B` is on the key `2`, and it is the second letter on that digit, so the letter `B` would be encrypted as `22`. `L` is on key `5`, and its the third letter, so the code for `L` is `53`.

In addition, when there is a space character that separates the words, we put a `0` or `1` in between (`1` is more common), working as a *delimiter*.

Using the Telephone Keypad Cipher, we could encrypt

`HISTORY OF MATHEMATICS`

into

`4243748163739316333161218142326121814323 74`

where I picked `1` as delimiter.

4.1.2 Morse Code

A famous example for a varying-length encoding system is the *Morse Code*. I use the term *encoding system* rather than cryptosystem is because this is well-known, and more likely to be a standard in telecommunication.

When transferring signals, we use a thing called *telegraph key*, or *key*.

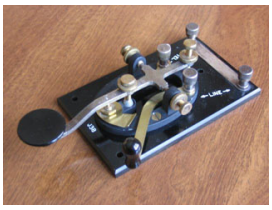

Figure 4: A Telegraph Key [6]

When the sender pressed the switch using a finger, the circuit is connected and send out a high signal, indicating `1`. When it is not pressed, it send out a low signal, indicating `0`.

In Morse Code, there are two more advanced signals, `.` (called *dits* or *dots*), and `-` (called *dahs* or *dashes*). Their difference is their length of timing. A dot is pressing the switch, and when it hits the bottom, release the finger. A dash's length is three times that of a dot, so the sender holds for a while before his releases his finger.

Each letter is then, according to the standard, encoded into different dots and dashes:

A	.-	B	-...	C	-.-.	D	-..	E	.	F	..-.
G	--.	H	....	I	..	J	.---	K	-.-	L	.-..
M	--	N	-.	O	---	P	.--.	Q	--.-	R	.-.
S	...	T	-	U	..-	V	...-	W	.--	X	-..-
Y	-.--	Z	--..								

Table 4: International Morse Code

The delimiters between each letter is one unit of dots low. In other words, after sending some letter, you wait for a unit of time not pressing anything, and then sends your second letter. That unit of time should be the same as if you send a dot. Moreover, the delimiter between words is three units of dots low, and between the sentences is seven units.

The Morse Code encryption of

`HISTORY OF MATHEMATICS`

is

```
.... .. ... - --- .-. -.-- / --- ..-. /
-- .- - .... . -- .- - .. -.-. ...
```

where I intentionally use the `/` symbol to indicate a space, so it could display in text instead of three whitespaces.

4.2 Not the Letter, but the Position

So far, many cyptosystems I introduced above are all based on the old-fashioned *substitution* method. There are also a few ways that doesn't have the root of substitution. One cryptosystem is called *Rail Fence Cipher*. This kind of transformation is focus on changing the position of each letters, more likely to be an anagram.

Rail Fence Cipher, or alternatively called *zig-zag cipher*, is by putting the ciphers into a box, and do a transpose operation. It is better to explain this cryptosystem using a visualization example. Here, again, we'll encrypt

`HISTORY OF MATHEMATICS`

There are 20 letters here, excluding the spaces. We first pick a divisor of 20, say 4, call it the *fence size*. Next, we'll write down the 20 letters into 4 columns:

1	2	3	4
H	I	S	T
O	R	Y	O
F	M	A	T
H	E	M	A
T	I	C	S

And now we do a transpose operation:

1	H	O	F	H	T
2	I	R	M	E	I
3	S	Y	A	M	C
4	T	O	T	A	S

and write down the letters by row.

`HOFHTIRMEISYAMCTOTAS`

The problem of this method is we lost the delimitors. True, we could add the delimiters into the ciphers, but in that case we may not choose a good *fence size*. If we add back the two spaces, the total number of letters is 22, with only contains 1, 2, 11 and 22 as divisors. When choosing the fence size as something like 2, it may not have the best effect of encryption, as it is easier to recognize than other larger fence sizes. A good choice of fence size is usually 3 to 8.

One alternative solution is to add some delimiters that doesn't really represents anything. It is just there to keep a position. The receiver would remove the delimiters after the decoding process.

4.3 Graphs instead of Letters

At some moment, Latin letters are not enough for security purposes. Because they are texts! Even though they are somewhat difficult to break down, one might still guess they have some meaning. However, there is a special way of substitution, that even normal people saw those ciphers, they could hardly recognize they were actually ciphers. The graphical cryptosystem, a special branch of substitution, replaces letters with graphics. Probably one of the most famous graphical cryptosystem example is in Arthur Conan Doyle's novel *Sherlock Holmes*, "The Dancing Men".

Figure 5: Graphic codes "Dancing Men" in *Sherlock Holmes* [3]

As Conan Doyle wrote in his novel, "The object of those who invented the system has apparently been to conceal that these characters convey a message, and to give the idea that they are the mere random sketches of children" [3]. Without seeing a lot of these texts, one may not understand that these graphical symbols are actually meaningful ciphers. However, once we realized these are cipher codes, we could apply the *frequency analysis* method to break them down, as I performed in the previous chapter.

4.4 Book Codes

The *Book Code*, or *book cipher*, involves an extra book when encoding or deciphering it. Usually a dictionary, or a booklet, but contains a lot of common words.

The method of book code is somewhat similar to the *Telephone Keypad Cipher*, but instead of using two digits to represent a single letter, it uses a pair (or a 3-tuple) of numbers to represent a word. For a word we want to encrypt, we look over the book to see if this word appears somewhere. Assume we found the word is on page X.

If the book we used is a dictionary, we could count that word appears on which item on that page, is it the first one, the second one, or the fifteenth one? Take that number as Y. We then use a pair of number (X, Y) to represent a word.

If the book is something like a novel, we could find which line is it in, and also the position of the word in that line. For example, the 4^{th} line and the 5^{th} word, then we use $(X, 4, 5)$ to represent this word.

As you see, there are various types of book ciphers, since there are billions of books in the world. It is usually difficult to determine which book is used to encode. Book ciphers are usually easy to recognize because the numbers appear in pairs or 3-tuples. However, the difficulty to break down a book cipher usually falls to find which book they are using.

5 Ending

The need of secure transmitting is there all the time, that is why over 3000 years, Cryptography is evolving. Hundreds of types of cryptosystems were created, and most of the traditional cryptosystems are broken down, but all of them became a pride page in the book of Cryptography history.

Looking forward to the post-computer era, people created public key encryption. New cryptosystems like *RSA* and *AES-256* appeared. Nevertheless, as what Sherlock Holmes said, "What one man can invent another can discover" [3]. Quantom Computing is trying to break down those encryptions. Nevertheless, whenever a cryptosystem is down, some new system is born. I hope I could see the day that Cryptography evolves again in future.

References

[1] Harold. Abelson. *Blown to bits : your life, liberty, and happiness after the digital explosion*. Addison-Wesley, Upper Saddle River, NJ, 2008.

[2] R. F. Churchhouse. *Codes and ciphers : Julius Caesar, the Enigma, and the internet*. Cambridge University Press, Cambridge ;, 2002.

[3] Arthur Conan Doyle. The dancing men. `https://sherlock-holm.es/stories/html/danc.html`.

[4] Abraham Lincoln. The gettysburg address. `http://www.abrahamlincolnonline.org/lincoln/speeches/gettysburg.htm`.

[5] Daniel Rodriguez-Clark. Frequency analysis: Breaking the code. `https://crypto.interactive-maths.com/frequency-analysis-breaking-the-code.html`.

[6] Lou Sander. J38 telegraph key. `https://en.wikipedia.org/wiki/File:J38TelegraphKey.jpg`.

[7] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379,423, 1948-07.