Cryptography with Liberty Basic

101: Introduction to Cryptography

By Onur Alver (*CryptoMan*)

INTRODUCTION

It has been noted that there are a number of Liberty Basic Forum members who have shown some interest in cryptography. There are also some members who have developed some crypto software with Liberty Basic and contributed these to the Liberty Basic Community. However, generally these works belong to simple or naive cryptography class and have no real protection capability. Therefore, we have decided to publish a series of articles about cryptography with practical applications and examples. It should be noted that Liberty Basic is especially well suited for cryptography which was a surprise for me when I first discovered certain undocumented features. In these series of articles, I will share my experience from my professional life and academic experience in the university as an invited lecturer on cryptography.

WHAT IS CRYPTOGRAPHY?

There are many descriptions you can find on internet and you can find a compilation of some of them below:

- Cryptograpy is the principles, means, and methods of rendering information unintelligible, and for restoring encrypted information to intelligible form.
- Or, it can be defined as, the conversion of data into a secret code for protection of privacy using a specific algorithm and a secret key. The original text, or "plaintext" is converted into a coded equivalent called "ciphertext" via an encryption algorithm. The ciphertext can only be decoded (decrypted) using a predefined secret key.
- It is the science of providing security for information through the reversible transformation of data. It is a science of great antiquity. (Julius Caesar used a simple letter substitution cipher that still bears his name.) The development of digital computing revolutionized cryptography, has made today's highly complex and secure cryptographic systems possible.
- The primary goal of cryptography is to conceal data to protect it against unauthorized third-party access by applying encryption. The more theoretical or mathematical effort is required for an unauthorized third party to recover data, the stronger is the encryption.

I believe that the above paragraphs give you a clear picture of cryptography. It is based on the premise that whatever communication channel you are using, be it paper and envelope over traditional postal system, voice / data transmission over landline telephone lines, terrestial radio communications, cellular phones, sattelite communications, going online to check your daily trusted internet email system, in fact any messenger system, is unsecure and uprotected.

What we mean from this is the following: *it is possible to listen to or read your communications and possibly alter it during transmission*. Therefore, it is essentially insecure to keep your messages on web based email systems, or send it via your popular email system from your PC, or discuss any sensitive commercial information in the chat systems, or talk with your secret lover on your mobile phone, etc. All of this can be intercepted, recorded and altered and may be used against your personal or commercial interests.

So, if you have any secrets to hide, which all of us do to some extent, you need cryptography.

TERMINOLOGY

In order to discuss cryptography, we must first learn about the basic terminology of the cryptography which is given below:

Plaintext : This is our open message which we wish to hide. For example your login password to the computer, or your PIN number for your ATM card, or location of the treasure chest full of gold coins.

Ciphertext: This is the scrambled transformation of plaintext using a specific algorithm and encryption key. It must be such that there will be a known feasible way to obtain the plaintext without knowing the algorithm and the key which you must keep secret.

Key: This is the element of a cryptographic system which defines the schedule of transformations in such a unique way that it will not be possible to deduce the plaintext even if the algorithm used is known but when this special secret value is unknown. Generally, it will get harder to break the cipher as the key gets longer.

Encryption: The process of transforming plaintext into ciphertext using an algorithm and a key, i.e. scrambling the message to make it unintelligble.

Decryption: The process of transforming ciphertext back to plaintext using an algorithm and a key, i.e. descrambling an enciphered message to an understanble format.

Algorithm: The systematic method of applying various transformations and transitions using computer functions and subroutines to process data in a deterministic way to achive the objectives of cryptography. An example of an algorithm is substituting a different ASCII code of a string character with another character by looking up a translation table and selecting a different translation table for every position in the string.

Confidententiality: The need to make message unreadable by others.

Integrity: The need to make sure the message is unchanged by others.

Authentication: The need to identify the origin of the message.

Non-Repudiation: The need that neither the sender nor the receiver of a message be able to deny the

transmission.

Hash Value: The need to make a short summary of a long message by applying a one way transformation algorithm to produce a value which can be predictibly produced again and again but not being able to come back to the original text under any circumstance.

CLASSICAL CRYPTO SYSTEMS

We will start by examining classical crypto systems to illustrate the concepts of cryptography. Many Liberty Basic Forum contributors are using some of the more simple, naive techniques that will be described in the beginning of this part.

There are two techniques : ***Substitution*** and ***Transposition***. Discussion in this article will be limited to ***Substitution***. ***Transposition*** will be discussed in a future article.

A ***substitution*** technique is a system where the letters of plaintext are ***replaced*** by other letters or numbers or symbols.

Caesar Cipher

- The oldest known use of substitution cipher was by Julius Caesar who was using this technique to scramble his messages to his commanders. In this technique, characters are shifted by an arbitrary number. For example:

```
PLAIN:    !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ
CIPHER:   56789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ!"#$%&'()*+,-./01234
```

Liberty BASIC example CAESAR.BAS

```
PLAINKEY$=""
CRYPTKEY$=""
SHIFT=20

FOR I=0 TO 255
  PLAINKEY$=PLAINKEY$+CHR$(I)
  CRYPTKEY$=CRYPTKEY$+CHR$( (I+SHIFT) MOD 255)
NEXT I

PRINT "PLAIN :";MID$(PLAINKEY$,33,71)
PRINT "CIPHER:";MID$(CRYPTKEY$,33,71)

PLAINTEXT$="ATTACK AT 23:45 25 JUN 2001"
```

```
CIPHERTEXT$=""
FOR I=1 TO LEN(PLAINTEXT$)
     CurrentChar$=MID$(PLAINTEXT$,I,1)
     CIPHERTEXT$ = CIPHERTEXT$ + CHR$( ASC( CurrentChar$ )+SHIFT )
NEXT I


DECIPHEREDTEXT$=""
FOR I=1 TO LEN(CIPHERTEXT$)
        CurrentChar$=MID$(CIPHERTEXT$,I,1)
        DECIPHEREDTEXT$ = DECIPHEREDTEXT$ + CHR$( ASC( CurrentChar$ )-
SHIFT )
NEXT I
```

Execution of CAESAR.BAS

```
PLAIN : !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^
_`abcdef
CIPHER:456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqr
stuvwxyz
PLAIN TEXT:ATTACK AT 23:45 25 JUN 2001
CIPHERTEXT:UhhUW_4Uh4FGNHI4FI4^ib4FDDE
DECIPHERED:ATTACK AT 23:45 25 JUN 2001
```

Caesar Cipher is the simplest cipher and probably most of us discovered and played with this when we were in primary school. Surprisingly enough, many people think derivatives of this cipher system, such as the ***XOR Cipher***, to be one of the most incredible and difficult to break cipher systems in use.

XOR Cipher

Liberty BASIC example XORCIPHER.BAS

```
PLAIN TEXT:ATTACK AT 23:45 25 JUN 2001
CIPHERTEXT:¾«¾´ß¾«ßÍÌÅËÊßÍÊßµª±ßÍÏÏÎ
DECIPHERED:ATTACK AT 23:45 25 JUN 2001

PLAINTEXT$="ATTACK AT 23:45 25 JUN 2001"
CIPHERTEXT$=""
FOR I=1 TO LEN(PLAINTEXT$)
        CurrentChar$=MID$(PLAINTEXT$,I,1)
        CIPHERTEXT$ = CIPHERTEXT$ + CHR$( ASC(CurrentChar$) XOR 255 )
NEXT I
```

```
DECIPHEREDTEXT$=""
FOR I=1 TO LEN(CIPHERTEXT$)
        CurrentChar$=UPPER$(MID$(CIPHERTEXT$,I,1))


  DECIPH
EREDTEXT$ = DECIPHEREDTEXT$ + CHR$( ASC(CurrentChar$) XOR 255 )
NEXT I

PRINT "PLAIN TEXT:";PLAINTEXT$
PRINT "CIPHERTEXT:";CIPHERTEXT$
PRINT "DECIPHERED:";DECIPHEREDTEXT$
```

Execution of XORCIPHER.BAS

```
PLAIN TEXT:ATTACK AT 23:45 25 JUN 2001
CIPHERTEXT:¾«¾´ß¾«ßÍÌÅËÊßÍÊßµª±ßÍÏÏÎ
DECIPHERED:ATTACK AT 23:45 25 JUN 2001
```

This looks to be a vastly improved encryption technique compared to ***Caesar Cipher*** but it is the same thing only shifting characters by 256. There is really not much difference shifting by 3, 13 or 256.

Monoalphabetic Ciphers

Improvement over ***Caesar*** and ***XOR Ciphers*** can be made by creating a table of random substitution of characters instead of simply shifting N characters because when one character is discovered in the Caesar cipher, the shifting factor N can be determined and the whole cipher breaks. Making a random substitution table makes it more difficult it to crack.

```
PLAIN  "0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ:. "
CIPHER "LNOP4KQ6RSTFVEWZ:M.01X2H3G5A789BY CUDIJ"
```

Liberty BASIC example MONOALPHA.BAS

```
PLAINKEY$="0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ:. "
CRYPTKEY$="LNOP4KQ6RSTFVEWZ:M.01X2H3G5A789BY CUDIJ"

PLAINTEXT$="ATTACK AT 23:45 25 JUN 2001"

CIPHERTEXT$=""
FOR I=1 TO LEN(PLAINTEXT$)
    FOR J=1 TO LEN(PLAINKEY$)
        CurrentChar$=UPPER$(MID$(PLAINTEXT$,I,1))
```

```
        CurrentPos$ =MID$(PLAINKEY$,J,1)
        IF CurrentChar$=CurrentPos$ THEN
             CIPHERTEXT$ = CIPHERTEXT$ + MID$( CRYPTKEY$, J, 1)
             EXIT FOR
        END IF
    NEXT J
NEXT I

DECIPHEREDTEXT$=""
FOR I=1 TO LEN(CIPHERTEXT$)
    FOR J=1 TO LEN(CRYPTKEY$)
        CurrentChar$=UPPER$(MID$(CIPHERTEXT$,I,1))
        CurrentPos$ =MID$(CRYPTKEY$,J,1)
        IF CurrentChar$=CurrentPos$ THEN
             DECIPHEREDTEXT$ = DECIPHEREDTEXT$ + MID$( PLAINKEY$, J, 1
)
             EXIT FOR
        END IF
    NEXT J
NEXT I

PRINT "PLAIN TEXT:";PLAINTEXT$
PRINT "CIPHERTEXT:";CIPHERTEXT$
PRINT "DECIPHERED:";DECIPHEREDTEXT$
```

Execution of MONOALPHA.BAS

```
PLAIN TEXT:ATTACK AT 23:45 25 JUN 2001
CIPHERTEXT:T88TV1JT8JOPD4KJOKJ09HJOLLN
DECIPHERED:ATTACK AT 23:45 25 JUN 2001
```

CRYPTANALYSIS

Cryptanalysis is the science of breaking the ciphers. The most famous case in history is allied effort centered in Bletchley Park in England for breaking the German cipher Enigma. This effort led to development of the first digital computer in history. Contrary to popular belief, ENIAC is not the first computer but it was this computer which was used to break the German Enigma.

We will not go too deep into this part of the cryptography but will touch this briefly to illustrate why the previously discussed algorithms are not safe.

There are two ways to break the ciphers. First method is the **Brute Force Method** where you use a high speed computer to test all possible keys and try the outcome to see if an intelligible result can be

obtained. This can be automatically tested by checking the resultant words from an electronic dictionary. The second method is the **`Analytical Method`** where you use various mathematical and statistical techniques to discover the encryption key.

The most important analytic technique is called ***Frequency Analysis***. The following is the relative frequency of each letter in English alphabet:

Letter	Frequency (%)	Letter	Frequency (%)
E	12.75	U	3.00
T	9.25	P	2.75
R	8.50	M	2.75
N	7.75	Y	2.25
I	7.75	G	2.00
O	7.50	V	1.50
A	7.25	W	1.50
S	6.00	B	1.25
D	4.25	X	0.50
L	3.75	K	0.50
C	3.50	Q	0.50
H	3.50	Z	0.25
F	3.00	J	0.25

If you have a sufficiently large encrypted text and if you make the statistical analysis of repetition of each encrypted character and symbol, you will be able to guess most of the characters directly and for the others only 2 to 4 possible selection sets remain. It is rather like solving a crossword puzzle.

Next you can use the knowledge of ***digrams and trigrams in English***. The design of the relative occurrance percentage of diagrams in English is as follows. You make a table of digrams from your cipher text and sort them according to their relative occurances and correlate this with the digram table given below:

AN 1.81	ON 1.83
AT 1.51	OR 1.28
ED 1.32	RE 1.90
EN 1.53	ST 1.22
ER 2.31	TE 1.30
ES 1.36	TH 3.21
HE 3.05	TI 1.28
IN 2.30	

You know that the most frequent character will be the space character. Hence you can easily find the spaces and first two characters (E's and T's) very easily in the text. After substituting these, you can open

the dictionary and discover the two or three letter words involving E and T like BE, TO, IT, AT, etc. When you find the O's, I's, A's, and B's you can back to substitute. After this, you can use your digram table to discover R, N, V, D, S, H positions. By now your solution text will start looking quite meaningful, afterward it is like completing a half filled crossword puzzle.

From this you are now understanding the concept of algorithms leaking the statistics of the letters which is yielding to breaking of those ciphers and the need to make ciphers which are stronger and stronger to hide this statistical relationship.

Polyalphabetic Ciphers

The first improvement over ***Monoalphabetic Ciphers*** is by using what is known as ***Polyalphabetic Ciphers***. ***Polyalphabetic Ciphers*** have not just one table but multiple tables:

```
0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ:.
LNOP4KQ6RST FVEWZ:M.01X2H3G5A789BYCUDIJ
4KQ6RSTFVEWL7IJ89BYCU DNOPZ:M.01X2H3G5A
59B YCUDIJLA78NOP4KQ6RSTFVEWZ:M.01X2H3G
```

The above example has 3 tables but you can easily make 30 or 300 or 30000 tables. The more tables you have the stronger it becomes as the statistical relationship gets progressively divorced from the encrypted results. If you have a system with

1. the number of tables equal to the number of characters in the message to be encrypted, and
2. you are able to protect these tables at the sending and receiving end and never use these tables ever again, and
3. if these tables were formed with perfect randomisation, i.e. their generation is unpredictable,

then this cipher is known as **ONE TIME PAD** and is unbreakable. However, it is very difficult to operate due to key management and key distribution problems.

Liberty BASIC example POLYALPHA.BAS

```
POLYDEPTH=5:DIM CRYPTKEY$(POLYDEPTH)
PLAINKEY$=   "0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ:. "
CRYPTKEY$(1)="LNOP4KQ6RSTFVEWZ:M.01X2H3G5A789BY CUDIJ"
CRYPTKEY$(2)="4KQ6RSTFVEWL7IJ89BYCU DNOPZ:M.01X2H3G5A"
CRYPTKEY$(3)="59B YCUDIJLA78NOP4KQ6RSTFVEWZ:M.01X2H3G"
CRYPTKEY$(4)="VEWZ:M.01X2H3LNOP4KQ6RSTFG5A789BY CUDIJ"
CRYPTKEY$(5)="4KQL7IJ89BYCU DNO6RSTFVEWPZ:M.01X2H3G5A"
PLAINTEXT$="ATTACK AT 23:45 25 JUN 2001"
CIPHERTEXT$=""
```

```
FOR I=1 TO LEN(PLAINTEXT$)
    FOR J=1 TO LEN(PLAINKEY$)
        CurrentChar$=UPPER$(MID$(PLAINTEXT$,I,1))
        CurrentPos$ =MID$(PLAINKEY$,J,1)
        IF CurrentChar$=CurrentPos$ THEN

    CIPHERTEXT$ =
CIPHERTEXT$ + MID$( CRYPTKEY$( (I MOD POLYDEPTH)+1), J, 1)
             EXIT FOR
        END IF
    NEXT J
NEXT I
DECIPHEREDTEXT$=""
FOR I=1 TO LEN(CIPHERTEXT$)
    FOR J=1 TO LEN(CRYPTKEY$(1))
        CurrentChar$=UPPER$(MID$(CIPHERTEXT$,I,1))
        CurrentPos$ =MID$(CRYPTKEY$( (I MOD POLYDEPTH)+1),J,1)
        IF CurrentChar$=CurrentPos$ THEN
             DECIPHEREDTEXT$ = DECIPHEREDTEXT$ + MID$( PLAINKEY$, J, 1
)
             EXIT FOR
        END IF
    NEXT J
NEXT I
PRINT "PLAIN TEXT:";PLAINTEXT$
PRINT "CIPHERTEXT:";CIPHERTEXT$
PRINT "DECIPHERED:";DECIPHEREDTEXT$
```

Execution of POLYALPHA.BAS

```
PLAIN TEXT:ATTACK AT 23:45 25 JUN 2001
CIPHERTEXT:W:8YVUG2.JQ D7KABMA00TJQL49
DECIPHERED:ATTACK AT 23:45 25 JUN 2001
```

CONCLUSION

As you can see from the output of ***POLYALPHA.BAS***, polyalphabetic encryption is a marked improvement over monoalphabetic encryption, but still not quite perfect. You will note that 2 is translated into Q twice because the same table is used by chance. When the opponent discovers this sort of faults, the breaking of the cipher starts. In fact, one of the techniques used by allies is KNOWN PLAINTEXT ATTACK**.** The allies discovered that messages coming from Uboats were starting with a weather report. Using this knowledge and TRAFFIC ANALYSIS, i.e. by discovering the position of the submarine by triangulation in a way similar to GPS used today, they knew the position, time and day of the transmission.

Once they knew the weather at that location it was not hard to guess the plaintext. Thus the moral of the story is that you have to make your tables in such a way that this sort of chance leakage is minimized or totally eliminated.

Therefore, in the next article in this series we will start with the techniques of Transposition, Compression and ***Diffusion***. Later we will progress into modern cryptography with DEA: Data Encryption Algorithm used in ANSI DES standard and its variation TRIPLE DES, also referred to as 3DES.

We will also present a DES implemention written purely in Liberty Basic.

CryptoMan

---

www.verisoft.com
onur@verisoft.com
