Cryptography (and Information Security)
6CCS3CIS / 7CCSMCIS

Prof. Luca Viganò

Department of Informatics
King's College London, UK

First term 2020/21

Lecture 3.3: Substitution ciphers — Vernam cipher, One-time pad

Table of contents I

Table of contents I

Table of contents I

Table of contents I

Table of contents I

Table of contents I

Table of contents I

Vernam cipher: XOR

- Gilbert Vernam (AT&T engineer, 1918) proposed a system where keyword is as long as plaintext and has no statistical relationship to it.
- It works on binary data (bits) rather than letters, using **XOR** $\oplus$:

$$\begin{aligned} 0 \oplus 0 &= 0 \\ 0 \oplus 1 &= 1 \\ 1 \oplus 0 &= 1 \\ 1 \oplus 1 &= 0 \end{aligned}$$

so that

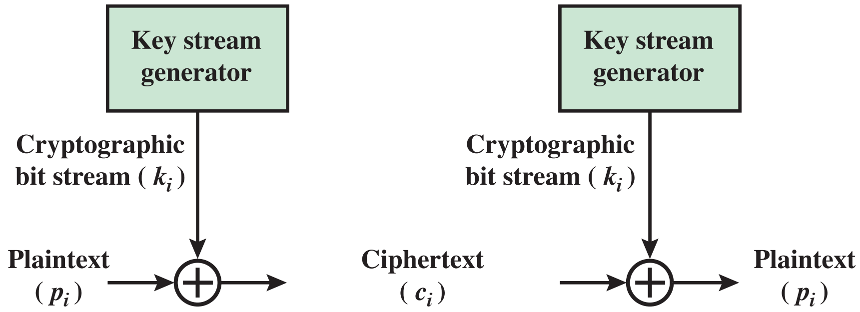
$$\begin{aligned} a \oplus a &= 0 \\ a \oplus 0 &= a \\ a \oplus b &= b \oplus a \\ a \oplus b \oplus b &= a \\ (a \oplus b) \oplus c &= a \oplus (b \oplus c) \end{aligned}$$

XOR can be used as polyalphabetic cipher:

$$\begin{aligned} P \oplus K &= C \\ C \oplus K &= P \end{aligned}$$

Vernam cipher: idea

$$
\begin{aligned}
c_i &= p_i \oplus k_i \\
p_i &= c_i \oplus k_i
\end{aligned}
$$



- $p_i/c_i/k_i$ = i^{th} binary digit of plaintext/ciphertext/key.
- Ciphertext generated by bitwise XOR of plaintext and key.
- Decryption: simply same bitwise operation (by properties of XOR).
- Essence of this cipher: means of construction of the key.
- Vernam proposed use of a running loop of tape that eventually repeated the key (hence: very long but repeating keyword).
- Difficult to break if key is long, but still breakable with sufficient ciphertext, use of known or probable plaintext sequences, or both.

A small example of how perfect secrecy is achievable

- Mr X is about to make a decision that will have serious repercussions on the share value of a company.
  - If he makes the decision “buy”, then the shares will increase in value.
  - If he makes the decision “sell”, then the shares will collapse.
- Suppose also that it is publicly known that Mr X will soon be transmitting one of these two messages to his broker.
  - Anyone who received this decision before the broker would have the opportunity to use that information to either make a profit or to avoid a disastrous loss.
- At any time, anyone is free to guess what the message will be and act accordingly.
  - They have a 50% chance of being right... such an action would be nothing more than gambling.

A small example of how perfect secrecy is achievable

- Mr X wants to be able to send his message over a **public** network.
- In order to protect their interests, Mr X and his broker decide to encrypt the message that will convey the decision.
- Since a substitution cipher would be easy to break with such a short (and predictable) message, they decide to use a system with two keys, K_1 and K_2 are **equally likely**.
    - K_1 encrypts "buy" to 0 and "sell" to 1:
      $$E_{K_1}(\text{"buy"}) = 0 \text{ and } E_{K_1}(\text{"sell"}) = 1.$$
    - K_2 encrypts "buy" to 1 and "sell" to 0:
      $$E_{K_2}(\text{"buy"}) = 1 \text{ and } E_{K_2}(\text{"sell"}) = 0.$$
- If the attacker intercepts a 0, then all that he can deduce is that the message might be "sell" if K_2 was used, or "buy" if K_1 was used.
- Since each key is equally likely, the attacker is forced to guess which key was used: the chances of guessing correctly are 50%.

A small example of how perfect secrecy is achievable

In essence:

- Before the ciphertext was intercepted, the attacker's only option was to try to guess the message.
- Once the ciphertext was intercepted, the attacker could also guess the key.
- Since the number of keys is the same as the number of messages, the chances of either guess being correct are equal.

This is **perfect secrecy** (but, as we will see, it comes at a price).

Table of contents I

One-time pad

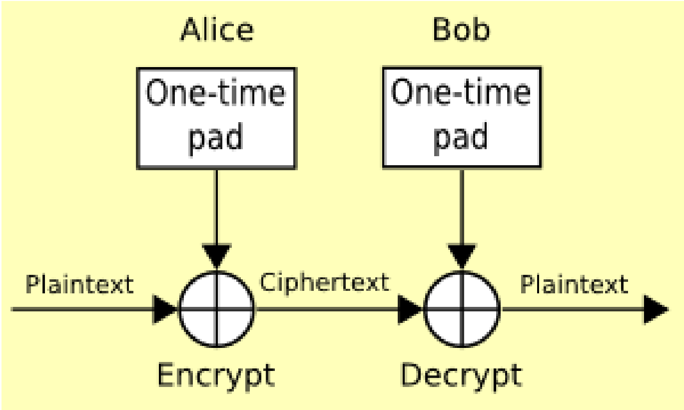
One-time pad

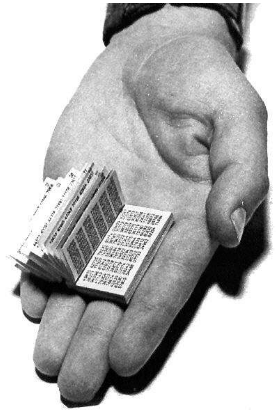
(improvement to Vernam proposed by Joseph Mauborgne, US Army Signal Corp officer)

Use a truly **random key** that is

- as long as the message, so that the key need not be repeated,
- used to encrypt and decrypt a single message, and then discarded.

- Each new message P requires a new key of same length as P.
- Produces random output with no statistical relation to plaintext.
- **Unbreakable: C contains no information whatsoever about P.** Only cryptosystem that exhibits so-called *perfect secrecy*.





One-time pad: example

- Exhaustive search of all possible keys yields many legible plaintexts, with no way of knowing which was the intended one, e.g.

ciphertext	A	N	K	Y	O	D	K	Y	U	R	E	P	F	J	B	Y	O	J	D	S	P	L	R	E	Y	I	U	N	O	F	D	O	I	U	E	R
key	O	W	Y	E	W	K	K	H	R	V	W	W	Y	Q	U	U	M	J	Q	P	E	H	Z	L	Q	G	K	F	B	M	W	K	B	U	T	G
plaintext	M	r	M	u	s	t	a	r	d	W	i	t	h	T	h	e	C	a	n	d	l	e	s	t	i	c	k	I	n	T	h	e	H	a	l	l
ciphertext	A	N	K	Y	O	D	K	Y	U	R	E	P	F	J	B	Y	O	J	D	S	P	L	R	E	Y	I	U	N	O	F	D	O	I	U	E	R
key	O	F	S	G	W	B	K	H	J	N	L	W	J	B	I	R	V	C	Z	I	C	D	M	A	Q	V	B	G	K	U	V	N	R	U	N	T
plaintext	M	i	s	s	S	c	a	r	l	e	t	W	i	t	h	T	h	e	K	n	i	f	e	I	n	T	h	e	L	i	b	r	a	r	y

- Suppose a cryptanalyst managed to find these two keys.
  - Two plausible plaintexts are produced.
  - Which is the correct decryption (i.e., which is the correct key)?
- If the actual key were produced in a truly random fashion, then cryptanalyst cannot say that one key is more likely than the other.
- Given any P of equal length to C, there is K that produces that P.

No patterns or regularities: if stream of characters that constitute K is truly random, then so will be stream of characters that constitute C.

One-time pad: practical difficulties

- Two fundamental practical difficulties:
  - Making large quantities of random keys.
  - Key distribution and protection, where for every message to be sent, a key of equal length is needed by both sender and receiver.
- Hence:
  - limited utility,
  - useful primarily for low-bandwidth channels requiring very high security (Moscow–Washington communication previously secured this way).

Listen carefully...

I shall say this only once.