

# 第七讲 密码算法 Lecture 7 Cryptographic Algorithms

明玉瑞 Yurui Ming  
yrming@gmail.com

# 声明

## Disclaimer

- ▶ 本讲义在准备过程中由于时间所限，所用材料来源并未规范标示引用来源。所引材料仅用于教学所用，作者无意侵犯原著者之知识产权，所引材料之知识产权均归原著者所有；若原著者介意之，请联系作者更正及删除。

The time limit during the preparation of these slides incurs the situation that not all the sources of the used materials (texts or images) are properly referenced or clearly manifested. However, all materials in these slides are solely for teaching and the author is with no intention to infringe the copyright bestowed on the original authors or manufacturers. All credits go to corresponding IP holders. Please address the author for any concern for remedy including deletion.

# 密码系统

## Cryptosystem

- ▶ 一个密码系统是满足以下条件的五元组  $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$  :
  - ▶  $\mathcal{P}$ 表示所有可能的明文组成的有限集;
  - ▶  $\mathcal{C}$ 表示所有可能的密文组成的有限集;
  - ▶  $\mathcal{K}$ 代表密钥空间, 由所有可能密钥组成的有限集;
  - ▶ 对每一个  $K \in \mathcal{K}$ , 都存在一个加密规则  $e_K \in \mathcal{E}$ 和相对应的解密规则  $d_K \in \mathcal{D}$ , 对每对  $e_K: \mathcal{P} \rightarrow \mathcal{C}$ ,  $d_K: \mathcal{C} \rightarrow \mathcal{P}$ , 满足条件: 对每一个明文  $x \in \mathcal{P}$ , 均有  $d_K(e_K(x)) = x$ 。

# 代换密码

## Substitution Cipher

### 代换密码系统

- 令  $\mathcal{P} = \mathcal{C} = \mathbb{Z}_{26}$ ,  $\mathcal{K}$  是由26个数字0到25的所有可能的置换组成的集合, 对任意的置换  $\pi \in \mathcal{K}$ , 定义  $e_{\pi}(x) = \pi(x)$ ,  $d_{\pi}(y) = \pi^{-1}(y)$ , 这里  $\pi^{-1}$  表示  $\pi$  的逆置换。
- 在代换密码的情形下, 可以认为  $\mathcal{P}$  和  $\mathcal{C}$  是26个英文字母, 将加密与解密过程看作一个字母表上的置换。
- 下面展示了一个代换密码的例子, 其中小写字母表示明文, 大写字母表示密文:

<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	<i>g</i>	<i>h</i>	<i>i</i>	<i>j</i>	<i>k</i>	<i>l</i>	<i>m</i>
X	N	Y	A	H	P	O	G	Z	Q	W	B	T

  

<i>n</i>	<i>o</i>	<i>p</i>	<i>q</i>	<i>r</i>	<i>s</i>	<i>t</i>	<i>u</i>	<i>v</i>	<i>w</i>	<i>x</i>	<i>y</i>	<i>z</i>
S	F	L	R	C	V	M	U	E	K	J	D	I

加密

<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>	<i>I</i>	<i>J</i>	<i>K</i>	<i>L</i>	<i>M</i>
d	l	r	y	v	o	h	e	z	x	w	p	t

  

<i>N</i>	<i>O</i>	<i>P</i>	<i>Q</i>	<i>R</i>	<i>S</i>	<i>T</i>	<i>U</i>	<i>V</i>	<i>W</i>	<i>X</i>	<i>Y</i>	<i>Z</i>
b	g	f	j	q	n	m	u	s	k	a	c	i

解密

# 置换密码

## Permutation Cipher

### ► 置换密码系统

- 令 $m$ 为一正整数，令 $\mathcal{P} = \mathcal{C} = (\mathbb{Z}_{26})^m$ ， $\mathcal{K}$ 是由所有定义在集合 $\{1, 2, \dots, m\}$ 上的所有可能的置换组成的集合，对任意的置换 $\pi \in \mathcal{K}$ ，定义：

$$e_{\pi}(x_1, x_2, \dots, x_m) = \pi(x_{\pi(1)}, x_{\pi(2)}, \dots, x_{\pi(m)})$$

$$d_{\pi}(y_1, y_2, \dots, y_m) = \pi^{-1}(y_{\pi^{-1}(1)}, y_{\pi^{-1}(2)}, \dots, y_{\pi^{-1}(m)})$$

这里 $\pi^{-1}$ 表示 $\pi$ 的逆置换。

- 下面展示了一个置换密码的例子。

$x$	1	2	3	4	5	6
$\pi(x)$	3	5	1	6	4	2

加密

$x$	1	2	3	4	5	6
$\pi^{-1}(x)$	3	6	1	5	2	4

解密

# 置换密码

## Permutation Cipher

► 假设需要加密的明文是shesellsseashellsbytheseashore

1. 则首先将明文以6个字母为单位进行分组：

shesel | lsseas | hellsb | ythese | ashore

2. 然后根据选定的置换 $\pi$ ，对每组内的字母进行置换，得到如下结果：

EESLSH | SALSES | LSHBLE | HSYEET | HRAEOS

3. 最后，得到密文： EESLSHSALSESLSHBLEHSYEETHRAEOS

# 分组密码

## Block Cipher

- ▶ 分组密码或块密码是每次只能处理特定长度的一块数据的一类密码算法，这里的一个数据块（block）就称为一个分组。
- ▶ 一个分组的比特数就称为分组长度（block length）。例如，DES和3DES的分组长度为64比特，AES的分组长度为128比特。
- ▶ 分组密码常用的一种设计是迭代密码的设计，每次迭代都包含一系列置换和替换操作，称为轮函数，明文加密将经过 N 轮类似的过程。因此，基于迭代机制的分组密码需要指定一个轮函数和一个密钥表。
- ▶ 分组密码处理完一个分组就结束了，因此不需要通过内部状态来记录加密的进度，分组密码算法只能加密固定长度的分组，但是我们需要加密的明文长度可能会超过分组密码的分组长度，这时就需要对分组密码算法进行迭代，以便将一段很长的明文全部加密。而迭代的方法就称为分组密码的模式（mode）。

# 分组密码

## Block Cipher

- ▶ 下面从整体上讲述一下分组密码的加密过程。
- ▶ 设 $K$ 是某个指定长度的随机二进制密钥，一般通过固定的公开算法，由 $K$ 构造 $N$ 个轮密钥（也称为子密钥），记为 $(K^1, \dots, K^N)$ 。该密钥列表称为密钥编排方案。
- ▶ 假定已经选定轮函数 $g$ ，其第 $r$ 轮迭代以轮密钥 $K^r$ 和上次迭代输出（亦称为当前状态） $w^{r-1}$ 两个变量作为输入，其输出 $w^r$ 定义为下一个状态。初态 $w^0$ 被定义为明文 $x$ ，密文 $y$ 定义为经过 $N$ 后的状态，如下所示：

$$\begin{aligned}w^0 &\leftarrow x \\w^1 &\leftarrow g(w^0, K^1) \\&\dots \\w^{N-1} &\leftarrow g(w^{N-2}, K^{N-1}) \\w^N &\leftarrow g(w^{N-1}, K^N) \\y &\leftarrow w^N\end{aligned}$$



