رقم البحث هو : **01603**

مقال (3)

عنوان المقالة : **Cryptanalysis of Vigenère cipher**

أسم المقرر ورقمه : التشفير **Comp308**

المستوى / الفرقة : الثالث

البرنامج/الشعبة : علوم حاسب

استاذ المقرر: د/حاتم بهيج & د/ضياء نصر

Introduction:

We will take a brief history about (**vigenere cipher**) , it was first described in 1553 and is cipher known for its simplicity to understand and use.

(Blaise de vigenere)

It was unbreakable for over **300** years.

This is why it has the name '**The Indecipherable cipher**'.

It belongs to the class of **polyalphabetic** ciphers.

The definition of vigenere cipher:

The Vigenere cipher is a method of encrypting alphabetic text by using a series of interwoven Caesar ciphers, based on the letters of a keyword. It employs a form of polyalphabetic substitution. (s.1)

So, The Algebraic description of vigenere cipher :

(s.1) (s.2 → p.44)

If n is a positive integer , $M = C = K = (Z_{26})^n$

Where: (M is the message , C is cipherText and K= e,d are the key**)** , we define:

$$E_e((m_1, m_2, \ldots, m_n)) = (m_1+k_1, m_2+k_2, \ldots, m_n+k_n);$$

$$D_d((c_1, c_2, \ldots, c_n)) = (c_1-k_1, c_2-k_2, \ldots, c_n-k_n);$$

where ,operation are performed in Z_{26} i.e., $(m_1+k_1 \bmod 26, m_2+k_2 \bmod 26, \ldots, m_n+k_n \bmod 26)$

$(c_1-k_1 \bmod 26, C_2-k_2 \bmod 26, \ldots, C_n-k_n \bmod 26)$

First , write a single line m . secondly , on the line below write your key , The first letter of the key is added to the first letter of the plaintext, mod 26 and so on through the first m letters of the plaintext. For the next m letters of the plaintext, the key letters are repeated. This process continues until all the plaintext sequence is encrypted. So, the general equation of the encryption, as shown above is :

-**Encryption** E , use **key** e , written as **: $C_i = E_e (M_i) = (M_i + K_i) \bmod 26$**

And to decrypt a message , we put the ciphertext (which we had get it form encryption method) in the first line and the key in the second line We subtract it from each other mod 26.we will use the general equation of decryption which is:

-**Decryption** D, use **key** d, written as**: $M_d = D_d (C_i) = (C_i - K_i + 26) \bmod 26$**

Note: A cipher is called (monoalphabetic) if "each alphabetic character is mapped to a unique alphabetic character".

Now, we will take an example on encryption and decryption:

We will consider the set of alphabet. So, we need to $Z_{26} = \{0,1,2, \ldots, 25\}$.

Letter	a	b	c	d	e	f	g	h	I	j	k	l	m
Code	0	1	2	3	4	5	6	7	8	9	10	11	12
Multiplicative inverse modulo 26	-	1	-	9	-	21	-	15	-	3	-	19	-

Letter	n	o	p	q	r	s	t	u	v	w	x	y	z
Code	13	14	15	16	17	18	19	20	21	22	23	24	25
Multiplicative inverse modulo 26	-	-	7	-	23	-	11	-	5	-	17	-	25

Ex: if length of key = 5 , e = (2 , 5 , 1 , 10 , 20)

Note : the plaintext is String and all capital letters are with out space.

M : TOMORROW AT THE SUNSET

M:	**T**	**O**	**M**	**O**	**R**	**R**	**O**	**W**	**A**	**T**	**T**	**H**	**E**	**S**	**U**	**N**	**S**	**E**	**T**
	19	14	12	14	17	17	14	22	0	19	19	7	4	18	20	13	18	4	19
e:	2	5	1	10	20	**2**	**5**	**1**	**10**	**20**	2	5	1	10	20	**2**	**5**	**1**	**10**
c:	21	19	13	24	11	19	19	23	10	13	21	12	5	2	14	15	23	5	3

+

c:	**V**	**T**	**N**	**Y**	**L**	**T**	**T**	**X**	**K**	**N**	**V**	**M**	**F**	**C**	**O**	**P**	**X**	**F**	**D**

First letter→ $E_e(M_i) = (M_i + e_i) \bmod 26 = (19+2) \bmod 26 = 21 \bmod 26 = 21$ → (V)

Second letter = (14 + 5) mod 26 = (19) mod 26 = 19 →(T)

Third letter =(12 + 1) mod 26 = (13) mod 26 = 13 →(N)

=(14+10) mod 26 = 24 →(Y) | =(17 + 20) mod 26 = (37) mod 26 = 11 →(L)

=(17 + 2) mod 26 = 19 →(T) | =(14 + 5) mod 26 = 19 →(T)

=(22+1) mod 26 = 23 →(X) | =(0+10) mod 26 = 10 →(K)

=(19+20) mod 26 = (39) mod 26 = 13 →(N) | =(19+2) mod 26 = 21 →(V)

(7+5)mod26 →(M) | (4+1)mod26 →(F) | (18+10)mod26 →(C) | (14)mod26 →(O)

(13+2)mod26 →(P)| (18+5)mod26 →(X)| (4+1)mod26 →(F)| (3)mod26 →(D)

So, the encryption $E_e(M_i) = (M_i + e_i) \bmod 26 =$ (VTNYLTTXKNVMFCOPXFD)

Let's do the decryption for it:

c:	**V**	**T**	**N**	**Y**	**L**	**T**	**T**	**X**	**K**	**N**	**V**	**M**	**F**	**C**	**O**	**P**	**X**	**F**	**D**
	21	19	13	24	11	19	19	23	10	13	21	12	5	2	14	15	23	5	3
e:	2	5	1	10	20	**2**	**5**	**1**	**10**	**20**	2	5	1	10	20	**2**	**5**	**1**	**10**
	19	14	12	14	17	17	14	22	0	19	19	7	4	18	20	13	18	4	19
M:	**T**	**O**	**M**	**O**	**R**	**R**	**O**	**W**	**A**	**T**	**T**	**H**	**E**	**S**	**U**	**N**	**S**	**E**	**T**

First letter→ $M_d = D_d(C_i) = (C_i - K_i + 26) \bmod 26 = (21-2+26) \bmod 26 = 19$→(T)

second letter→ = (19-5+26) mod 26 = 30 mod 26 = 14 →(O)

Third letter =(13-1+26) mod 26 = (12) mod 26 = 12 →(M)

=(24-10+26) mod 26 = 14 →(O) | =(11-20+26) mod 26 = 17 →(R)

=(19-2+26) mod 26 = 17 →(R) | =(19-5+26) mod 26 = 14 →(O)

=(23-1+26) mod 26 = 22 →(W) | =(10-10+26) mod 26 = 0 →(A)

=(13-20+26) mod 26 = (19) mod 26 = 19 →(T) | =(21-2+26) mod 26 = 19 →(T)

(12-5+26) mod 26 →(H) | (5-1+26) mod 26 →(E) | (2-10+26) mod 26 →(S)

(14-20+26) mod26 →(U) | (15-2+26) mod 26 →(N)| (23-5+26) mod 26 →(S)

(5-1+26) mod 26 →(E) | (3-10+26) mod 26 →(T)

So, the decryption=$D_e(C_i)=(C_i - e_i + 26) \bmod 26$ =(TOMORROWATTHESUNSET)

Now, let's Discuss the security of Vigenere cipher:

As, we Know the longer the keyword , the more difficult it will be for a third person to crack it . especially against Brute-Force Attack.

Brute-force attack: The attacker tries every possible key on a piece of ciphertext until an intelligible translation into plaintext is obtained. On average, half of all possible keys must be tried to achieve success. (s.2 → p.31)

→**Important** : the message must always be written without any space , because Of the spaces could help a third person to decode your m with the keyword.

Q:what is the problem of this cipher or why it is bad?

-Because by todays, standard it is **easily broken.** Once the key length is found, a cryptanalyst can apply frequency Analysis secret key will repeat So, words will repeat. we utilize that some letters are used more frequently in the English language

---

Let's take a look about the types of vigenere cipher attack:

Type of attack	Known to Cryptanalyst	Examples
Ciphertext Only	• Encryption algorithm , Ciphertext	-All ciphers
Known Plaintext	• E , C (to be decoded) $,(m_i ,c_i)$ for some I $E(m_i) = c_i$	-Analyst may know that certain plaintext patterns will appear in m. Ex. Postscript file always begin with the same pattern.
Chosen Plaintext	• E_e , c (to be decoded), and (c, m) • c : supposed ciphertext chosen by cryptanalyst, • m =Dd (c) decrypted plaintext generated with e	-attacker accesses encryption machine/software

So, let's know more about Cryptanalysis of vigenere cipher: (s.3 → p.32)

- First step to find the keyword length.
- Divide the m into many shift cipher encryptions.
- To solve the resulting shift ciphers we Applying the frequency analysis

So, what is the definition of cryptanalysis how to find the length of the key?

→**Cryptanalysis:** Cryptanalytic attacks rely on the nature of the algorithm plus perhaps some knowledge of the general characteristics of the plaintext or even some sample plaintext–ciphertext pairs. This type of attack exploits the characteristics of the algorithm to attempt to deduce a specific plaintext or to deduce the key being used. (s.2 → p.31)

-There is two methods to find the length of the key:

1) Kasisky Test. 2) Index of coincidence (friedman).

Now , let's know how kasisky test find key length:

We will encrypt two identical segments of plaintext to the same ciphertext, if they appear in the text at the index δ , ($\delta \equiv 0 \pmod m$, m is the key length).

we will try to do this algorithm to get the length: first, search for pairs of identical segments of length at >=3 , second, record distances between the two segments δ_1 , δ_2. finally m divides gcd(δ_1 ,δ_2 , ….).

*** Let's write a program to cryptanalysis the Vigenere using any technique.**

I use (know plaintext attack) which enable getting the key if we get some letters in ciphertext and there corresponding letters in the plaintext (the number of them at least greater than or equal the length of the key) and by that we can **cryptanalysis the Vigenere cipher.**

Note : I use this general equation in (keyword_function) to get the key:

$$K_i = (C_i - M_d + 26) \bmod 26$$

Where: (M is the message , C is cipherText and K= e,d are the key**)**

-I took the above example to increase confirmation :

His data were → message ; TOMORROWATTHESUNSET

Key = (2 , 5 , 1 , 10 , 20) = (CFBKU)

As we know the message with capital letters and with out space.

The code:

I used these libraries to build the program.

```
#include<iostream>
#include<string>
#include<algorithm>
using namespace std;
```

- Now, we will explain the funtions we will use

-This function takes (message, key_word) As parameters from type string and repeated the key to make its length as long as the plainText of message and return it.

```
string length_key_function(string message, string key_word)
{
    int size = message.size();
    for (int i = 0; true; i++) {
        if (size == i)
            i = 0;
        if (key_word.size() == message.size())
            break;
        key_word.push_back(key_word[i]);
    }
    return key_word;
}
```

-This function (encryption_fn) takes (message , keyword) as parameters From type string and use the general Equation of encryption to return the cipherText.

```
string encryption_function(string message, string keyword)
{
    string cipher_text;
    for (int i = 0; i < message.size(); i++) {
        char c = (message[i] + keyword[i]) % 26;
        /*A = 65  (i push it back) to make sure the range from 0 to 25 */
        c += 'A';
        cipher_text.push_back(c);
    }
    return cipher_text;
}
```

-This function (decryption_fn) takes (cipherText , keyword) as parameters From type string and use the general Equation of decryption to return the Old message.

```
string decryption_function(string cipher_text, string keyword) {
    string original_message;
    for (int i = 0; i < cipher_text.size(); i++) {
        char m = (cipher_text[i] - keyword[i] + 26) % 26;
        /*A = 65  (i push it back) to make sure the range from 0 to 25 */
        m += 'A';
        original_message.push_back(m);
    }
    return original_message;
}
```

-In this function we use known plain Text attack to get the key .

-It takes some letters from the message , there corresponding of the cipherText and the length of the key as parameters

```
string keyword_function(string message, string cipherText, int length_of_key) {
    string keyword;
    for (int i = 0; i < length_of_key; i++) {
        char e = (cipherText[i] - message[i] + 26) % 26;
        /*A = 65  (i push it back) to make sure the range from 0 to 25 */
        e += 'A';
        keyword.push_back(e);
    }
    return keyword;
}
```

and use a general equation to get the key to return the it as string.

Note: the length of the key must be equal or less than the length of the message which you want to attack.

- **In main function :**

We make it more efficient as we can see if we press (1) we can get a ready example Or press (2) to use known plaintext attack to get the key

```
void main(){
    do {
        int x;
        cout << "==>Press (1) to see a ready example . \n";
        cout << "==>Press (2) if you want to use known plain text attack to get the key.\n";
        cout << "==>Press (0) if you want to exit . \n";
        cin >> x;
        if (x == 1)
        {
```

The display of pressing (1):

```
==>Press (1) to see a ready example .
==>Press (2) if you want to use known plain text attack to get the key.
==>Press (0) if you want to exit .
1

 The plain text we will encrypte it : ( TOMORROWATTHESUNSET ) and the length of the key : 5 ( CFBKU ).

* the encryption ==> Ciphertext : VTNYLTTXKNVMFCOPXFD

* the decryption ==> old message : TOMORROWATTHESUNSET

* by using known plain text ,the key :CFBKU
```

The display of pressing (2):

```
==>Press (1) to see a ready example .
==>Press (2) if you want to use known plain text attack to get the key.
==>Press (0) if you want to exit .
2
===>Note , the message and the keyword that you will enter
    those will be with capital letters with out any spaces .


===>Please , Enter your message : tomor

===>Please , Enter the ciphet text : vtnyl

===>Please , Enter the length of the key : 5

* by using known plain text ,the key is :CFBKU
```

Conclusion:

In the end, we have learned what viginere cipher is And we took an example to explain more how to encrypt and decrypt the message , We discussed his security and the types of attacks on it . And I made a program that encrypts and decrypts messages, breaks the code and get its key.

Sources:

(s.1) https://en.wikipedia.org/wiki/Vigen%C3%A8re_cipher

(s.2) (Book) Cryptography-and-Network-Security-Principles-and- Practice-6th-Edition-Shannon.ir (The author of the book : William stallings)

(s.3) (Book) [Douglas_R._Stinson]_Cryptography_Theory_and_Prac(BookFi)