高级数据加密标准AES

密码理论与技术研究中心
于红波

2015年3月30日

提纲

- 高级数据加密标准AES
- AES数学基础
- AES算法介绍

AES History

- AES
  - 1997年，NIST公开征集数据加密算法以取代DES
  - 1998年，共收到15个算法
  - 1999年，从15个中选中5个算法：MARS、RC6、Rijndael、Serpent和Twofish
  - 2000年10月：Rijndael获胜，Vincent Rijmen 和Joan Daemen
  - 2001年11月：NIST公布新的数据加密标准AES

高级数据加密标准AES

- AES
  - 分组密码
  - 分组长度128比特
  - Substitution-Permutation Network (SPN)
  - 三种不同长度的密钥和轮数
    - AES-128：128比特密钥 + 10轮
    - AES-192： 192比特密钥 + 12轮
    - AES-256： 256比特密钥 + 14轮

AES 应用

- AES
  - 免费使用
  - 简单漂亮的设计
  - 安全性高
  - 实现效率高
- 美国国家标准
  - AES-128用于SECRET信息
  - AES-192和AES-256用于TOP SECRET 信息
- 商业应用

AES算法：State

- 状态(State): 与消息分组相同，16个字节
- 表示成4乘4的矩阵

input bytes

in_0	in_4	in_8	in_{12}
in_1	in_5	in_9	in_{13}
in_2	in_6	in_{10}	in_{14}
in_3	in_7	in_{11}	in_{15}

$\rightarrow$

State array

$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$

$\rightarrow$

output bytes

out_0	out_4	out_8	out_{12}
out_1	out_5	out_9	out_{13}
out_2	out_6	out_{10}	out_{14}
out_3	out_7	out_{11}	out_{15}

AES算法：总体

State=Plaintext

AddRoundKey(State, Key_0)

For i=0 to r-1

SubBytes(State)

ShiftRows(State)

MixColumns(State)

AddRoundKey(State, $RoundKey_i$)

End for

r-1轮

$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$

SubBytes(State)

ShiftRows(State)

~~MixColumns(State)~~

AddRoundKey(State, $RoundKey_r$)

Ciphertext=State

最后轮

需要r+1个轮密钥

AES：SubBytes

State=Plaintext

AddRoundKey(State, Key_0)

For i=0 to r-1

SubBytes(State)

ShiftRows(State)

MixColumns(State)

AddRoundKey(State, $RoundKey_i$)

End for

SubBytes(State)

ShiftRows(State)

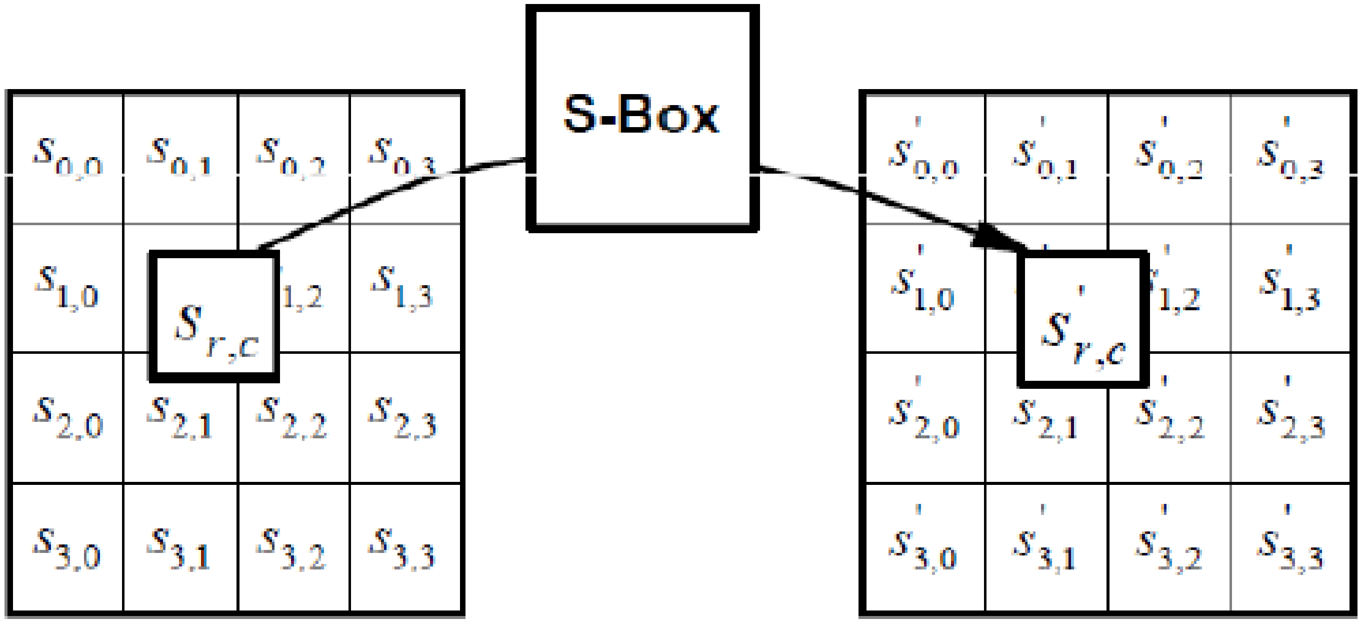
MixColumns(State)

~~AddRoundKey(State,~~ $RoundKey_r$)

Ciphertext=State

AES： SubBytes

- 字节代替变换(SubBytes( )): 对每个字节进行S盒查表代换



AES：SubBytes(续)

- S-box：8比特输入、8比特输出、可逆
- 由以下两个步骤计算 $b' = S(a)$
  - 在GF(2^8)中求 $b = a^{-1}$ (使用扩展Euclid算法）
  - 对b应用以下仿射变换

$$
\begin{bmatrix} b_0' \\ b_1' \\ b_2' \\ b_3' \\ b_4' \\ b_5' \\ b_6' \\ b_7' \end{bmatrix}
=
\begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\
1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 & 1
\end{bmatrix}
\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix}
+
\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}
$$

AES：ShiftRows

State=Plaintext

AddRoundKey(State, $RoundKey_0$)

For i=0 to r-1

SubBytes(State)

ShiftRows(State)

MixColumns(State)

AddRoundKey(State, $RoundKey_i$)

End for

SubBytes(State)

ShiftRows(State)

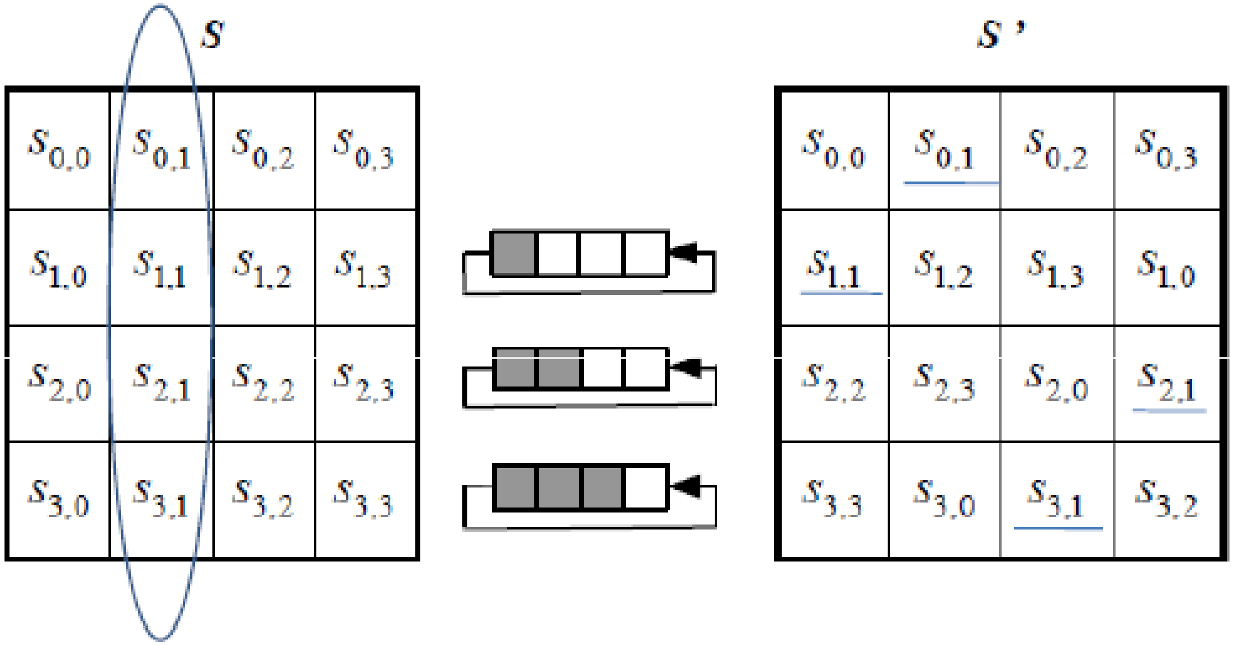
MixColumns(State)

AddRoundKey(State, $RoundKey_r$)

Ciphertext=State

AES： ShiftRows

- 行移位变换(ShiftRows)



第1行:不移位
第2行:左移1位
第3行:左移2位
第4行:左移3位

经过行移位后，1列中的4个字节被分布到不同的列中

AES：MixColumns

State=Plaintext

AddRoundKey(State, Key_0)

For i=0 to r-1

Subbytes(State)

ShiftRows(State)

~~MixColumns(State)~~

AddRoundKey(State, $RoundKey_i$)

End for

Subbytes(State)

ShiftRows(State)

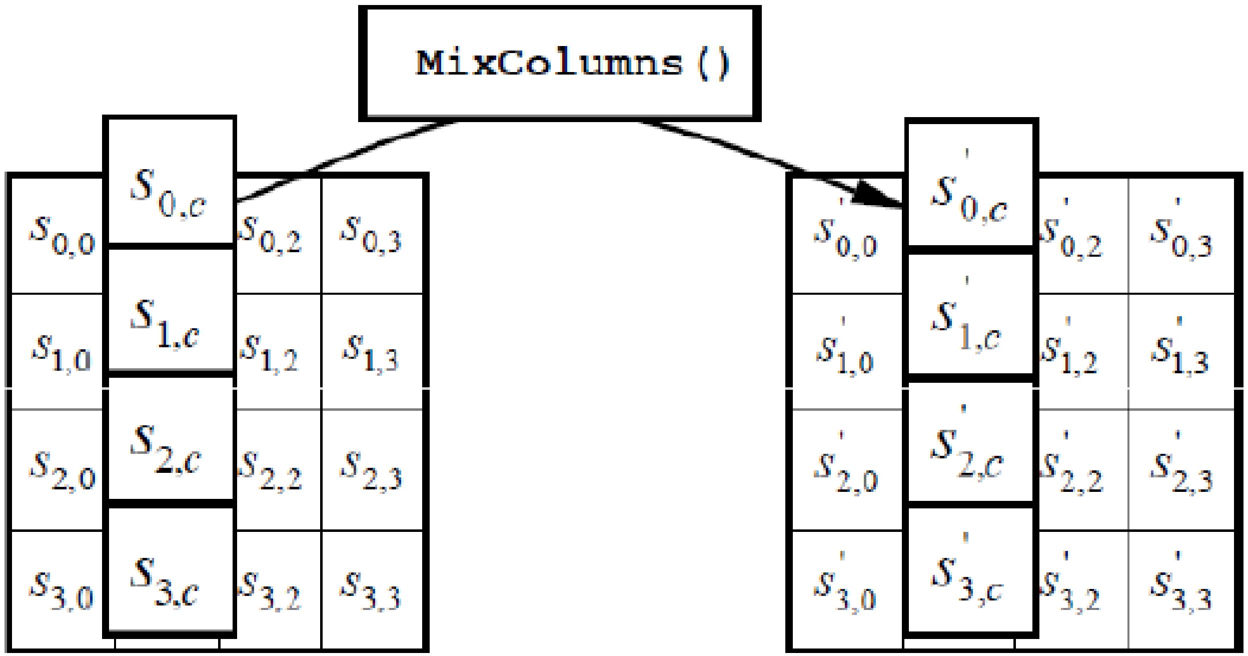
~~MixColumns(State)~~

AddRoundKey(State, $RoundKey_r$)

Ciphertext=State

AES：MixColumns

- 列混合变换(MixColumns( )): 对一个状态逐列进行变换



$$a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$$

$$s'(x) = a(x) \otimes s(x)$$

AES：MixColumns

- 列混合

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

$$\begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} a_0 & a_3 & a_2 & a_1 \\ a_1 & a_0 & a_3 & a_2 \\ a_2 & a_1 & a_0 & a_3 \\ a_3 & a_2 & a_1 & a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

$$a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$$

AES：MixColumns

- 列混和的一些属性
  - 一个输入的字节影响所有的4个输出字节
  - 假设有t_1个非零输入字节，输出有t_2个非零字节，则$t_1+t_2 \geq 5$
  - 最大距离可分码(MDS): 对任意的x, 则在$GF(2^8)$中， (x, MixColumns(x))至少是5。

AES： AddRoundKey

State=Plaintext

AddRoundKey(State, Key_0)

For i=0 to r-1

Subbytes(State)

ShiftRows(State)

MixColumns(State)

~~AddRoundKey(State, $RoundKey_i$)~~

End for

Subbytes(State)

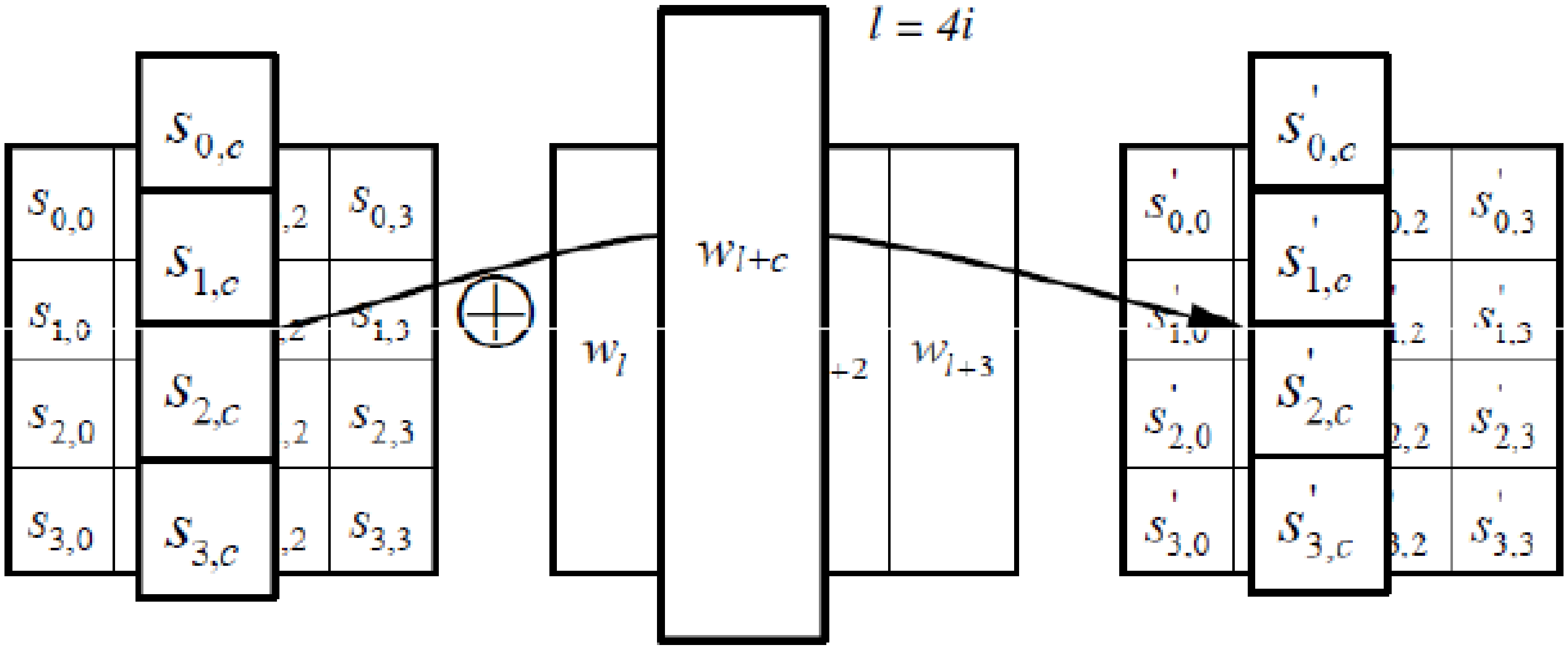
ShiftRows(State)

MixColumns(State)

~~AddRoundKey(State, $RoundKey_r$)~~

Ciphertext=State

AES： AddRoundKey



AES:密钥生成方案

- 每个轮密钥128比特
- 轮密钥用32比特字的数组来表示 w[i]
  - 第一轮： w[0],w[1],w[2],w[3]
  - 第二轮：w[4],w[5],w[6],w[7]
  - ……

AES:密钥生成方案

- Example: AES-128

W[0] = (K[0],K[1],K[2],K[3])
W[1] = (K[4],K[5],K[6],K[7])
W[2] = (K[8],K[9],K[10],K[11])
W[3] = (K[12],K[13],K[14],K[15])

$$
\left.\begin{aligned}
&RCon[1] \leftarrow 01000000\\
&RCon[2] \leftarrow 02000000\\
&RCon[3] \leftarrow 04000000\\
&RCon[4] \leftarrow 08000000\\
&RCon[5] \leftarrow 10000000\\
&RCon[6] \leftarrow 20000000\\
&RCon[7] \leftarrow 40000000\\
&RCon[8] \leftarrow 80000000\\
&RCon[9] \leftarrow 1B000000\\
&RCon[10] \leftarrow 36000000
\end{aligned}\right\}\ \text{Round constants}
$$

for $i \leftarrow 0$ **to** 3

do $w[i] \leftarrow (key[4i], key[4i+1], key[4i+2], key[4i+3])$ } Load the key into w[ ]

for $i \leftarrow 4$ **to** 43

$$
\textbf{do} \begin{cases} temp \leftarrow w[i-1] \\ \textbf{if } i \equiv 0 \pmod 4 \\ \quad \textbf{then } temp \leftarrow \text{SUBWORD}(\text{ROTWORD}(temp)) \oplus RCon[i/4] \\ w[i] \leftarrow w[i-4] \oplus temp \end{cases}
$$

S盒变换

循环左移1个字节

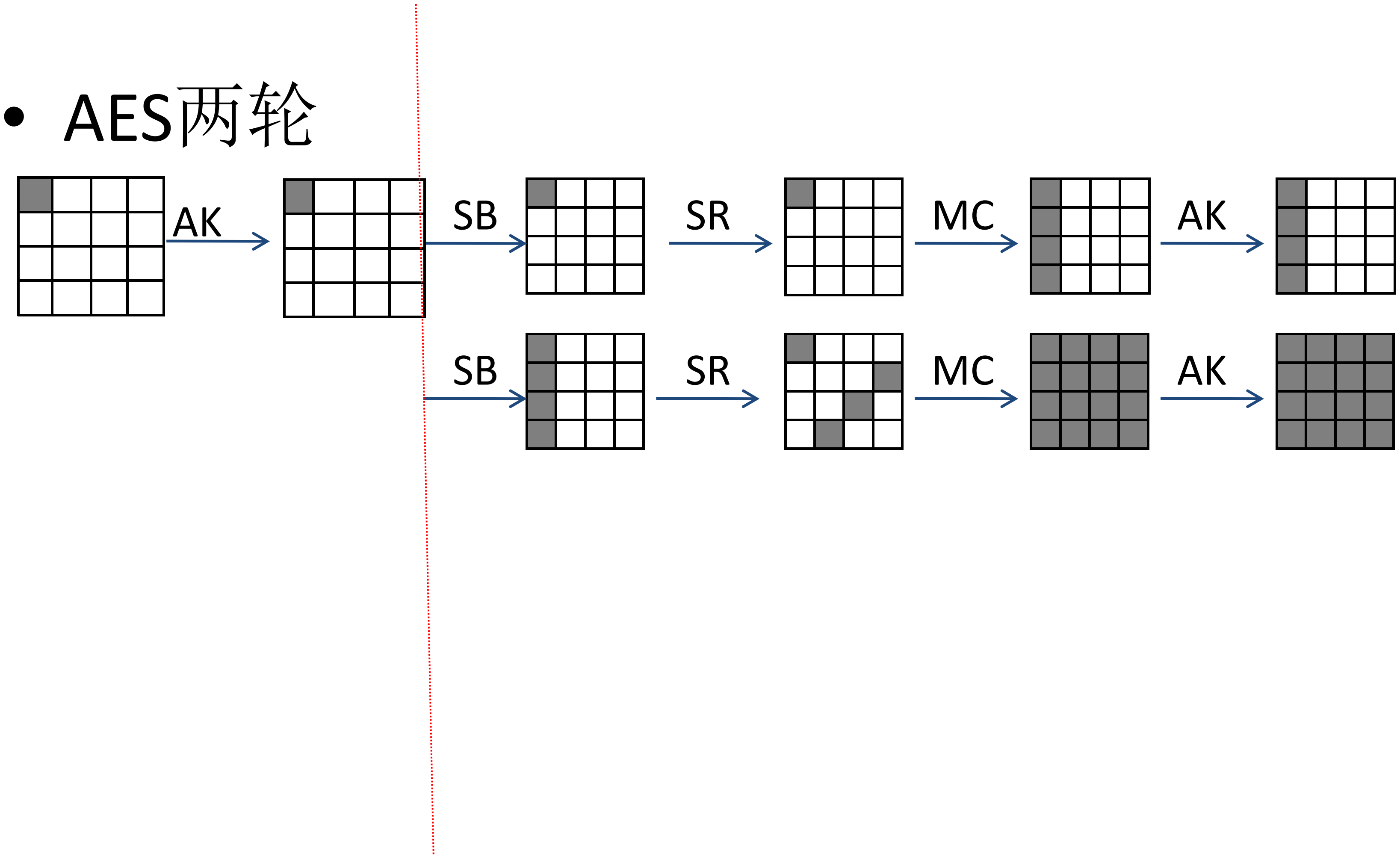
return (w[0],w[1],…,w[43])

AES:密钥生成方案

```
KeyExpansion(byte key[4*Nk], word w[Nb*(Nr+1)], Nk)
begin
    word  temp

    i = 0

    while (i < Nk)
        w[i] = word(key[4*i], key[4*i+1], key[4*i+2], key[4*i+3])
        i = i+1
    end while

    i = Nk

    while (i < Nb * (Nr+1)]
        temp = w[i-1]
        if (i mod Nk = 0)
            temp = SubWord(RotWord(temp)) xor Rcon[i/Nk]
        else if (Nk > 6 and i mod Nk = 4)
            temp = SubWord(temp)
        end if
        w[i] = w[i-Nk] xor temp
        i = i + 1
    end while
end
```

	Key Length (*Nk words*)	Block Size (*Nb words*)	Number of Rounds (*Nr*)
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

- AES两轮



AES加解密

State=Plaintext

AddRoundKey(State, Key_0)

For i=0 to r-1

Subbytes(State)

ShiftRows(State)

MixColumns(State)

AddRoundKey(State, $RoundKey_i$)

End for

Subbytes(State)

ShiftRows(State)

~~MixColumns(State)~~

AddRoundKey(State, $RoundKey_r$)

Ciphertext=State

State=Ciphertext

AddRoundKey(State, $RoundKey_r$)

For i=0 to r-1

InvShiftRows(State)

InvSubBytes(State)

AddRoundKey(State, $RoundKey_{r-i}$)

InvMixColumns(State)

End for

InvShiftRows(State)

InvSubBytes(State)

AddRoundKey(State, $RoundKey_{r-i}$)

~~InvMixColumns(State)~~

Plaintext=State

AES数学基础

Euclid算法(辗转相除法）：设a和b是给定的两个整数，$b\neq 0$, b不能整除a, 重复应用带余除法得到下列k个等式：

$a=q_0 b+r_0$, $0<r_0<|b|$

$b=q_1 r_0+r_1$, $0<r_1<r_0$

$r_0=q_2 r_1+r_2$, $0<r_2<r_1$,

….. ……

$r_{k-5}=q_{k-3} r_{k-4}+r_{k-3}$, $0<r_{k-3}<r_{k-4}$

$r_{k-4}=q_{k-2} r_{k-3}+r_{k-2}$, $0<r_{k-2}<r_{k-3}$

$r_{k-3}=q_{k-1} r_{k-2}$

则$r_{k-2}=GCD(a,b)$。复杂度 $O(\log_2^2 a)$

AES数学基础（续）

1. 欧几里德算法(Euclid)，辗转相除法
   - 用于计算a和b的最大公因子 GCD(a,b)
     - 使用GCD(a,b)=GCD(b, a mod b)

$$
\begin{aligned}
& a = q_0 b + r_0, \quad 0 < r_0 < |b| \\
& b = q_1 r_0 + r_1, \quad 0 < r_1 < r_0 \\
& r_0 = q_2 r_1 + r_2, \quad 0 < r_2 < r_1 \\
& \cdots\cdots \\
& r_{k-3} = q_{k-1} r_{k-2} + r_{k-1}, \quad 0 < r_{k-1} < r_{k-2} \\
& r_{k-2} = q_k r_{k-1} + r_k, \quad 0 < r_k < r_{k-1} \\
& r_{k-1} = q_{k+1} \boxed{r_k .}
\end{aligned}
$$

2. 扩展的欧基里德算法
   - 用于寻找x和y, 满足$ax+by$=GCD(a,b)
     - 基本思想：在欧几里德算法的第i步，寻找 $r_i=ax_i+by_i$.

AES数学基础（续）

- 例：求42823和6409的最大公因子，并将它表示成42823和6409的整系数线形组合形式。

  42823=6 · 6409+4369

  6409=1 · 4369+2040

  4369=2 · 2040+289

  2040=7 · 289+17

  289=7 · 17

 (42823,6409)

=(6409,4369)

=(4369,2040)

=(2040,289)

=(289,17)=17

- 上面过程的逆过程

17 = 2040 -7 · 289

17 = 2040-7 · (4369-2 · 2040)

  =-7 · 4369 + 15 · 2040

17 = -7 · 4369 + 15 · (6409-4369)

  =15 ·6409-22 ·4369

17= 15 ·6409-22 ·(42823-6·6409)

  $= -22 \cdot 42823 + 147 \cdot 6409$

 即(42823, 6409)

  $= -22 \cdot 42823 + 147 \cdot 6409.$

AES数学基础（续）

- 群：设G是一个非空集合，在G中定义了一个二元运算。，若∘满足下面条件，则G称为一个群。

1. 任意 $a, b \in G$，则$a \circ b \in G$
2. 对于任意的 $a, b, c \in G$，则 $(a \circ b) \circ c = a \circ (b \circ c)$

半群

3. 在G中存在一个元素e, 它对G中任何一个元素g, 有

$$e \circ g = g \circ e = g$$

4. 对G中任何一个元素g，都存在一个元素g’，使得

$$g \circ g' = g' \circ g = e$$

e唯一，称为单位元

g’唯一，称为g的逆元

AES数学基础（续）

- 例：

如整数集合Z对 数的加法 构成一个群；

全体不等于0的有理数对普通数的乘法构成一个群

设 $n \in Z$ ，模n剩余类 $Z_n = \{[k] \mid k \in Z\} = \{\overline{0}, \overline{1}, \ldots, \overline{n-1}\}$ 对模n 加法构成一个群

交换群：

一个群，如果对所有的$a \circ b \in G$, 都有$a \circ b = b \circ a$, 则称G是一个交换群（Abel）如加群

AES数学基础（续）

- 环

设R是一个非空集合，在其上定义两种运算加法(+)和乘法($\bullet$), 如果这些运算满足

1. (R, +) 是一个加群， 即(R, +) 对叫加法做成一个交换群
2. $(R,\bullet)$ 对另一个叫乘法的运算做成一个半群
3. 加法对乘法的左右分配律成立：对任意的 $a,b,c \in R,$

$$a \bullet (b+c) = a \bullet b + a \bullet c$$

$$(b+c) \bullet a = b \bullet a + c \bullet a$$

则称$(R,+,\bullet)$ 是一个环。 记为R

AES数学基础（续）

- 环 例子：

1. 整数Z对数的加法和乘法做成一个环， 称为整数环

2. 模n剩余类对模加法和模n乘法成为一个环。

交换环：一个环*R*成为一个交换环，若

$$\forall a,b \in R, a \bullet b = b \bullet a$$

单位元：一个环*R*的一个元素*e*叫做一个单位元，若

$$\forall a \in R, e \bullet a = a \bullet e = a$$

逆元：一个有单位元环的一个元*b*称为*a*的一个逆元，假如

$$a \bullet b = b \bullet a = e$$

AES数学基础（续）

定义：一个至少含有两个元素的环*R*叫做域，假如

1. *R*是交换环。
2. *R*有一个单位元。
3. *R*的每一个不等于0的元有一个逆元。

则*R*为域， 记*R* 为*F*

等价定义： 一个至少含有两个元素的集合F定义了两种运算+和*，如果

1. (F, +) 为一个可换加群。
2. (F*, ●) 为一个可换乘群， F*表示F中所有的非零元，

则F为域。

3. 加法对乘法满足分配律。

如全体有理数的集合、全体实数、全体复数按普通意义下的加、乘构成域： 有理数域、实数域和复数域

AES数学基础（续）

- 有限域(Galois Field)
- 一个只含有有限个元素的域
- 一个阶为m的有限域存在，当且仅当存在一个素数p和一个正整数n, 使得$m=p^n$
- 例如：GF(7)={0,1,2,3,4,5,6}
  - 模7整数加
  - 模7整数乘
- $GF(p^n)$: 系数在p上的次数为n-1的多项式的集合

Évariste Galois
(1811-1832)

AES数学基础（续）

- 有限域GF(2^8)：一个GF(2^8)中的元素的两种表示方式
  - 二进制表示：$b = b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0$
  - 多项式表示

$$b_7 x^7 + b_6 x^6 + b_5 x^5 + b_4 x^4 + b_3 x^3 + b_2 x^2 + b_1 x + b_0$$

- 例如16进制数0x57的二进制表示01010111，对应的多项式为

$$x^6 + x^4 + x^2 + x + 1$$

AES数学基础（续）

- 有限域GF(2^8)中两个元素的加

$$(x^6+x^4+x^2+x+1)\oplus(x^7+x+1)=x^7+x^6+x^4+x^2$$

$$\{01010111\}\oplus\{10000011\}=\{11010100\}$$

$$\{57\}\oplus\{83\}=\{64\}$$

- 有限域GF(2^8)中两个元素的乘法
  - 用●表示
  - 模二元域GF(2)上一个8次不可约多项式的乘积
  - AES选择不可约多项式为

$$m(x)=x^8+x^4+x^2+x+1$$

AES数学基础（续）

- 有限域GF(2⁸)上两个元素的乘法，例

$$\underline{(x^6+x^4+x^2+x+1)}\bullet\underline{(x^7+x+1)}=x^{13}+x^{11}+x^9+x^8+x^7+$$

$$x^7+x^5+x^3+x^2+x+$$

$$x^6+x^4+x^2+x+1$$

$$=x^{13}+x^{11}+x^9+x^8+x^6+x^5+x^4+x^3+1$$

$$x^{13}+x^{11}+x^9+x^8+x^6+x^5+x^4+x^3+1 \bmod (x^8+x^4+x^3+x+1)$$

$$=\underline{x^7+x^6+1}$$

$$\{57\}\bullet\{83\}=\{c1\}$$

AES数学基础（续）

- x与多项式$b(x)$的乘积

$$b_7x^8 + b_6x^7 + b_5x^6 + b_4x^5 + b_3x^4 + b_2x^3 + b_1x^2 + b_0x$$

- 计算 $x \bullet b(x) \bmod m(x)$

$$\mathrm{xtime}(b) = '02' \bullet b = \begin{cases} b \ll 1, & if \ \ b_7 = 0 \\ (b \ll 1) \oplus 1B, & if \ \ b_7 = 1 \end{cases}$$

- 例： $'57' \bullet '13' = 'FE'$ $\quad '13' = '01' \oplus '02' \oplus '10'$

$$'57' \bullet '02' = \mathrm{xtime}('57') = 'AE'$$
$$'57' \bullet '04' = \mathrm{xtime}('AE') = '47'$$
$$'57' \bullet '08' = \mathrm{xtime}('47') = '8E'$$
$$'57' \bullet '10' = \mathrm{xtime}('8E') = '07'$$

$$\begin{aligned} '57' \bullet '13' &= '57' \bullet ('01' \oplus '02' \oplus '10') \\ &= '57' \oplus 'AE' \oplus '07' \\ &= 'FE' \end{aligned}$$

AES数学基础（续）

- 有限域GF(2^8)中乘法逆元的求解：使用扩展欧几里德算法
  - 给定a和b, 寻找x和y，满足

    $$ax + by = \gcd(a, b)$$

  - 若 $\gcd(a, b) = 1$，则 $ax \bmod b = 1$

    即x是a mod b的乘法逆元

AES数学基础（续）

- 系数在GF(2^8)中的多项式
  - 设 $[a_0, a_1, a_2, a_2]$是4个字节(bytes), 对应着一个系数在GF(2^8)，次数小于4的多项式

    $$a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

  - 上面的多项式与GF(2^8)中的多项式是不同的
    - 系数在GF(2^8)
    - 使用一个不同的模多项式： $M(x) = x^4 + 1$
      - x^4+1不是GF(2^8)上的一个不可约多项式
      - 一个固定多项式模M(x)不一定有乘法逆元
      - AES中选择了一个有乘法逆元的固定多项式

AES数学基础（续）

- 系数在GF(2⁸)中的多项式的加法运算

$$a(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

$$b(x) = b_3x^3 + b_2x^2 + b_1x + b_0$$

$$a(x) + b(x) = (a_3 \oplus b_3)x^3 + (a_2 \oplus b_2)x^2 + (a_1 \oplus b_1)x + (a_0 \oplus b_0)$$

AES数学基础（续）

- 系数在GF(2⁸)中的多项式的乘法运算： $\otimes$

$$a(x)=a_3x^3+a_2x^2+a_1x+a_0$$

$$b(x)=b_3x^3+b_2x^2+b_1x+b_0$$

$$c(x)=a(x)\bullet b(x)$$

$$=c_6x^6+c_5x^5+c_4x^4+c_3x^3+c_2x^2+c_1x+c_0$$

$$c_0=a_0\bullet b_0$$

$$c_1=a_1\bullet b_0\oplus a_0\bullet b_1$$

$$c_2=a_2\bullet b_0\oplus a_1\bullet b_1\oplus a_0\bullet b_2$$

$$c_3=a_3\bullet b_0\oplus a_2\bullet b_1\oplus a_1\bullet b_2\oplus a_0b_3$$

$$c_4=a_3\bullet b_1\oplus a_2\bullet b_2\oplus a_1\bullet b_3$$

$$c_5=a_3\bullet b_2\oplus a_2\bullet b_3$$

$$c_6=a_3\bullet b_3$$

AES数学基础（续）

- 系数在GF(2^8)中的多项式的乘法运算（续）

$$d(x) = a(x) \otimes b(x)$$

$$= a(x) \bullet b(x) \bmod (x^4 + 1)$$

由于$x^i \bmod (x^4 + 1) = x^{i \bmod 4}$

故$d(x) = c_3 x^3 + (c_6 \oplus c_2) x^2 + (c_5 \oplus c_1) x + (c_4 \oplus c_0)$

AES数学基础（续）

- 系数在GF(2^8)中的多项式的乘法运算（续）

设 $d(x) = d_3x^3 + d_2x^2 + d_1x + d_0$

则
$$d_0 = a_0 \bullet b_0 \oplus a_3 \bullet b_1 \oplus a_2 \bullet b_2 \oplus a_1 \bullet b_3$$
$$d_1 = a_1 \bullet b_0 \oplus a_0 \bullet b_1 \oplus a_3 \bullet b_2 \oplus a_2 \bullet b_3$$
$$d_2 = a_2 \bullet b_0 \oplus a_1 \bullet b_1 \oplus a_0 \bullet b_2 \oplus a_3 \bullet b_3$$
$$d_3 = a_3 \bullet b_0 \oplus a_2 \bullet b_1 \oplus a_1 \bullet b_2 \oplus a_0 \bullet b_3$$

AES数学基础（续）

- 系数在GF(2^8)中的多项式的乘法运算（续）

对一个固定的多项式 $a(x), d(x) = a(x) \otimes b(x)$
能够表示成下列矩阵形式

$$\begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} a_0 & a_3 & a_2 & a_1 \\ a_1 & a_0 & a_3 & a_2 \\ a_2 & a_1 & a_0 & a_3 \\ a_3 & a_2 & a_1 & a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

AES数学基础（续）

- 系数在GF(2^8)中的多项式的乘法运算 （续）

  由于x^4+1不是GF(2^8)中的不可约多项式，一个固定的多项式模M(x)不一定有乘法逆元，AES选择了一个有逆元的固定多项式

$$a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$$

$$a^{-1}(x) = \{0b\}x^3 + \{0d\}x^2 + \{09\}x + \{0e\}$$

$$\begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

需要关注的分组密码算法

- ISO/IEC分组密码标准
  - TDEA，分组长度64，密钥长度128、192
  - MSITY1、CAST-128， 分组64，密钥128
  - AES、Camellia，分组128，密钥128、192、256
  - SEED，分组128，密钥128
- NESSIE建议的分组密码
  - IDEA、Khazad、MISTY1、SAFER++， TDEA、Camellia、RC6
  - SHACAL-2，分组256，密钥512

需要关注的分组密码算法

- 日本: CRYPTREC项目
- 中国: SM4
- 韩国: ARIA
- 轻量级分组密码算法
  – PRESENT、LED、SIMON等

需要关注的分组密码算法

谢谢！