# Cryptography Coursework 2 - One-Time Pad

ricardo.moreira@acad.pucrs.br

April 28, 2015

**Abstract**

This paper describes the cryptanalysis of One-Time Pad cipher for the Cryptography course classwork n.o 2. A basic explanation on how One-Time Pad cipher works and how the sample ciphers could be braked.

## 1 Introduction

The One-Time Pad cipher is unbreakable if used correctly. The key should be random and they were printed on pads of paper that could be torn off after use so that each key would be used only once. Although unbreakable there are some practical problems that prevented it from being widely used.

## 2 One-Time Pad

A one-time pad (OTP) was first described by Gilbert Vernam in 1917 for use in automatic encryption and decryption of telegraph messages [3].

The Vernam Cipher is a stream cipher defined on the alphabet $A = \{0, 1\}$. A binary message $m_1 m_2 \ldots m_t$ is operated on by a binary key string $k_1 k_2 \ldots k_t$ of the same length to produce a ciphertext string $c_1 c_2 \ldots c_t$ where:

$$c_i = m_i \oplus k_i, 1 \leq i \leq t \tag{1}$$

If the key string is randomly chosen and never used again, the Vernam cipher is called a one-time pad.

| | |
|---|---|
| Plain: | 0 1 1 0 1 1 1 |
| Key: | 1 0 1 1 0 1 0 |
| Cipher: | 1 1 0 1 1 0 1 |

Table 1: Example of OTP cipher.

OTP is an encryption technique that cannot be cracked if used correctly used:

1. key is truly random

2. key is used only one-time

3. key is as long as the message

When a truly random key is combined with a plaintext, the result is a truly random ciphertext. To find key or plaintext, an adversary only has the random ciphertext at his disposal.

Shannon's definition of perfect secrecy states that a ciphertext should reveal no information about plaintext. A cipher $(E, D)$ over $(K, M, C)$ has a perfect secrecy if:

$$\forall m_0, m_1 \in M(len(m_0) = len(m_1)) \wedge \forall c \in C$$
$$Prob(E(k, m_0) = c) = Prob(E(k, m_1) = c)$$

The theorem of perfect secrecy requires that $|K| >= |M|$. Key must be at least as big as the message length which is one of the problems of OTP to be widely use.

As an example, lets assign each letter a numerical value: "A" is 0, "B" is 1, and so on. In the example on tables below, the technique is to combine the key and the message using modular addition. The numerical values of corresponding message and key letters are added together, modulo 26.

| | | | | | | |
|---|---|---|---|---|---|---|
| Message: | | H | E | L | L | O |
| | | 7 | 4 | 11 | 11 | 14 |
| Key: | | X | M | C | K | L |
| | + | 23 | 12 | 2 | 10 | 11 |
| | = | 30 | 16 | 13 | 21 | 25 |
| mod 26 | = | 4 | 16 | 13 | 21 | 25 |
| Ciphertext: | | E | Q | N | V | Z |

Table 2: One-time pad example encrypt.

| | | | | | | |
|---|---|---|---|---|---|---|
| Ciphertext: | | E | Q | N | V | Z |
| | = | 4 | 16 | 13 | 21 | 25 |
| Key: | | X | M | C | K | L |
| | - | 23 | 12 | 2 | 10 | 11 |
| | = | -19 | 4 | 11 | 11 | 14 |
| mod 26: | | 7 | 4 | 11 | 11 | 14 |
| Message: | | H | E | L | L | O |

Table 3: One-time pad example decrypt.

To demonstrate that given a ciphertext, an adversary cannot distinguish if message is $m_0$ or $m_1$, with enough computation power we could find that the key "XMCKL" produces the plaintext "HELLO", but also that the key "TQURI" would produce the plaintext "LATER", an equally plausible message:

| Ciphertext: | | E | Q | N | V | Z |
|---|---|---|---|---|---|---|
| | = | 4 | 16 | 13 | 21 | 25 |
| Key: | | T | Q | U | R | I |
| | - | 19 | 16 | 20 | 17 | 8 |
| | = | -15 | 0 | -7 | 4 | 17 |
| mod 26: | | 11 | 0 | 19 | 4 | 17 |
| Message: | | L | A | T | E | R |

Table 4: One-time pad example of multiple keys producing valid messages.

# 3 How to attack OTP

If the key string is reused there are ways to attack the system. For example if $c_1c_2 \dots c_t$ and $c'_1c'_2 \dots c'_t$ are two ciphertext strings produced by the same key string $k_1k_2 \dots k_t$ then:

$$c_i = m_i \oplus k_i \ , \ c'_i = m'_i \oplus k_i \text{ and } c_i \oplus c'_i = m_i \oplus m'_i \tag{2}$$

Know, lets consider what happens when we $\oplus$ a letter character with "spaces": $'A' \oplus ' ' =' a'$ and $'a' \oplus ' ' =' A'$. If we combine this behavior with the property expressed in equation 2 we can explore the ciphers to find letter characters from the $\oplus$ operation. Every time we find a valid letter we have only two options:

- The plaintext character is a space ' '.
- The plaintext character is the opposite case: $'a' \rightarrow' A'$ or $'A' \rightarrow' a'$.

This knowledge allows to calculate the probability of each letter considering the result of the $\oplus$ operation between the know ciphers. This is only valid if we hold the assumption that the ciphertexts were encrypted with the same key. We basically create a probability array measuring the result of $\oplus$ operations on ciphertexts.

Let's consider just the first letters of the 9 sample ciphertexts from the classwork: $0x39$ $0x4D$ $0x56$ $0x50$ $0x4D$ $0x5C$ $0x5A$ $0x50$ $0x5C$. If we $\oplus$ each character with each other cipher to compute the probability of each character at the given position. The valid letters we find and their frequency are presented in table 3. The result characters are: ' ', 'T', 'O', 'I', 'T', 'E', 'C', 'I', 'E'.

By following this approach we can find the majority of letters from the original plaintext as presented in appendix B. To find the key, we can use the property $k = m \oplus c$.

# References

[1] One-time pad *One-time pad — Wikipedia, The Free Encyclopedia*, Online; accessed 28-April-2015, 2015.

| Cipher | Letter frequency | Result |
|---|---|---|
| 0 | ' ': 8, 'E': 2, 'I': 2, 'T': 2, 'C': 1, 'O': 1 | ' ' |
| 1 | ' ': 1, 'T': 1 | 'T' |
| 2 | ' ': 1, 'O': 1 | 'O' |
| 3 | ' ': 1, 'I': 1 | 'I' |
| 4 | ' ': 1, 'T': 1 | 'T' |
| 5 | ' ': 1, 'E': 1 | 'E' |
| 6 | ' ': 1, 'C': 1 | 'C' |
| 7 | ' ': 1, 'I': 1 | 'I' |
| 8 | ' ': 1, 'E': 1 | 'E' |

Table 5: Probability for the first letter.

[2] Cryptography: Theory and Practice, Stinson, Douglas R., CRC Press, Inc., 1995.

[3] Handbook of Applied Cryptography, Alfred J. Menezes and Paul C. Van Oorschot and Scott A. Vanstone and R. L. Rivest, CRC Press, Inc. 1996

# A

## Coursework 2 ciphertexts

3939252352554c5f51592621294d5c5229382f5d454b485d4554413132275458482d3157415046495c5b2a435a46543527364d5059394847382a2b4b555746404a38202d4c525652455c2a

4d514c503134405f305f4e44292c41534a57414f2c2c2d4054544d5f552741424c5931345f415d2c2e4d4454405e504142522e31424a434c5f2a2b415a412f585a55452d2e20394941562a

56574023455d4449305b4745295b5b5a45385f59435a4457452645313244 5c5a454843404d585a2c4146464e32465441465545314 5c49574a435f573a

5057255759512 92c53594f515d435544484847522c2c4e5f4155575441274c45582d465d444c2e404b495859324f4f4227424131545644444d594e2e554a4b2c4757204947465f4c53572a

4d5625565f504c5e435f474f4d2c565f5a5b5d4e58492d4352494650504e59435954315d5b2047415e4758435349543553592e44595d4f504b5e4a4050244c5e4a4854424952584948 4b2a

5c57424f5847412c5c4e52554c5e41364f4a4a5a5949435059264 55d5e4842592d545e412854412c4c5a4f565927565c40534054455c2a43564e2b4d55415c4d413843445e485c4b533c

5a4b5c53455b4e5e515b4e5829454136594a4a5849425933494824425751505 84c4131504146484 95c4d4443325359454245 2e5e51394b524846424d555046435d4b20434157585d414b5722

505f255a5e41294a5f5e484529585a532957414e2c58445e45265450562b355e45485f34514f5b2c46495c523241495b4e45465453395e4a51592b2e574b5a5e405d57425c4b37
5c6f607168346a607f7e6221616d613668387c62607a6861206a6d7f7b697224

# B

## Coursework 2 decrypted ciphertext

```
    CAESA?   A?D   ?IGENERE   A?E  CIPHER? ?HA?   CA? B?   EOSIL?   BR ????
THIS  IS ?HE  ?ECON?   STRING TFAT  WAS  ?N?RY?TED  ?SI?G  O?E T?ME    ??J?
ONE TIME ?AD W?LL P?OVIDE A   I?HERTEXT O?L? A?TACK ?EC?RITY
IN THE  C?IPTO?RAPH?  CLASSES YAU WILL LE?R? H?W TO ?ON?USE OND ?O DIF ??K?
TO UNDERS?AND ?ISCR?TE PROBAB LGTY IS IMP?R?AN? TO U?DE?STANJ CR?PTOGR ??W?
ENGLISH L?TTER? FRE?UENCY ALL W? YOU TO B?E?K ?IGENE?E ?ND COESA? CIPH ??
CRYPTOGRA?HY I? PRE?ENT IN SE E?AL DIFFER?N? T?PES O? A?PLICOTIO?S NOW ??W??
IF YOU FO?ND T?E ON? TIME PAD  ZHEN YOU H?V? F?NISHE? T?IS  MOUR?EWORK
Every clo?d ha? a s?lver lini g
```