ΥΣ13 - Computer Security

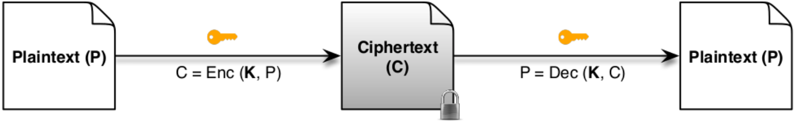
Symmetric Cryptography

Κώστας Χατζηκοκολάκης

Context

- **Goal**
  - Confidentiality
  - Alice wants to send a message P (plaintext) to Bob
  - Only Bob should be able to read it
- **Solution** : symmetric encryption
  - Share a key K with Bob
  - Only Alice and Bob should know the key
  - Alice constructs an (encrypted) message C (ciphertext) from P, K
  - Bob uses K to decrypt C and obtain P

Context



Correctness: $P = \mathrm{Dec}(K, \mathrm{Enc}(K, P)))$

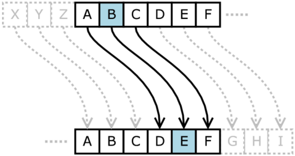
Context

Adversary model

- Knows everything except P, K
- Including all algorithms, protocols, conventions
  - **Important**: obscurity is not security
- Having all information public actually makes the system more secure

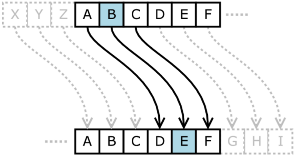
First attempt

- Caesar's cipher (50 BC)
  - Replace A→D, B→ E, . . .
  - In other words $C_i = P_i + K \mod 26$
  - $K = 3$ (or $K =$ ”D”) is the key



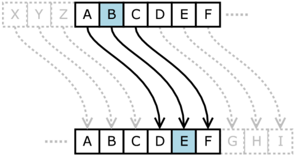
First attempt

- Caesar's cipher (50 BC)
  - Replace A→D, B→ E, . . .
  - In other words $C_i = P_i + K \mod 26$
  - $K = 3$ (or $K =$ "D") is the key
- Augustus Caesar used A→C, …
  - i.e. changed the key to $K =$"C"



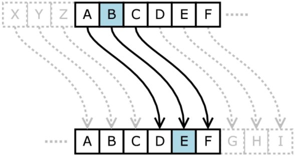
First attempt

- Caesar's cipher (50 BC)
  - Replace A→D, B→ E, . . .
  - In other words $C_i = P_i + K \mod 26$
  - $K = 3$ (or $K =$ "D") is the key
- Augustus Caesar used A→C, . . .
  - i.e. changed the key to $K =$"C"
- ROT13
  - $K = 13$ (decrypt is the same as encrypt)
  - Win XP registry keys!



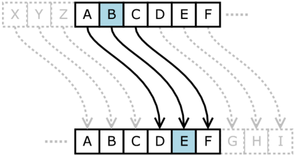
First attempt

- Generally : mono-alphabetic substitution cipher
  - use a single permutation of the alphabet
  - How can we break this?



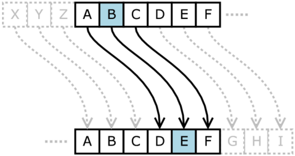
First attempt

- Generally : mono-alphabetic substitution cipher
  - use a single permutation of the alphabet
  - How can we break this?
- Frequency analysis
  - observe the frequence of each symbol in the ciphertext



First attempt

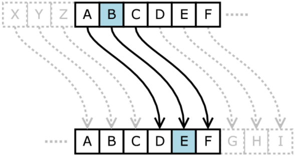
- Generally : mono-alphabetic substitution cipher
  - use a single permutation of the alphabet
  - How can we break this?



- Frequency analysis
  - observe the frequence of each symbol in the ciphertext
- How can we do better?

First attempt

- Generally : mono-alphabetic substitution cipher
  - use a single permutation of the alphabet
  - How can we break this?



- Frequency analysis
  - observe the frequence of each symbol in the ciphertext
- How can we do better?
  - Stream cipher : substitution depends on the character's position
  - Block cipher : encrypt many letters at once in a block

Vigenère cipher

An early stream cipher (1553)

- Idea
  - Key: **C**CCCCCCCCCCC… change to
  - Key: **WORD**WORDWORD…
- Frequency analysis much harder
  - Unbreakable for 300 years

Vigenère cipher

An early stream cipher (1553)

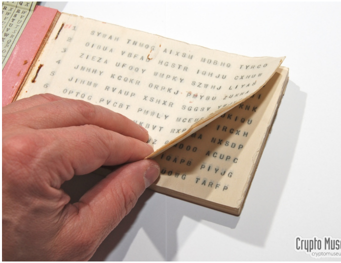
- Idea
  - Key: **C**CCCCCCCCCC… change to
  - Key: **WORD**WORDWORD…
- Frequency analysis much harder
  - Unbreakable for 300 years
- Problem
  - Repeated patters at multiples of the keyword length
  - Find out the keyword length
  - Then?

One time pad

- Repeating key letters was problematic
- Solution?

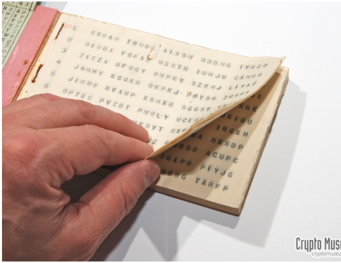
One time pad

- Repeating key letters was problematic
- Solution?
  - Key at least as big as the plaintext
  - Randomly chosen (uniformly)
  - Key: `AFEMIONOASNEPOZLMOIUW…`



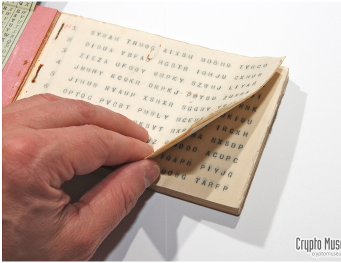
One time pad

- Repeating key letters was problematic
- Solution?
  - Key at least as big as the plaintext
  - Randomly chosen (uniformly)
  - Key: `AFEMIONOASNEPOZLMOIUW…`
- How good is this cipher?



One time pad

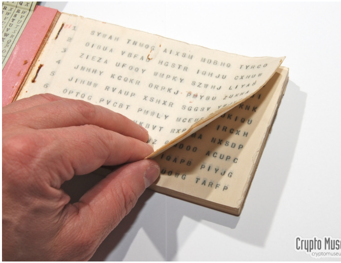
- Repeating key letters was problematic
- Solution?
  - Key at least as big as the plaintext
  - Randomly chosen (uniformly)
  - Key: `AFEMIONOASNEPOZLMOIUW…`



- How good is this cipher?
  - **Perfect!** : unconditional security
    - $p(P|C) = p(P)$ equivalently $p(C|P) = p(C|P')$

One time pad

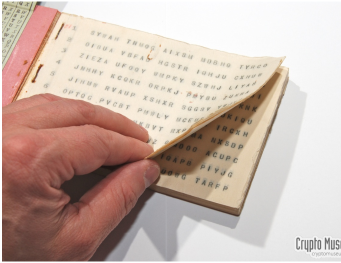
- Repeating key letters was problematic
- Solution?
  - Key at least as big as the plaintext
  - Randomly chosen (uniformly)
  - Key: `AFEMIONOASNEPOZLMOIUW...`



- How good is this cipher?
  - **Perfect!** : unconditional security
    - $p(P|C) = p(P)$ equivalently $p(C|P) = p(C|P')$
  - Idea: choose $P = 0|1$ arbitrarily, choose $K = 0|1$ uniformly
    - What is the probability that $P \oplus K = 0$?

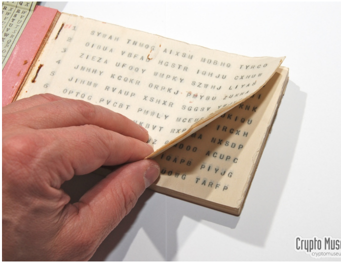
One time pad

- Repeating key letters was problematic
- Solution?
  - Key at least as big as the plaintext
  - Randomly chosen (uniformly)
  - Key: `AFEMIONOASNEPOZLMOIUW...`



- How good is this cipher?
  - **Perfect!** : unconditional security
    - $p(P|C) = p(P)$ equivalently $p(C|P) = p(C|P')$
  - Idea: choose $P = 0|1$ arbitrarily, choose $K = 0|1$ uniformly
    - What is the probability that $P \oplus K = 0$?
- Why "one time"?

One time pad

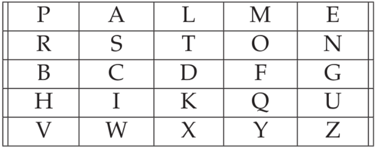


- Repeating key letters was problematic
- Solution?
  - Key at least as big as the plaintext
  - Randomly chosen (uniformly)
  - Key: `AFEMIONOASNEPOZLMOIUW...`
- How good is this cipher?
  - **Perfect!** : unconditional security
    - $p(P|C) = p(P)$ equivalently $p(C|P) = p(C|P')$
  - Idea: choose $P = 0|1$ arbitrarily, choose $K = 0|1$ uniformly
    - What is the probability that $P \oplus K = 0$?
- Why "one time"?
- Drawbacks?

Playfair Cipher

An early block cipher (1854)

- Key: 5x5 permutation of all letters (I/J combined)
- Encrypt pairs of letters (blocksize: 2 letters)

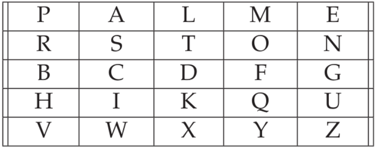


P	A	L	M	E
R	S	T	O	N
B	C	D	F	G
H	I	K	Q	U
V	W	X	Y	Z

Playfair Cipher

An early block cipher (1854)

- Key: 5x5 permutation of all letters (I/J combined)
- Encrypt pairs of letters (blocksize: 2 letters)
- Same row/column : replace by succeeding letters
  - AM → LE
- Different row/column : replace by opposite corners
  - LO → MT



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

Playfair Cipher

An early block cipher (1854)

- Key: 5x5 permutation of all letters (I/J combined)
- Encrypt pairs of letters (blocksize: 2 letters)
- Same row/column : replace by succeeding letters
  - AM → LE
- Different row/column : replace by opposite corners
  - LO → MT
- Much better than Vigenère
  - But how much better?
  - Change a single letter of plaintext?

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

Random Oracle

- Reverse question
  - what is an ideal cipher?

Random Oracle

- Reverse question
  - what is an ideal cipher?
- Random Oracle
  - Generate a random answer
  - Repeat it in future queries

Random Oracle

- Reverse question
  - what is an ideal cipher?
- Random Oracle
  - Generate a random answer
  - Repeat it in future queries
- Ideal ciphers
  - Stream : key → long keystream
  - Block : key → random permutation

Random Oracle

- Reverse question
  - what is an ideal cipher?
- Random Oracle
  - Generate a random answer
  - Repeat it in future queries
- Ideal ciphers
  - Stream : key → long keystream
  - Block : key → random permutation
- Good real cipher
  - indistinguishable from a suitable oracle
  - given certain abilities of the adversary

How can we create a good block cipher?

Principles

- Confusion
  - Drastic (non-linear) change to the input
  - Basic tool : substitution
  - Invertible function $\{0,1\}^n \to \{0,1\}^n$ (permutation of $\{0,1\}^n$)
    - For a subset of the block, eg 4 bits

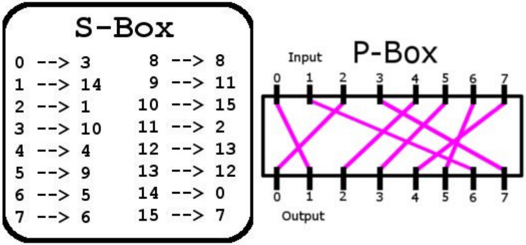
How can we create a good block cipher?

Principles

- Confusion
  - Drastic (non-linear) change to the input
  - Basic tool : substitution
  - Invertible function $\{0,1\}^n \to \{0,1\}^n$ (permutation of $\{0,1\}^n$)
    - For a subset of the block, eg 4 bits
- Diffusion
  - changing a single character of the input will change many characters of the output.
  - Basic tool : permutation of bits

How can we create a good block cipher?

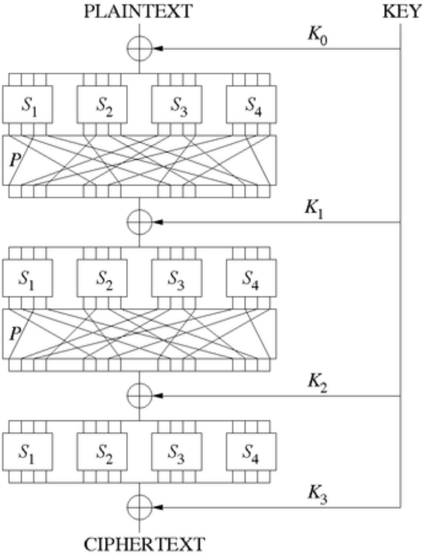
- Substitution (confusion)
- Permutation (diffusion)



S-Box

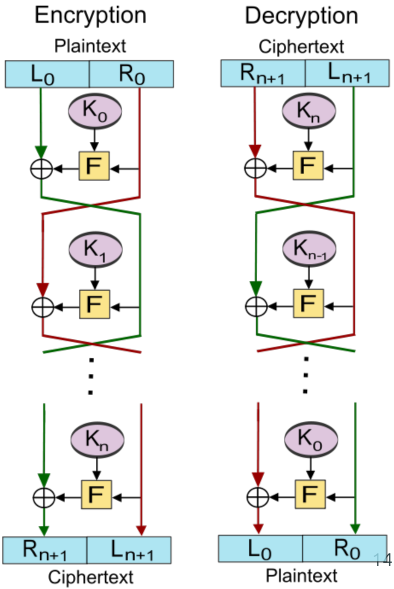
0	-->	3	8	-->	8
1	-->	14	9	-->	11
2	-->	1	10	-->	15
3	-->	10	11	-->	2
4	-->	4	12	-->	13
5	-->	9	13	-->	12
6	-->	5	14	-->	0
7	-->	6	15	-->	7

Substitution–permutation network



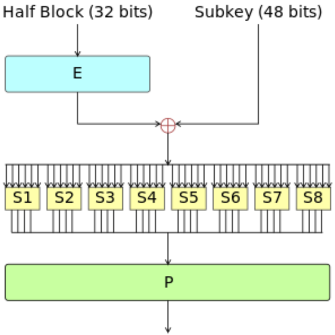
Feistel cipher

- No need for invertible F!
- IF F is a random function then
  - indist. from random permutation
  - 3 rounds: chosen plaintext
  - 4 rounds: chosen plaintext/ciphertext



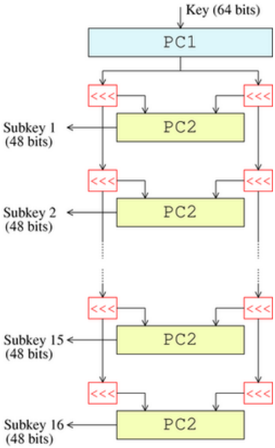
Data Encryption Standard (DES)

- IBM, 1975
- Feistel cipher
- 56bit keys
- 64bit block size



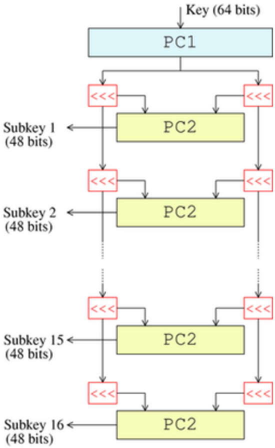
Data Encryption Standard (DES)

- IBM, 1975
- Feistel cipher
- 56bit keys
- 64bit block size



Data Encryption Standard (DES)

- IBM, 1975
- Feistel cipher
- 56bit keys
- 64bit block size
- Weaknesses
  - Brute force (< day)
  - Linear cryptanalysis



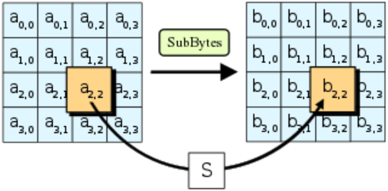
Advanced Encryption Standard (AES)

- NIST, 2001
  - Key: 128, 192, 256 bits
  - Block: 128bits
- SP-network: multiple rounds of
  - Substitution
    - SubBytes
  - Permutation
    - MixColums
    - ShiftRows
- No known practical attack

$$\begin{bmatrix} b_0 & b_4 & b_8 & b_{12} \\ b_1 & b_5 & b_9 & b_{13} \\ b_2 & b_6 & b_{10} & b_{14} \\ b_3 & b_7 & b_{11} & b_{15} \end{bmatrix}$$

Advanced Encryption Standard (AES)

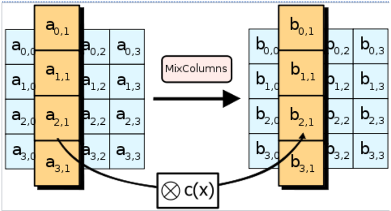
- NIST, 2001
  - Key: 128, 192, 256 bits
  - Block: 128bits
- SP-network: multiple rounds of
  - Substitution
    - SubBytes
  - Permutation
    - MixColums
    - ShiftRows
- No known practical attack



$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
SubBytes
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,1}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$
S

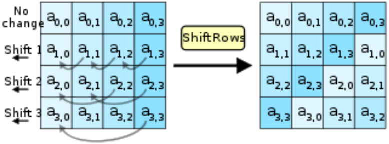
Advanced Encryption Standard (AES)

- NIST, 2001
  - Key: 128, 192, 256 bits
  - Block: 128bits
- SP-network: multiple rounds of
  - Substitution
    - SubBytes
  - Permutation
    - MixColums
    - ShiftRows
- No known practical attack



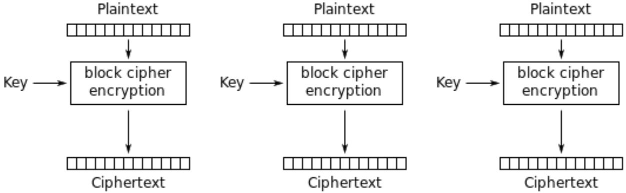
Advanced Encryption Standard (AES)

- NIST, 2001
  - Key: 128, 192, 256 bits
  - Block: 128bits
- SP-network: multiple rounds of
  - Substitution
    - SubBytes
  - Permutation
    - MixColums
    - ShiftRows
- No known practical attack

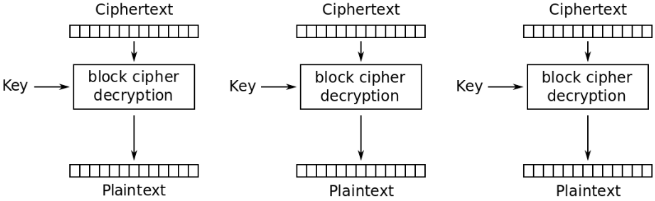

No change
Shift 1
Shift 2
Shift 3
$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
ShiftRows
$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,1}$ $a_{1,2}$ $a_{1,3}$ $a_{1,0}$
$a_{2,2}$ $a_{2,3}$ $a_{2,0}$ $a_{2,1}$
$a_{3,3}$ $a_{3,0}$ $a_{3,1}$ $a_{3,2}$

Mode of operation

Problem?

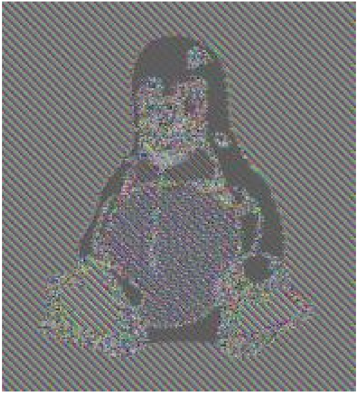


Electronic Codebook (ECB) mode encryption



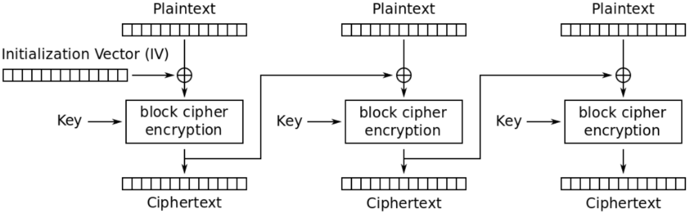
Electronic Codebook (ECB) mode decryption

Mode of operation

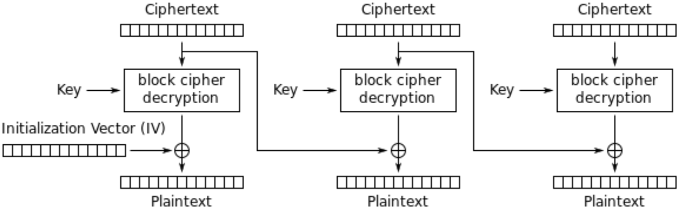


Patterns!

Mode of operation



Cipher Block Chaining (CBC) mode encryption



Cipher Block Chaining (CBC) mode decryption

References

- Ross Anderson, Security Engineering, Sections 5.1 - 5.5
- `https://blog.filippo.io/the-ecb-penguin/`