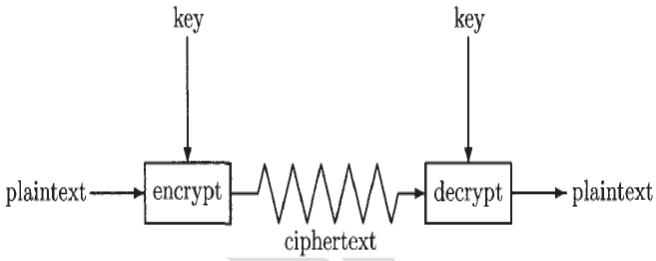
# Crypto Basics

The basic terminology of crypto includes the following:

• *Cryptology* — the art and science of making and breaking "secret codes."
• *Cryptography* — the making of "secret codes."
• *Cryptanalysis* — the breaking of "secret codes."
• *Crypto* — a synonym for any or all of the above and more.

- A *cipher* or *cryptosystem* is used to *encrypt* data. The original unencrypted data is known as *plaintext*, and the result of encryption is *ciphertext*. *Decrypting* the ciphertext to recover the original plaintext.
- A *key* is used to configure a cryptosystem for encryption and decryption.

In a ***symmetric* cipher**, the same key is used to encrypt and to decrypt, as illustrated by the black box cryptosystem in below Figure.



Crypto as a Black Box

In **public key crypto**, the encryption key is known as the *public key*, whereas the decryption key, which must remain secret, is the *private key*.

**[ Kerckhoffs's principle** goes as follows: A cryptographic system should be secure even if everything about the system, except the key, is public knowledge].

## Classic Crypto:

- Four classic ciphers:
  - Simple Substitution Cipher.
  - Double Transposition Cipher

- Codebook Cipher
- One-Time Pad

## Simple Substitution Cipher:

The message is encrypted by substituting the letter of the alphabet *n* places ahead of the current letter. For example, with $n = 3$, the substitution—which acts as the key—is

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

The convention that the plaintext is lowercase,and the ciphertext is uppercase. Using the key 3, encrypt the plaintext message:

**fourscoreandsevenyearsago**

to the resulting ciphertext is:

**IRXUVFRUHDAGVHYHABHDUVDIR**

To decrypt this simple substitution, look up the ciphertext letter in the ciphertext row and replace it with the corresponding letter in the plaintext row, or shift each ciphertext letter backward by three. The simple substitution with a shift of three is known as the **Caesar's cipher**.

- If we limit the simple substitution to shifts of the alphabet, then the possible keys are $n \in \{0,1,2,..., 25\}$.
- Attacker can suspect that received text was encrypted with a simple substitution cipher using a shift by *n*. Then he can try each of the 26 possible keys, decrypt the message, this **Brute force approach** is known as ***Exhaustive key search.***
- It's necessary that the number of possible keys be too large for the attacker to simply try them all in any reasonable amount of time.

**Keyspace:** Suppose attacker has a fast computer that's able to test $2^{40}$ keys each second. Then a keyspace of size $2^{56}$ can be exhausted in $2^{16}$ seconds, or about 18 hours, whereas a keyspace of size $2^{64}$ would take more than half a year for an exhaustive key search, and a keyspace of size $2^{128}$ would require more than nine quintillion years.

.

**Permutation:** Any permutation of the 26 letters will serve as a key.

For example, the following permutation, gives us a key for a simple substitution cipher:

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

- A simple substitution cipher can employ any permutation of the alphabet as a key, which implies that there are **26! = $2^{88}$ possible keys**.
- With attacker's superfast computer that tests $2^{40}$ keys per second, trying all possible keys for the simple substitution would take more than 8900 millennia.
- Attacker would expect to find the correct key half that time, or just 4450 millennia. Since $2^{88}$ keys is far more than attacker can try in any reasonable amount of time. The keyspace is big enough so that an **exhaustive key search** is infeasible.

## Cryptanalysis of a Simple Substitution:

Suppose attacker intercepts the following ciphertext, which he suspects was produced by a simple substitution cipher, where the key could be any permutation of the alphabet:

```
PBFPVYFBQXZTYFPBFEQJHDXXQVAPTPQJKTOYQWIPBVWLXTOXBTFXQWA
XBVCXQWAXFQJVWLEQNTOZQGGQLFXQWAKVWLXQWAEBIPBFXFQVXGTVJV
WLBTPQWAEBFPBFHCVLXBQUFEVWLXGDPEQVPQGVPPBFTIXPFHXZHVFAG
FOTHFEFBQUFTDHZBQPOTHXTYFTODXQHFTDPTOGHFQPBQWAQJJTODXQH
FOQPWTBDHHIXQVAPBFZQHCFWPFHPBFIPBQWKFABVYYDZBOTHPBQPQJT
QOTOGHFQAPBFEQJHDXXQVAVXEBQPEFZBVFOJIWFFACFCCFHQWAUVWFL
QHGFXVAFXQHFUFHILTTAVWAFFAWTEVOITDHFHFQAITIXPFHXAFQHEFZ
QWGFLVWPTOFFA
```

Assuming the plaintext is English, attacker can make use of the English letter frequency counts in Figure 2.2 together with the frequency counts for the ciphertext in (2.2), which appear in Figure 2.3.

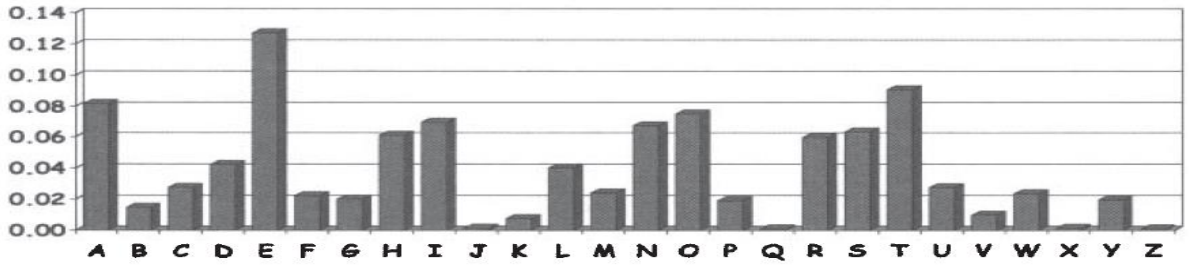



Figure 2.2: English Letter Frequency Counts

From the ciphertext frequency counts in Figure 2.3

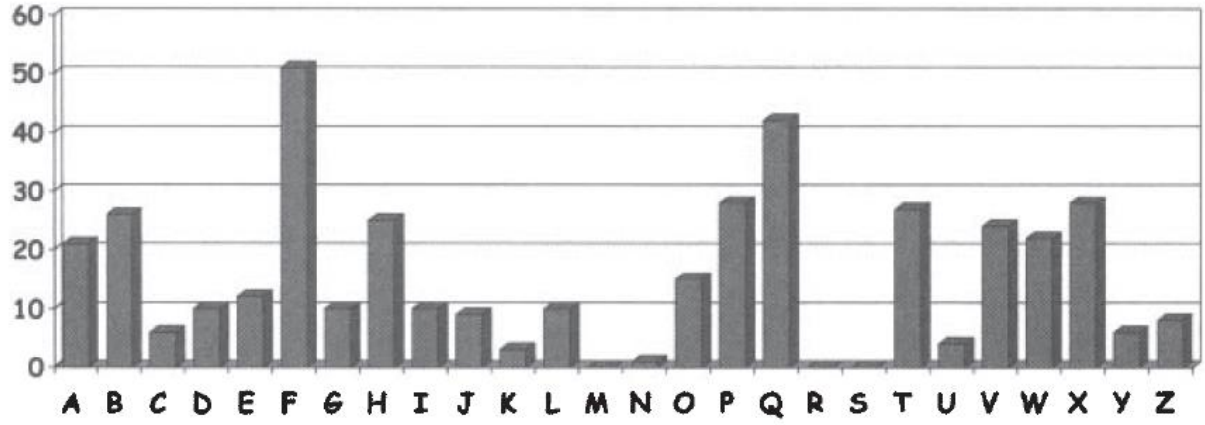



Figure 2.3: Ciphertext Frequency Counts

- "F" is the most common letter in the encrypted message and, according to Figure 2.2, "E" is the most common letter in the English language.
- Attacker therefore conclude that it's likely that "F" has been substituted for "E." Continuing in this manner, attacker can try likely substitutions until he recognizes words, at which point he can be confident in her guesses.

**Conclusion:** Above discussed attack on the simple substitution shows that a large keyspace is not sufficient to ensure security.

## Double Transposition Cipher:

- Write the plaintext into an array of a given size.
- Then permute the rows and columns according to specified permutations.
- For example, suppose consider the plaintext:

attackatdawn into a 3 x 4 array

$$\begin{bmatrix} a & t & t & a \\ c & k & a & t \\ d & a & w & n \end{bmatrix}$$

- Transpose (or permute) the rows according to (1,2,3) —> (3,2,1) and then transpose the columns according to (1,2,3,4) —> (4,2,1,3), we obtain

$$\begin{bmatrix} a & t & t & a \\ c & k & a & t \\ d & a & w & n \end{bmatrix} \longrightarrow \begin{bmatrix} d & a & w & n \\ c & k & a & t \\ a & t & t & a \end{bmatrix} \longrightarrow \begin{bmatrix} n & a & d & w \\ t & k & c & a \\ a & t & a & t \end{bmatrix}$$

- The ciphertext is then read from the final array:

**NADWTKCAATAT**

- The key consists of the size of the matrix and the row and column permutations.
- **Attackatdawn:** If anyone who knows the key can put the ciphertext into the appropriate sized matrix and undo the permutations to recover the plaintext.

  For example, to decrypt the ciphertext is first put into a 3 x 4 array. Then the columns are numbered as (4,2,1,3) and rearranged to (1,2,3,4), and the rows are numbered (3,2,1) and rearranged into (1,2,3), and we have recovered the plaintext.

$$\begin{bmatrix} N & A & D & W \\ T & K & C & A \\ A & T & A & T \end{bmatrix} \longrightarrow \begin{bmatrix} D & A & W & N \\ C & K & A & T \\ A & T & T & A \end{bmatrix} \longrightarrow \begin{bmatrix} A & T & T & A \\ C & K & A & T \\ D & A & W & N \end{bmatrix}$$

**Conclusion:** The double transposition appears to thwart an attack that relies on the statistical information contained in the plaintext, since the plaintext statistics are disbursed throughout the ciphertext.

## One-Time Pad:

- The one-time pad, which is also known as the Vernam cipher, is a provably secure cryptosystem.
- Let's consider an alphabet with only eight letters and the corresponding binary representation of letters appear in the below table.

  Note: **It's important to note that the mapping between letters and bits is not secret.**

| letter | e | h | i | k | l | r | s | t |
|---|---|---|---|---|---|---|---|---|
| binary | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |

- Onetime pad is used to encrypt the plaintext message

  **h e i l h i t l e r**.

  convert the plaintext letters to the **bit string** using above table.

  001 000 010 100 001 010 111 100 000 101

- The one-time pad key consists of a randomly selected string of bits that is the same length as the message.
- The key is then XORed with the plaintext to yield the ciphertext.
- Decryption is accomplished by XOR-ing the same key with the ciphertext.

## Example:

Consider that Alice has the key :

111 101 110 101 111 100 000 101 110 000

which is of the proper length to encrypt her message above. Then to encrypt, Alice computes the ciphertext as:

| | h | e | i | l | h | i | t | l | e | r |
|---|---|---|---|---|---|---|---|---|---|---|
| plaintext: | 001 | 000 | 010 | 100 | 001 | 010 | 111 | 100 | 000 | 101 |
| key: | 111 | 101 | 110 | 101 | 111 | 100 | 000 | 101 | 110 | 000 |
| ciphertext: | 110 | 101 | 100 | 001 | 110 | 110 | 111 | 001 | 110 | 101 |
| | s | r | l | h | s | s | t | h | s | r |

Converting these ciphertext bits back into letters, the ciphertext message to be transmitted is **srlhssthsr.**

Bob, receives Alice's message, he decrypts it using the same shared key and thereby recovers the plaintext:

| | s | r | l | h | s | s | t | h | s | r |
|---|---|---|---|---|---|---|---|---|---|---|
| ciphertext: | 110 | 101 | 100 | 001 | 110 | 110 | 111 | 001 | 110 | 101 |
| key: | 111 | 101 | 110 | 101 | 111 | 100 | 000 | 101 | 110 | 000 |
| plaintext: | 001 | 000 | 010 | 100 | 001 | 010 | 111 | 100 | 000 | 101 |
| | h | e | i | l | h | i | t | l | e | r |

**Different scenarios examples:**

1) Suppose that Alice has an enemy, Charlie, within her spy organization. Charlie claims that the actual key used to encrypt Alice's message is

   101 111 000 101 111 100 000 101 110 000.

   Bob decrypts the ciphertext using the key given to him by Charlie and obtains **killhitler** which is a wrong message.

| | s | r | l | h | s | s | t | h | s | r |
|---|---|---|---|---|---|---|---|---|---|---|
| ciphertext: | 110 | 101 | 100 | 001 | 110 | 110 | 111 | 001 | 110 | 101 |
| “key”: | 101 | 111 | 000 | 101 | 111 | 100 | 000 | 101 | 110 | 000 |
| “plaintext”: | 011 | 010 | 100 | 100 | 001 | 010 | 111 | 100 | 000 | 101 |
| | k | i | l | l | h | i | t | l | e | r |

2) Suppose that Alice is captured by her enemies, who have also intercepted the ciphertext. The captors ask Alice is to provide the key for this super-secret message. Alice claims that she is actually a double agent and to prove it she provides the "key".

   111 101 000 011 101 110 001 011 101 101.

   When Alice's captors "decrypt" the ciphertext using this "key," they find it as **helikesike and** Alice's captors will release her.

| | s | r | l | h | s | s | t | h | s | r |
|---|---|---|---|---|---|---|---|---|---|---|
| ciphertext: | 110 | 101 | 100 | 001 | 110 | 110 | 111 | 001 | 110 | 101 |
| "key": | 111 | 101 | 000 | 011 | 101 | 110 | 001 | 011 | 101 | 101 |
| "plaintext": | 001 | 000 | 100 | 010 | 011 | 000 | 110 | 010 | 011 | 000 |
| | h | e | l | i | k | e | s | i | k | e |

**Advantage**: If the key is chosen at random, and used only once, then an attacker who sees the ciphertext provides no meaningful information at all about the plaintext.

## Why that the one-time pad can only be used once?

**Scenario 1:** Suppose we have two plaintext messages P1 and *P2* and encrypted these as as $\mathbf{C_1} = \boldsymbol{P_1} \oplus \boldsymbol{K}$ **and** $\mathbf{C_2} = \boldsymbol{P_2} \oplus \boldsymbol{K}$, i.e two messages encrypted with the same "one-time" pad ***K***. In the cryptanalysis , this is known as a ***depth***. With one-time pad ciphertexts in depth,

$$C_1 \oplus C_2 = P_1 \oplus K \oplus P_2 \oplus K = P_1 \oplus P_2$$

and the **key has disappeared** from the problem. In this case, the ciphertext does yield some information about the underlying plaintext.

**Scenario 2:** Another way is considering an exhaustive key search. If the pad is only used once, then the attacker has no way to know whether the guessed key is correct or not. But if two messages are in depth, for the correct key, both putative plaintexts must make sense.

Let's consider an example of one-time pad encryptions **that are in depth**. Using the same bit encoding as in Table. Suppose

$$P_1 = \texttt{like} = 100\,010\,011\,000 \quad \text{and} \quad P_2 = \texttt{kite} = 011\,010\,111\,000$$

and both are encrypted with the same key ***K*** **= 110 011 101 111**. Then

| | l | i | k | e |
|---|---|---|---|---|
| $P_1$: | 100 | 010 | 011 | 000 |
| $K$: | 110 | 011 | 101 | 111 |
| $C_1$: | 010 | 001 | 110 | 111 |
| | i | h | s | t |

and

| | k | i | t | e |
|---|---|---|---|---|
| $P_2$: | 011 | 010 | 111 | 000 |
| $K$: | 110 | 011 | 101 | 111 |
| $C_2$: | 101 | 001 | 010 | 111 |
| | r | h | i | t |

- If the attacker knows that the messages are in depth, immediately he sees that the **second** and **fourth** letters of $P_1$ and $P_2$ are the same, since the corresponding ciphertext letters are identical.
- Now attacker can guess a putative message $P_1$ and check her results using $P_2$.
- Suppose that attacker (who only has $C_1$ and $C_2$) suspects that **$P_1$ = k i l l = 011010100100**. Then he can find the corresponding putative key:

| | k | i | l | l |
|---|---|---|---|---|
| putative $P_1$: | 011 | 010 | 100 | 100 |
| $C_1$: | 010 | 001 | 110 | 111 |
| putative $K$: | 001 | 011 | 010 | 011 |

and he can then use this ***K* to "decrypt" $C_2$** and obtain

| | | | | |
|---|---|---|---|---|
| $C_2$: | 101 | 001 | 010 | 111 |
| putative $K$: | 001 | 011 | 010 | 011 |
| putative $P_2$: | 100 | 010 | 000 | 100 |
| | l | i | e | l |

- Since this *K* does not yield a sensible decryption for $P_2$, attacker can safely assume that his guess for $P_1$ was incorrect.
- Eventually attacker guesses $P_1$ = **like** he will obtain the correct key *K* and decrypt to find $P_2$= **kite**, thereby confirming the correctness of the key therefore, the correctness of both decryptions.

**Project VENONA:**

- The VENONA project is an example of a real-world use of the one-time pad. In the 1930s and 1940s, spies from the Soviet Union who entered the United States brought with them one-time pad keys.
- When it was time to report back to their handlers in Moscow, these spies used their one-time pads to encrypt their messages, which could then be safely sent back to Moscow.
- These spies were extremely successful, and their messages dealt with the most sensitive U.S. government secrets of the time.
- In particular, the development of the first atomic bomb was a focus of much of the espionage. The Rosenbergs, Alger Hiss, and many other well known traitorsfigure prominently in VENONA messages.
- The Soviet spies were well trained and never reused the key, yet many of the intercepted ciphertext messages were eventually decrypted by American cryptanalysts.
- There was a flaw in the method used to generate the pads, so that, long stretches of the keys were repeated. As a result, many messages were in depth, which enabled the successful cryptanalysis of much VENONA traffic.

## Codebook Cipher:

- A classic codebook cipher is a dictionary-like book containing (plaintext) words and their corresponding (ciphertext) codewords.
- To encrypt a given word, the cipher clerk would simply look up the word in the codebook and replace it with the corresponding codeword.
- Decryption, using the inverse codebook, was equally straightforward.

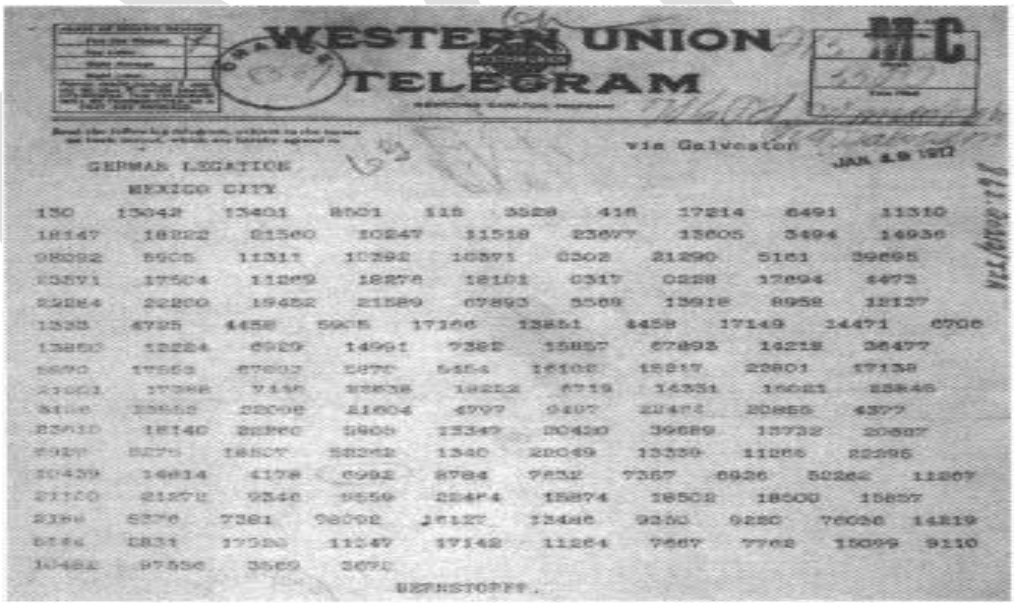
**Example**: Below table contains a famous codebook used by Germany during World War I(used to encrypt the famous Zimmermann telegram.)

Table 2.3: Excerpt from a German Codebook

| Plaintext | Ciphertext |
|---|---|
| Februar | 13605 |
| fest | 13732 |
| finanzielle | 13850 |
| folgender | 13918 |
| Frieden | 17142 |
| Friedenschluss | 17149 |
| ⋮ | ⋮ |

This codebook was used for encryption, while the corresponding inverse codebook, arranged with the 5-digit codewords in numerical order, was used for decryption. A codebook is a form of a substitution cipher.

- The German Foreign Minister, Arthur Zimmermann, sent an encrypted telegram to the German ambassador in Mexico City.
- The ciphertext message, which appears in below figure was intercepted by the British. At the time, the British and French were at war with Germany, but the U.S. was neutral.



- The Russians had recovered a damaged version of the German codebook, and the partial codebook had been passed on to the British.
- Through analyses, the British were able to fill in the gaps in the codebook so that by the time they obtained the Zimmermann telegram, they could decrypt it.
- The telegram stated that the German government was planning to begin "**unrestricted submarine warfare**" and had concluded that this would likely lead to war with the United States.
- As a result, Zimmermann told his ambassador that Germany should try to recruit Mexico as an ally to fight against the United States.
- The incentive for Mexico was that it would "reconquer the lost territory in Texas, New Mexico and Arizona."
- When the Zimmermann telegram was released in the U.S., public opinion turned against Germany and, after the sinking of the Lusitania, the U.S. declared war.

## Security:

- The security of a classic codebook cipher depends primarily on **the physical security** of the book itself. That is, the book must be protected from capture by the enemy.
- **Statistical attacks** analogous to those used to break a simple substitution cipher apply to codebooks, although the amount of data required is much larger.
- The reason that a statistical attack on a codebook is more difficult is due to the fact that **the size of the "alphabet" is much larger.**.

### Additive Book:

- Ciphers were subject to statistical attack, so codebooks **needed to be periodically replaced with new codebooks**. Since this was an expensive and risky process, techniques were developed to extend the life of a codebook. To accomplish this, a ***additive*** *book* was used.
- The codewords are all 5- digit numbers. Then the corresponding additive book would consist of a long list of randomly generated 5-digit numbers.

- After a plaintext message had been converted to a series of 5-digit codewords, a starting point in the additive book would be selected and beginning from that point, the sequence of 5- digit additives would be added to the codewords to create the ciphertext.
- To decrypt, the same additive sequence would be **subtracted from the ciphertext** before looking up the codeword in the codebook.
- The starting point in the additive book was selected at random by the sender and sent in the clear at the start of the transmission.
- This additive information was part of the ***message indicator,* or MI**.
- The MI included any non-secret information needed by the intended recipient to decrypt the message.
- If the additive material was only used once, the resulting cipher would be equivalent to a one-time pad and therefore, provably secure.
- If the additive was reused many times and, any messages sent with overlapping additives would have their codewords encrypted with the same key, where the key consists of the codebook and the specific additive sequence.

- Therefore, any messages with overlapping additive sequences could be used to gather the statistical information needed to attack the underlying codebook.
- The additive book dramatically increased the amount of ciphertext required to mount a statistical attack on the codebook, which effects the cryptographers.

## 2.3.8 Ciphers of the Election of 1876

- The contestants in the election were Republican Rutherford B. Hayes and Democrat Samuel J. Tilden.
- The Electoral College that determines the winner of the presidency.
- In the Electoral College, each state sends a delegation and for almost every state, the entire delegation is supposed to vote for the candidate who received the largest number of votes in that particular state.
- In 1876, the electoral college delegations of four states were in dispute.

- A commission of 15 members was appointed to determine which state delegations were legitimate, and thus determine the presidency.
- The commission decided that all four states should go to Hayes and he became president of the United States.
- Some months after the election, reporters discovered a large number of encrypted messages that had been sent from Tilden's supporters to officials in the disputed states.
- One of the ciphers used was a partial codebook together . with a transposition on the words.
- The codebook was only applied to important words and the transposition was a fixed permutation for all messages of a given length.
- The allowed message lengths were 10, 15, 20, 25, and 30 words, with all messages padded to one of these lengths. A snippet of the codebook appears in Table

Election of 1876 Codebook

| Plaintext | Ciphertext |
|---|---|
| Greenbacks | Copenhagen |
| Hayes | Greece |
| votes | Rochester |
| Tilden | Russia |
| telegram | Warsaw |
| ⋮ | ⋮ |

- The permutation used for a message of 10 words was

  9,3,6,1,10,5,2,7,4,8.

- One actual ciphertext message was

  **Warsaw they read all unchanged last are idiots can't situation**

  which was decrypted by undoing the permutation and substituting telegram for Warsaw to obtain

  **Can't read last telegram.**

  **Situation unchanged.**

  **They are all idiots.**

- A permutation of a given length was used repeatedly, many messages of particular length were in depth—with respect to the permutation as well as the codebook.
- A cryptanalyst could compare all messages of the same length, making it relatively easy to discover the fixed permutation, even without knowledge of the partial codebook.
- The overuse of a key can be an exploitable flaw.

## Modern Crypto History

- Late in the 20th century, cryptography became a critical technology for commercial and business communications as well, and it remains today as well.
- The Zimmermann telegram is one of the first examples that cryptanalysis has had in political and military affairs.

- In the Pacific theatre, the so-called Purple cipher was used for high level Japanese government communication. This cipher was broken by American cryptanalysts before the attack on Pearl Harbor, but the intelligence gained (code named MAGIC) provided no clear indication of the impending attack.
- The Japanese Imperial Navy used a cipher known as JN-25, which was also broken by the Americans, an inferior American force was able to halt the advance of the Japanese in the Pacific for the first time.
- In Europe, the German Enigma cipher (code named ULTRA) was a major source of intelligence for the Allies during the war.
- The Enigma was initially broken by Polish cryptanalysts. After the fall of Poland, these cryptanalysts escaped to France.
- The Polish cryptanalysts eventually made their way to England, where they provided their knowledge to British cryptanalysts.
- A British team that included the computing pioneer, Alan Turing, developed improved attacks on the Enigma.

An Enigma Cipher

Two fundamental cipher design principles: *confusion* and *diffusion.*

**Confusion**

- It is defined as obscuring the relationship between the plaintext and ciphertext.
- A simple substitution cipher and a one-time pad employ only confusion
- Confusion is provably secure

**Diffusion**

- It is the idea of spreading the plaintext statistics through the ciphertext.
- A double transposition is a diffusion-only cipher.
- Diffusion alone is not secure.

The National Bureau of Standards, or NBS, issued a request for cryptographic algorithms. The ultimate result of this process was a cipher known as the Data Encryption Standard, or DES and Public key cryptography was discovered) shortly after the arrival of DES.

## A Taxonomy of Cryptography

Three broad categories of ciphers: *symmetric* ciphers, *public key* cryptosystems, and *hash functions*.

***Symmetric* ciphers:**

- Modern symmetric ciphers can be subdivided into *stream ciphers* and *block ciphers*
- Stream ciphers generalize the one-time pad approach, sacrificing provable security for a key that is manageable.
- Block ciphers are the generalization of classic codebooks. In a block cipher, the key determines the codebook, and as long as the key remains fixed, the same codebook is used. Conversely, when the key changes, a different codebook is selected.
- Block ciphers are easier to optimize for software implementations, while stream ciphers are usually most efficient in hardware.

***Public key* cryptosystems**

- In public key crypto, encryption keys can be made public. For each public key, there is a corresponding decryption key that is known as a private key.
- Public key cryptography does not completely eliminate the key distribution problem, since the private key must be in the hands of the appropriate user, and no one else.

**Cryptographic hash functions:.**

- These functions take an input of any size and produce an output of a fixed size.
- If the input changes in one or more bits, the output should change in about half of its bits.
- It is computationally infeasible to find *any* two inputs that hash to the same output.

## A Taxonomy of Cryptanalysis

The goal of cryptanalysis is to recover the plaintext, the key, or both.

- ***Ciphertext only* attack:** If attacker only knows the algorithms and the ciphertext, then he must conduct a *ciphertext only* attack
- ***Known plaintext attack:*** Attacker might know some of the plaintext and observe the corresponding ciphertext. These matched plaintext-ciphertext pairs might provide information about the key.

- ***Chosen plaintext* attack:** Attacker can actually choose the plaintext to be encrypted and see the corresponding ciphertext.

  For example, Alice might forget to log out of her computer when she takes her lunch break. Attacker could then encrypt some selected messages before Alice returns. This type of "**lunchtime attack"** takes many forms.

  - ***Adaptively Chosen plaintext* attack**: Trudy chooses the plaintext, views the resulting ciphertext, and chooses the next plaintext based on the observed ciphertext.

- ***Related key* attacks:** The idea here is to look for a weakness in the system when the keys are related in some special way.

- ***Forward search (for Public Key Cryptography):*** Suppose attacker intercepts a ciphertext that was encrypted with Alice's public key. If attacker suspects that the plaintext message was either "yes" or "no," then she can encrypt both of these putative plaintexts with Alice's public key. If either matches the ciphertext, then the message has been broken.

*Note: The size of the **keyspace must be large enough** to prevent an attacker from trying all possible keys*