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版权，所引材料之知识版权均归原著者所有；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；
  - $\mathcal{K}$代表秘钥空间，由所有可能秘钥组成的有限集；
  - 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$d_K(e_K) = x$。

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π^{-1}表示π的逆置换。
- 在代换密码的情形下，可以认为$\mathcal{P}$和$\mathcal{C}$是26个英文字幕，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：

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

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

加密

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

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
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$\{1,2,\cdots,m\}$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：

$$e_\pi(x_1,x_2,\cdots,x_m)=\pi\left(x_{\pi(1)},x_{\pi(2)},\cdots,x_{\pi(m)}\right)$$
$$d_\pi(y_1,y_2,\cdots,y_m)=\pi^{-1}\left(y_{\pi^{-1}(1)},y_{\pi^{-1}(2)},\cdots,y_{\pi^{-1}(m)}\right)$$

  这里π^{-1}表示π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π，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 lenght）。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$(K^1,\cdots,K^N)$。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：

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

分组密码
Block Cipher

- 为了能够解密，轮函数g必须在其第二个自变量固定，即同一个密钥的条件下，是单射函数，这等价于存在函数g^{-1}，对所有的w和K，有$g^{-1}(g(w,K),K)=w$，其解密过程如下：

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

分组密码
Block Cipher

- 上面我们从定义的角度，引入了迭代密码设计并从形式上展示了加密与解密过程。为了具化迭代密码设计，我们引入代换-置换网络的概念。
- 设l和m都是正整数，代换-置换网络一般是从如下两个操作π_S与π_P构建：

$$\pi_S\colon \{0,1\}^l \to \{0,1\}^l$$
$$\pi_P\colon \{1,\cdots,lm\} \to \{1,\cdots,lm\}$$

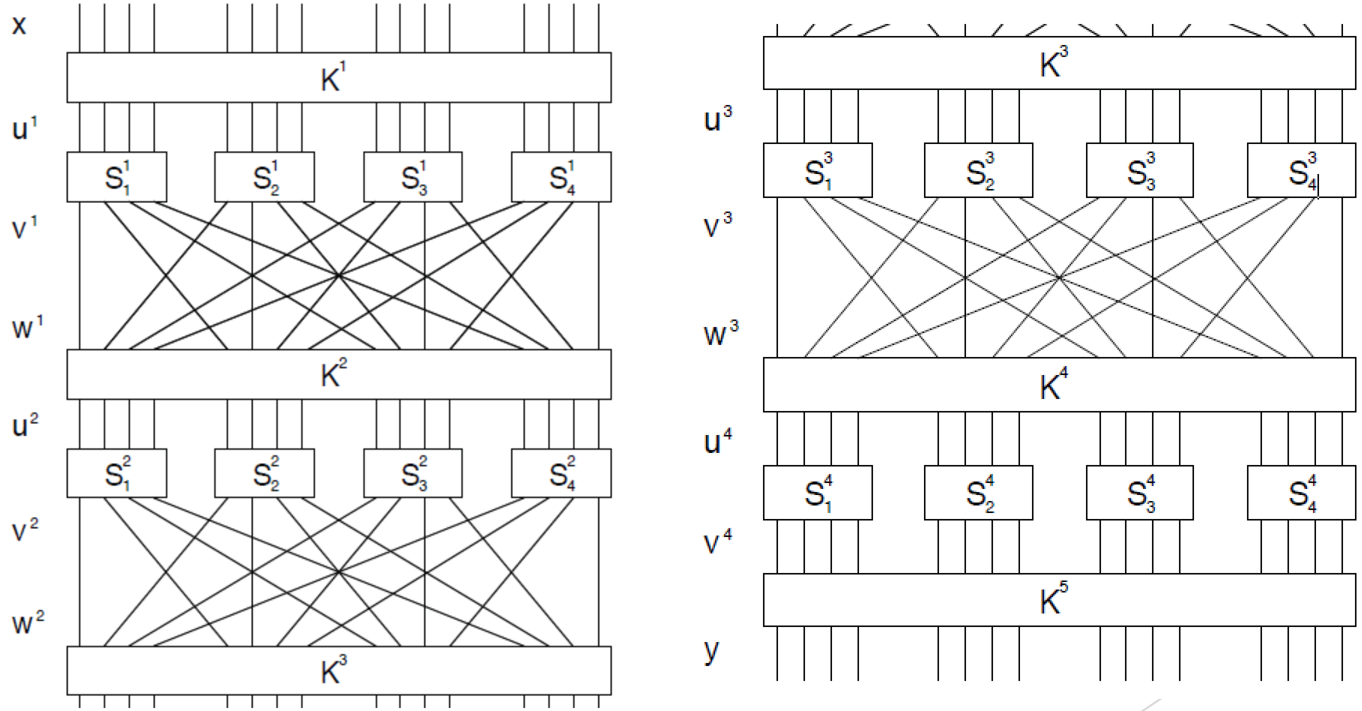
  其中π_S与π_P均为置换操作。 π_S称为S-盒（S为substitute的首字母缩写），是长度为l的2^l字符串上的置换操作，通常是用一个长度为l位的串，替换另一个长度为l位的串。对π_P而言，是通过改变长度为lm的串的每个位的位置的置换操作。
- 对于一个长度为lm的串，假设为$x=(x_1,x_2,\cdots,x_{lm-1},x_{lm})$，为了适配算法，$x$亦可看成$m$个长度为$l$的串的连接，记为$x=x_{\langle 1\rangle}\|x_{\langle 2\rangle}\|\cdots\|x_{\langle m-1\rangle}\|x_{\langle m\rangle}$，其中$x_{\langle i\rangle}$代表第$i$个长度为$l$的串。

分组密码
Block Cipher

- 对第i个长度为l的串，假设$1 \le i \le m$，则有
$$x_{\langle i \rangle} = \left(x_{(i-1)l+1}, x_{(i-1)l+2}, \cdots, x_{il-1}, x_{il}\right)$$
- 下面引入代换-置换网络的概念，令l、 m、 N为正整数，令$\pi_S: \{0,1\}^l \to \{0,1\}^l$，$\pi_P: \{1,\cdots,lm\} \to \{1,\cdots,lm\}$为置换操作。令$\mathcal{P} = \mathcal{C} = \{0,1\}^{lm}$， $\mathcal{K} \subseteq \left(\{0,1\}^{lm}\right)^{N+1}$包含所有能够从初始密码$K$推导的密码编排方案。对选定的编排方案$(K^1,\cdots,K^N)$，我们用上面讲解的加密过程加密明文$x$。
- 整体而言，对于由代换-置换网络构成的迭代密码系统，在N操作的每轮操作中（最后一轮除外），我们将先用π_S进行m次代换操作，然后再用π_P进行一次置换操作。注意，在代换操作之前，我们会用异或操作混入本轮密钥。

分组密码
Block Cipher

- 具体而言，对于第r轮操作，令u^r为S-盒的输入，u^r为S-盒的输出，同时作为π_P的输入，其输出为w^r。 u^{r+1}为将w^r混入第$r+1$轮密钥所得。在最后一步，只有S-盒的代换操作而没有置换操作，然后有额外的一步混入密钥操作（称为白化），这一步主要是为了防止攻击者在不知道密钥的情况下便尝试去解密，流程如下图所示：



分组密码
Block Cipher

- 例：设$l = m = N = 4$，π_S如下定义：

z	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$\pi_S(z)$	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7

注意这里输入z与输出$\pi_S(z)$均用十六进制表示，即

$$
\begin{gathered}
0000_2 = 0_{16} \rightarrow E_{16} = 1110_2 \\
0001_2 = 1_{16} \rightarrow 4_{16} = 0100_2 \\
\cdots \\
1110_2 = E_{16} \rightarrow 0_{16} = 0000_2 \\
1111_2 = F_{16} \rightarrow 7_{16} = 0111_2
\end{gathered}
$$

分组密码
Block Cipher

- π_P定义如下：

z	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\pi_P(z)$	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16

- 关于密码编排方案，假设是从一个32 bits的密钥流$K = (k_1, \cdots, k_{32})$产生。对于第$r$轮，定义$K^r$是由$K$中从$k_{4r-3}$开始的16个连续的比特组成，一个具体的例子如下：

$$K = 0011 \quad 1010 \quad 1001 \quad 0100 \quad 1101 \quad 0110 \quad 0011 \quad 1111$$

生成的轮密钥如下所示：

$$K^1 = 0011 \quad 1010 \quad 1001 \quad 0100 \quad , \quad K^2 = 1010 \quad 1001 \quad 0100 \quad 1101$$
$$K^3 = 1001 \quad 0100 \quad 1101 \quad 0110 \quad , \quad K^4 = 0100 \quad 1101 \quad 0110 \quad 0011$$
$$K^5 = 1101 \quad 0110 \quad 0011 \quad 1111$$

注意这样的密码编排方案只是为了展示，实际中一般不会采用这种不太安全的方式。

分组密码
Block Cipher

- 有了上面的铺垫，我们可以按照代换-置换网络的操作，进行加密与解密操作。