

# **MH4311 Cryptography**

## **Lecture 4**

### **Block Cipher (Part 1, Introduction)**

**Wu Hongjun**

# Lecture Outline

- **Classical ciphers**
- **Symmetric key encryption**
  - **One-time pad & information theory**
  - **Block cipher**
    - **Introduction**
    - **DES, Double DES, Triple DES**
    - **AES**
    - **Mode of Operations**
    - **Attacks**
  - **Stream cipher**
- **Hash function and Message Authentication Code**
- **Public key encryption**
- **Digital signature**
- **Key establishment and management**
- **Introduction to other cryptographic topics**

# Lecture Outline

- **Theoretical security & computational security**
- **Practical symmetric key ciphers**
- **Introduction to block cipher**

# Recommended Reading

- **CTP Section 3.1, 3.2**
- **HAC Section 7.1, 7.2.1**
- **Wikipedia:**
  - **Block cipher**

**[http://en.wikipedia.org/wiki/Block\\_cipher](http://en.wikipedia.org/wiki/Block_cipher)**

# Information-theoretical Security vs Computational Security

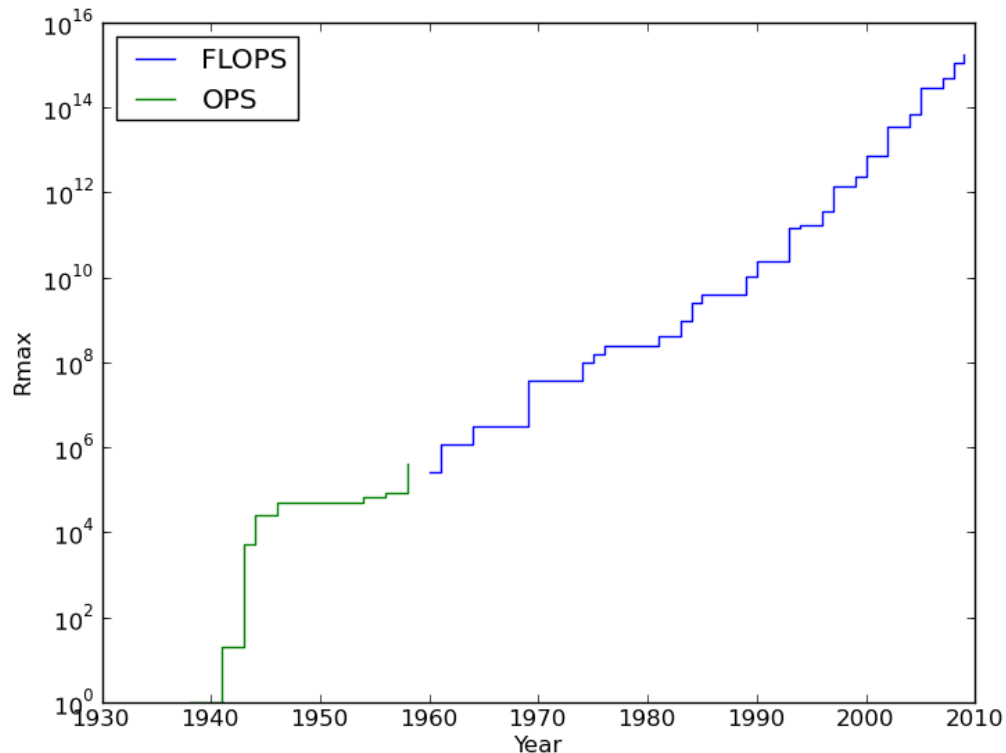
- **Information-theoretically secure**
  - Also called “unconditionally secure”, “perfectly secure”
  - A cryptosystem is unconditionally secure if it is secure against the attacks using **unlimited computing power**
  - **One example: One-time pad**
    - Key length is as long as the message length => inconvenient to use for many applications
  - **Another example: substitution cipher, and each key is used to encrypt a message with less than 25 letters**
    - Key has to be changed very frequently

# Information-Theoretical Security vs Computational Security

- **Computationally secure**
  - A cryptosystem is computationally secure if it cannot be broken with **the current computing technology** within **a given period of time**.
    - What is the computing technology today?
    - What will be the computing technology in the next 30, 50, or 100 years?
      - We may try to predict, but there is great uncertainty

# The Fastest Computers in History

<http://en.wikipedia.org/wiki/Supercomputer>



1938: **1 OPS** Germany  
1943: **5 KOPS** Post Office Research Station,  
Bletchley Park, UK  
1961: **1.2 MFLOPS**; Los Alamos National  
Laboratory, USA  
1984: **2.4 GFLOPS** ; Scientific Research  
Institute of Computer  
Complexes, USSR  
1997: **1.338 TFLOPS**; Sandia National  
Laboratories, USA  
2008: **1.026 PFLOPS**; Los Alamos National  
Laboratory, USA  
2018(?): **1 EFLOPs**  
????: **1 ZFLOPs** → required for two-  
week weather modeling

**FLOPS : FLoating Point Operations Per Second**

**Kilo:10<sup>3</sup> Mega:10<sup>6</sup> Giga:10<sup>9</sup> Tera:10<sup>12</sup> Peta:10<sup>15</sup> Exa:10<sup>18</sup> Zeta:10<sup>21</sup>**

# Computing power today

- **In the past several decades, the computing power almost doubled every 1.5 years (for the same price)**
  - It is related to the Moore's law
- **Currently, Intel Core-i3, i5, i7 CPUs**
  - Widely used in notebook and desktop computers
  - Most of them cost less than S\$400
  - Each CPU with two or four cores, running at the speed around 3 GHz (3 Giga clock cycles/second)
    - Normally, a CPU core performs one 64-bit (or 32-bit) integer (or floating point) operation per clock cycle



# Computing power today

- The most powerful supercomputers in the past several years

<http://en.wikipedia.org/wiki/Supercomputer>

– FLOPS : FLoating Point Operations Per Second

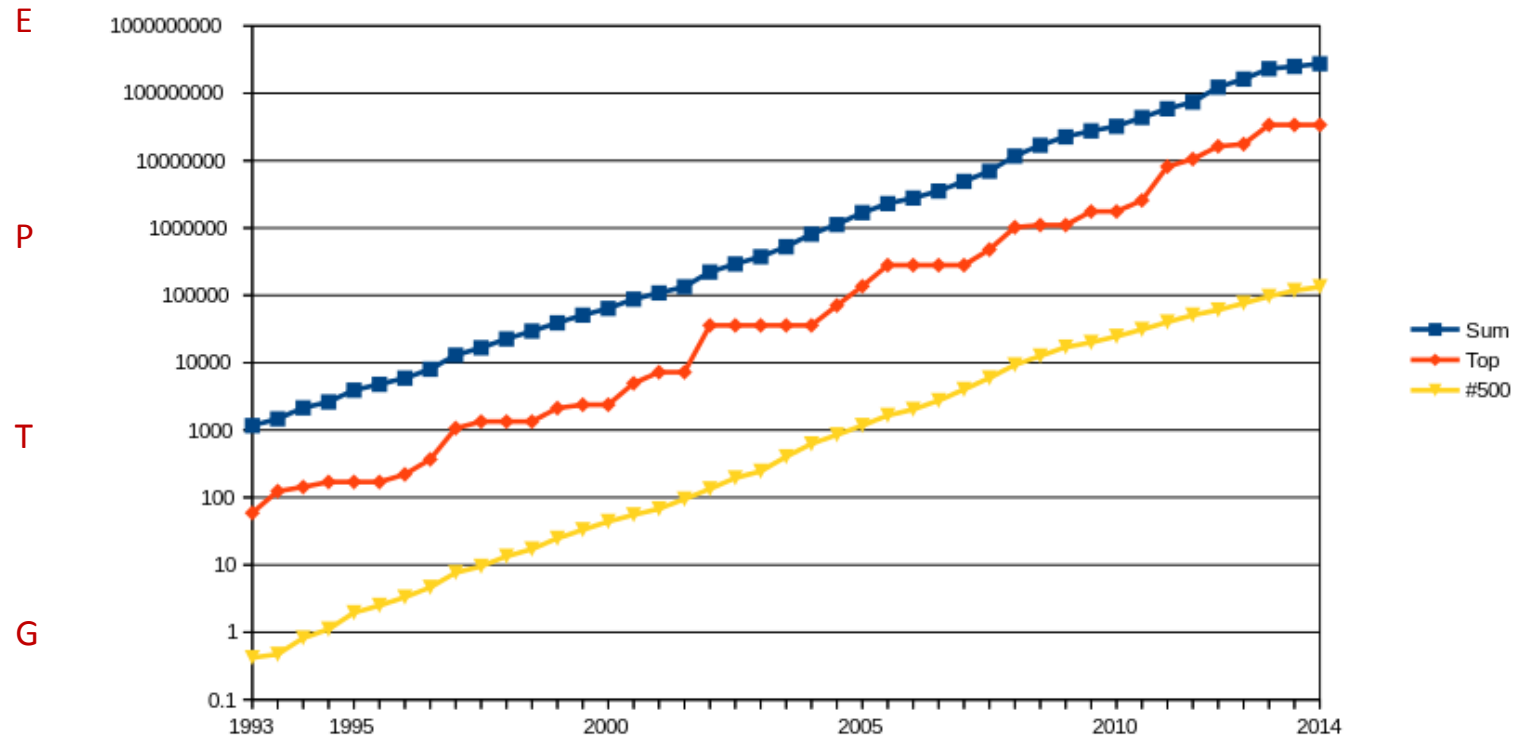
– PFLOPS: Peta ( $2^{50}$ , or  $10^{15}$ ) FLOPS

2018:	IBM Summit,	122.30	PFLOPS,	USA
2016/17:	TaihuLight,	93.01	PFLOPS,	China
2013/14/15:	Tianhe-2,	33.86	PFLOPS,	China
2012:	Cray Titan,	17.59	PFLOPS,	USA
2011:	K Computer,	10.51	PFLOPS,	Japan
2010:	Tianhe-1A,	2.57	PFLOPS,	China
2009:	Cray Jaguar,	1.76	PFLOPS,	USA

# Computing power today

<http://en.wikipedia.org/wiki/TOP500>

- The combined computing power of top 500 supercomputers



- Computer is getting faster and faster (for the same price)
  - 128-bit key is needed today
  - 192-bit or 256-bit key is needed if you want to keep your data secure for more than 30 years

# **Ciphertext-only attack & known-plaintext attack**

- **Ciphertext-only attack**
  - The attacker does not know the message
  - But the attacker knows the statistical information of the message (such as the distribution of letters, digrams, trigrams ...)
- **Known-plaintext attack**
  - The attacker knows part of the plaintext, then try to decrypt the ciphertext to get the rest of the plaintext
  - Known-plaintext attack is practical in many applications
    - Example: NTU webmail (protected using TLS/SSL)  
A lot of contents during the data transmission are known to everyone (such as the email protocol information)

# Kerckhoffs' principle

- **Kerckhoffs' principle**
  - We should assume that the attacker **knows the specifications of the cipher except the secret key**
  - A strong cipher should remain secure in the above scenario
- **Kerckhoff's principle does not mean that we need to make every cipher public**
  - Making a cipher public makes sense only if we assume that all the attackers are cooperative (i.e., they would tell you about their attacks), but such assumption is not true when national security gets involved

# Practical Symmetric Key Ciphers

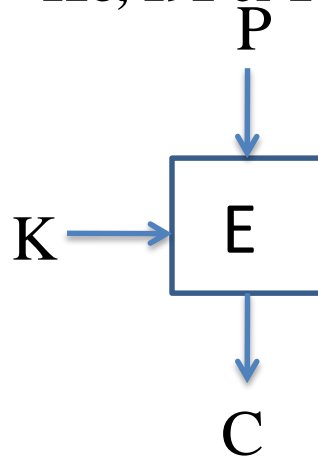
- **Practical ciphers are required to be at least**
  - **Computationally secure**
  - **Remain strong even when the attacker knows the cipher specifications (Kerchoffs' principle)**
  - **Remain strong against known-plaintext attack**
  - **Convenient to use**
    - **A relatively short key (for example: a 128-bit key) can be used to encrypt many messages without compromising security**
- **Two types of practical symmetric key ciphers**
  - **Block cipher**
  - **Stream cipher**

# Block Cipher: overall

- An  $n$ -bit plaintext block is encrypted to an  $n$ -bit ciphertext block
  - Block size:  $n$  bits
    - DES: 64 bits
    - AES: 128 bits
  - Key size
    - DES: 56 bits, (insecure today)
    - 3-DES: 112 or 168 bits
    - AES: 128, 192 or 256 bits

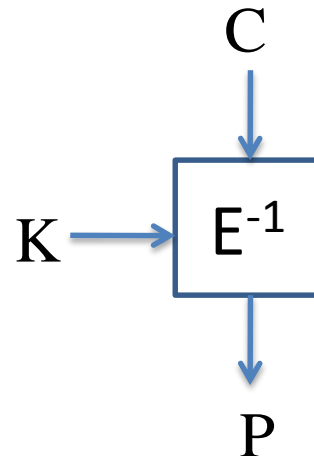
Encryption :

$$C = E_K(P)$$



Decryption :

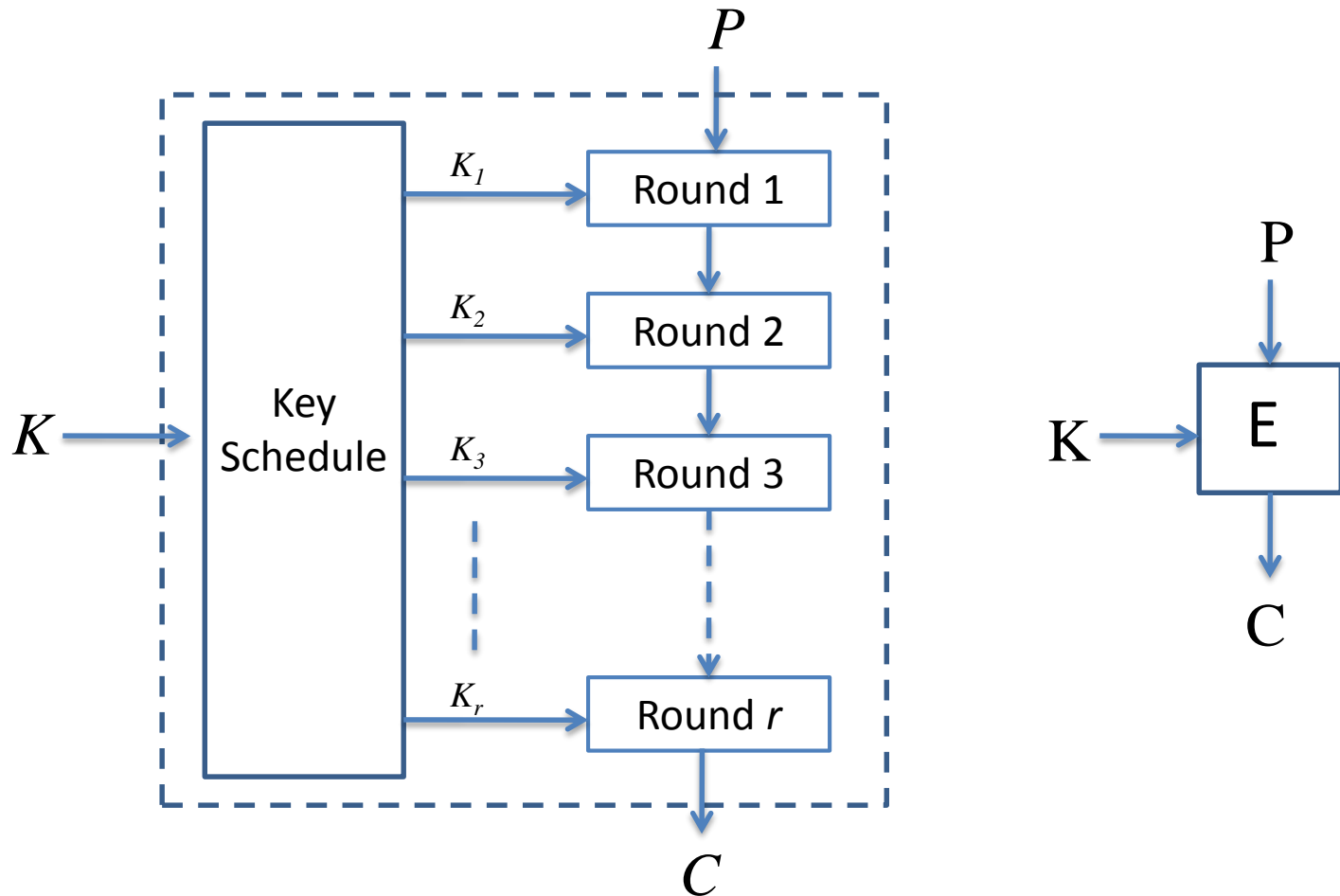
$$P = E_K^{-1}(C)$$



# Block Cipher: Iterated Structure

- To simplify the design, security evaluation and implementation of block cipher, a **round function** is used repeatedly in a block cipher
  - A typical block cipher consists of
    - $r$  rounds (a *round function* for each round)
      - Normally a *round key* is used for each round
    - *key schedule*
      - round keys are generated from the secret key

# Block Cipher: Iterated Structure (cont.)





# Block Cipher: Round Function

- **How to design the round function?**
  - **Design strategy**
    - **Confusion** (non-linear)
      - Combine bits in a nonlinear way
        - » Small substitution table
        - » Multiplication together with XOR
        - » addition together with XOR
        - » ....
    - **Diffusion**
      - Let all the bits affect each other
        - » Permutation
        - » Shift, Rotation
        - » ....

# **Block Cipher: Round Function (cont.)**


- **Two main approaches to design round function**
  - **Substitution-permutation network (SPN)**
    - AES ...
  - **Feistel network**
    - DES ...
- **We will learn the details soon ...**

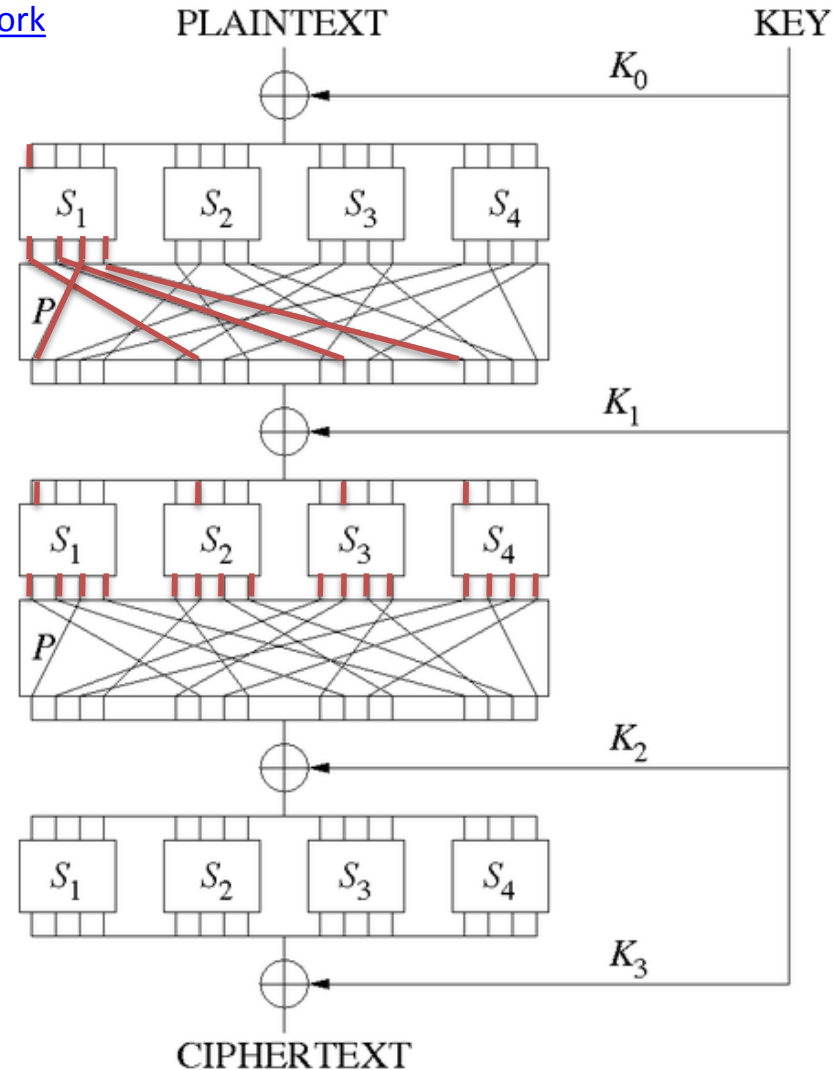
# Block Cipher: Key Schedule

- **How to design key schedule**
  - **Many different approaches**
  - **No perfect guideline so far**
  - **Typically, each bit of the key affects many round key bits at different locations**

# Example: A Simple Block Cipher

[http://en.wikipedia.org/wiki/Substitution-permutation\\_network](http://en.wikipedia.org/wiki/Substitution-permutation_network)

- **Substitution-Permutation network**
  - Confusion: substitution
    - Substitution table: secret or public
  - Diffusion: permutation
    - Permutation: normally public
- Example: A toy cipher 
  - Block size: 16 bits
  - 3 rounds
  - Substitution
    - four Sboxes
    - Each Sbox: 4×4-bit
  - Permutation
    - Bit-wise permutation
    - 16 positions being permuted
    - Carefully chosen to ensure quick diffusion
      - Each plaintext bit affects all the 16 bits in the state after 2 rounds.



**How to design the substitution table?**

# Summary

- **Information-theoretical security & computational security**
- **Practical symmetric key ciphers**
  - Computational security
  - Kerckhoffs' principle
  - Known-plaintext attack & ...
- **Block Cipher**
  - Iterated structure
    - Round function & round key
    - Key schedule
  - Round function
    - Design strategy: Confusion & diffusion
    - Methods: Substitution-permutation network, Feistel network