Rijndael 密码的逆序 Square 攻击 [1]

多 磊* 李 超*,**

*(国防科技大学数学与系统科学系 长沙 410073)
**(中国科学院软件研究所计算机重点实验室 北京 100080)

摘 要: 2000 年 10 月 Rijndael 被选为高级加密标准 (AES)，目前对它最有效攻击仍是由设计者提出的 Square 攻击。Square 攻击是利用密码 Square 特性提出的选择明文攻击，可以对六轮和六轮以下的 Rijndael 密码进行成功的攻击，攻击六轮 Rijndael 的所有密钥的计算量为 $2\times 2^{72}+2^{64}$，五轮密码的复杂度为 $3\times 2^{40}+2^{32}$。该文提出了逆序 Square 攻击算法，该算法是基于密码 Square 特性提出的选择密文攻击方法。它攻击六轮 Rijndael 密码的所有密钥的复杂度为 $2^{72}+2^{56}$，五轮密码的复杂度为 $2^{40}+2^{24}$。若改变密钥扩散准则中的圈循环顺序，五轮密码的逆序 Square 攻击复杂度由 2^{40} 降为 2^{32}，六轮的攻击复杂度由 2^{72} 降为 2^{64}。

关键词: Rijndael 密码，Square 攻击，逆序 Square 攻击
中图分类号: TN918.1 **文献标识码**: A **文章编号**: 1009-5896(2004)01-0065-07

The Inverse Square Attack of Rijndael Cipher

Duo Lei* Li Chao*,**

*(*Dept. of Math. and Sys. Science, Nat. Univ. of Defense Tech., Changsha* 410073, *China*)

**(*Lab. of Computer Sci., Inst. of Software, Chinese Academic of Scs., Beijing* 100080, *China*)

Abstract Rijndael was selected as the AES. The best-known attack against Rijndael is still the one presented by the designers called Square attack. Square attack is a chosen plaintext attack. In this paper a new kind of attack called Inverse Square attack is presented which is a kind of chosen cipher text attack and better than Square attack. It also shows that if only inverse the moving direction of RotByte transformation of key schedule, the complexity of the attack will be 2^8-factor bellow the complexity of Square attack.

Key words Rijndael cipher, Square attack, Inverse square attack

1 引言

2000 年 10 月，美国国家标准研究所 (NIST) 决定选用 Rijndael[1] 密码作为高级加密标准 (AES)。Rijndael 是一种基于 Square 结构的 Substitution Permutation Network(SPN) 结构分组密码，目前对它最有效的攻击仍是设计者 Daemen 等人提出的，后由 Ferguson 等人改进的 Square 攻击方法 [2,3]，它可用于攻击四、五和六轮的 Rijndael 密码。

本文首次提出了逆序 Square 攻击方法，并对四、五和六轮 Rijndael 密码进行了成功的攻击。本算法特点是：

第一，逆序 Square 攻击对五轮和六轮的 Rijndael 密码的攻击略优于目前给出的 Square 攻击。文献 [1] 指出对五轮和六轮 Rijndael 的 Square 攻击强度分别为 2^{40} 和 2^{72}，但一次攻击只

[1] 2002-07-22 收到，2003-01-08 改回
国防科技大学基础研究基金 (Jc02-02-007) 和中国科学院软件研究所计算机重点实验室开放基金 (Syskf0201) 资助课题

能得到 32bit 密钥. 而在逆序 Square 攻击中一次攻击可以得到 48bit 密钥。

第二，它是一种选择密文攻击方法。Square 攻击是利用密码 Square 特性提出的选择明文攻击，而逆序 Square 攻击是利用密码解密时具有 Square 特性提出的选择密文攻击方法，但攻击强度有了明显提高。

第三，逆序 Square 与 Rijndael 密码的密钥扩散准则有关。Rijndael 密码的密钥扩展算法是一种相关性较强的扩展算法。文献 [1] 指出 Square 攻击是一种选择明文攻击，它独立于 specific choices of ByteSub, the multiplication polynomial of MixColumn and the key schedule 。但逆序 Square 攻击中攻击的强度与密码中 RotByte(W) 的顺序选取有密切关系，如果我们将 $\mathrm{RotByte}(a,b,c,d)=(b,c,d,a)$ 改为 $\mathrm{RotByte}(a,b,c,d)=(d,a,b,c)$，五轮和六轮的攻击强度分别变为 2^{32} 和 2^{60}。

第四，文献 [3] 给出了 Rijndael 密码的 Square 攻击的改进算法，改进算法在六轮攻击中对明文的选择提出了新的思路，在逆序 Square 攻击中适当选择密文后仍保持了同样的优势。

本文结构如下：第 2 节中对密码进行较详细描述，第 3 节我们描述了 Rijndael 密码的 Square 攻击和逆序 Square 攻击算法，及其对密码的攻击和改变密钥后的攻击。第 4 节中对两种攻击算法进行了比较，第 5 节是总结。

2 Rijndael 密码的结构

Rijndael 密码中明文分组和密钥长度分别可以为 128 、 192 和 256bit 3 种长度，本文只讨论了 128bit 明文分组长度和密钥长度下的密码。它共有十轮变换，第一轮变换前有一个密钥加，最后一轮中无列混合。下面将用表格的形式对密码一轮变换进行描述。

2.1 加密算法

每一轮包含四种变换 (表 1 为初始状态)：

(1) 字节代替变换 (表 2)ByteSub(BS)，它是密码中唯一非线性变换。

(2) 行移位变换 (表 3)ShiftRow(SR)，它的 1,2,3 和 4 行分别循环左移 0,1,2,3 位。

(3) 列混合变换 (表 4)MixColumn(MC), 它是左乘一列混合矩阵，表 4 为乘积结果。

(4) 加圈密钥 (表 5)AddRoundKey(ARK)

表 1(轮初态)

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

表 2 (S 盒变换)

S[1]	S[2]	S[3]	S[4]
S[5]	S[6]	S[7]	S[8]
S[9]	S[10]	S[11]	S[12]
S[13]	S[14]	S[15]	S[16]

表 3 (行移位)

S[1]	S[2]	S[3]	S[4]
S[6]	S[7]	S[8]	S[5]
S[11]	S[12]	S[9]	S[10]
S[16]	S[13]	S[14]	S[15]

表 4 (列混合)

2S[1]+3S[6]+S[11]+S[16]	2S[2]+3S[7]+S[12]+S[13]	2S[3]+3S[8]+S[9]+S[14]	2S[4]+3S[5]+S[10]+S[15]
S[1]+2S[6]+3S[11]+S[16]	S[2]+2S[7]+3S[12]+S[13]	S[3]+2S[8]+3S[9]+S[14]	S[4]+2S[5]+3S[10]+S[15]
S[1]+S[6]+2S[11]+3S[16]	S[2]+S[7]+2S[12]+3S[13]	S[3]+S[8]+2S[9]+3S[14]	S[4]+S[5]+2S[10]+3S[15]
3S[1]+S[6]+S[11]+2S[16]	3S[2]+S[7]+S[12]+2S[13]	3S[3]+S[8]+S[9]+2S[14]	3S[4]+S[5]+S[10]+2S[15]

表 5　加圈密钥 (注：S[i] 表示对字节 i 做 S 盒变换，“+”为模 2 加，i 表示第 i 个字节)

2S[1]+3S[6]+S[11]+S[16]+k_1	2S[2]+3S[7]+S[12]+S[13]+k_2
S[1]+2S[6]+3S[11]+S[16]+k_5	S[2]+2S[7]+3S[12]+S[13]+k_6
S[1]+S[6]+2S[11]+3S[16]+k_9	S[2]+S[7]+2S[12]+3S[13]+k_{10}
3S[1]+S[6]+S[11]+2S[16]+k_{13}	3S[2]+S[7]+S[12]+2S[13]+k_{14}
2S[3]+3S[8]+S[9]+S[14]+k_3	2S[4]+3S[5]+S[10]+S[15]+k_4
S[3]+2S[8]+3S[9]+S[14]+k_7	S[4]+2S[5]+3S[10]+S[15]+k_8
S[3]+S[8]+2S[9]+3S[14]+k_{11}	S[4]+S[5]+2S[10]+3S[15]+k_{12}
3S[3]+S[8]+S[9]+2S[14]+k_{15}	3S[4]+S[5]+S[10]+2S[15]+k_{16}

2.2 密钥扩展算法

Rijndael 的密钥扩展算法是以 4-字节为单位的扩展算法。设第 i 轮密钥为 $[W_{i,1}, W_{i,2}, W_{i,3}, W_{i,4}]$，由第 $i-1$ 轮密钥得到第 i 轮密钥的递归算法如下：

$$W_{i,j} = \begin{cases} W_{i-1,j} \oplus eW_{i,j-1}, & j = 2,3,4 \\ W_{i-1,j} \oplus \mathrm{RotByte}(W_{i-1,4}) \oplus \mathrm{Rcon}(i), & j = 1 \end{cases}$$

其中

$$[W_{0,1}, W_{0,2}, W_{0,3}, W_{0,4}] = [k_{0,1}, k_{0,2}, k_{0,3}, k_{0,4}]$$
$$\mathrm{RotByte}(b_1, b_2, b_3, b_4) = (b_2, b_3, b_4, b_1)$$
$$\mathrm{Rcon}(i) = [(1 << i) \bmod (0x11b), 0, 0, 0]$$

对密钥扩展方案用表格描述，如表 6, 表 7 所示。

表 6　i 轮轮密钥

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

表 7　$i+1$ 轮轮密钥

1+S[8]+$r[i]$	1+2+S[8]+$r[i]$	1+2+3+S[8]+$r[i]$	1+2+3+4+S[8]+$r[i]$
5+S[12]	5+6+S[12]	5+6+7+S[12]	5+6+7+8+S[12]
9+S[16]	9+10+S[16]	9+10+11+S[16]	9+10+11+12+S[16]
13+S[4]	13+14+S[4]	13+14+15+S[4]	13+14+15+16+S[4]

注：$r[i]$ 表示圈常量 ($r[i] = x^{i-1}$，表示模 $x^8 + x^5 + x^4 + x + 1$ 的结果)

2.3 Rijndael 密码中密钥扩散准则的变种

在逆序 Square 攻击中，密钥扩展方案的 RotByte 的选取对密码的攻击强度影响很大，本文中的变种方案是

$$\mathrm{RotByte}(b_1, b_2, b_3, b_4) = (b_2, b_3, b_4, b_1) \text{ 变换 } \mathrm{RotByte}(b_1, b_2, b_3, b_4) = (b_4, b_1, b_2, b_3)$$

3 Square 攻击和逆序 Square 攻击

Square 攻击是针对 Square 密码提出的选择明文攻击方法，它利用 Square 密码的块操作特性和 SPN 结构密码中每一变换的可逆性提出的攻击方法。六轮和六轮以下的 Rijndael 密码的 Square 攻击均比穷尽密钥攻击快，因为 Rijndael 密码继承了 Square 密码的块操作特性和 SPN 结构。

3.1 Square 攻击

Rijndael 密码的 Square 特性是选择只有一个字节位互不相同的 256 组明文，三轮加密后这种取值的不同扩散到所有密文 16 个字节上且对应字节互不相同。将 256 组密文按字节模 2 加，和序列为全零序列。

四轮密码的 Square 攻击方法是：

for(CheckKey4[m] = 0; k < 256; CheckKey4[m] + +){// CheckKey4[m]：表示猜测的第四轮密钥的第 m 个字节

CheckSum=0;

for(i = 0; i < 256; i + +)

CheckSum=CheckSum∧ SI[Cipher4[m]∧ CheckKey[m]];

//SI：表示 ByteSub 的逆变换， Cipher4[m] 表示四轮加密后密文的第 m 个字节

if (CheckSum=0) printf("%02x", CheckKey4[m]); // 第四轮密钥的第 m 个字节 }

五轮 Rijndael 的加密过程中第四轮一个字节信息扩散到第五轮四个字节当中，图 1 中描述了第四轮第一个字节在四、五轮加密过程中的扩散情况和依赖的密钥。

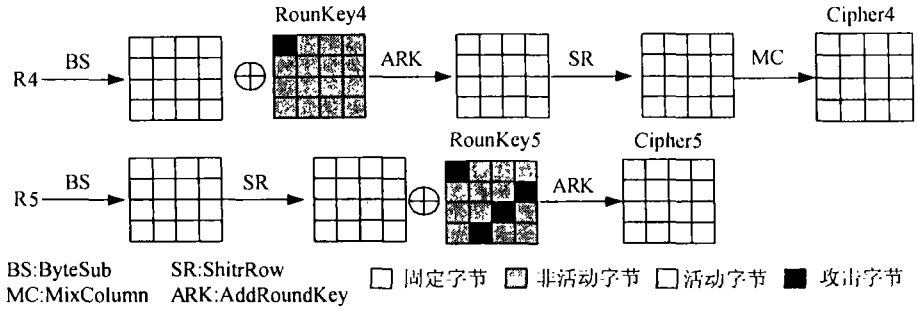

图 1 五轮 Rijndael 密码的 Square 攻击

由图 1 的描述知，由 Cipher5(1, 8, 11, 14)、RoundKey5(1, 8, 11, 14) 和 RoundKey4(1) 可以得到 Cipher3(1) 而 $\sum_{i=0}^{255}$ Cipher3(1)=0，因此通过选取 256 组明文，对其 Cipher5(1, 8, 11, 14) 进行解密猜测 RoundKey5(1, 8, 11, 14) 和 RoundKey4(1) 来验证是否有 $\sum_{i=0}^{255}$ Cipher3(1)=0，求得 4 个字节轮密钥. 为得到所有 16 个字节轮密钥需重复三次上述过程和一次穷尽搜索过程，穷尽量为 $3\times2^{40}+2^{32}$。

六轮 Square 攻击不同于五轮攻击处在于通过选择 2^{32} 组明文保证在四轮加密后，密文的对应字节和为零，所攻击的密钥是 RoundKey0(1, 5, 9, 13)、RoundKkey6(1, 8, 11, 14) 和 RoundKey5(1)，攻击强度为 2^{72}。得到所有密钥需进行两次上述搜索和一次穷尽搜索，穷尽量为 $2\times2^{72}+2^{64}$。

五轮和六轮的攻击中，我们发现每次搜索可以得到最后一轮的 4 个密钥和上一轮的 1 个密钥，其中为减少工作量将倒数第二轮的密钥加移到行移位前，它不影响穷尽结果，但不能通过前一轮的 1 个字节密钥得到后一轮密钥的更多有用信息。

3.2 逆序 Square 攻击

算法的理论根据是 Rijndael 密码的解密过程同样保持了 Square 特性. 即：如果选取只有一个字节互不相同的 256 组密文，对其解密后 256 组明文中各对应字节互不相同. 将 256 组明文对位模 2 加后得到全零序列. 逆序 Square 攻击依赖于密码的这一特性.

逆序 Square 攻击方法类似于 Square 攻击，四轮逆序 Square 攻击的算法是：

for(CheckKey0[m] = 0; k < 256; CheckKey0++){// CheckKey0[m]：表示初始密钥的第 m 个字节

CheckSum=0;

for($i = 0; i < 256; i++$)

CheckSum=CheckSum∧ S[Plain [m]∧ CheckKey[m]];

//S[ ]：表示 ByteSub 的逆变换，Plain[m]：表示明文的第 m 个字节

if (CheckSum=0)

printf("%02x", CheckKey[m]); // 初始密钥的第 m 个字节 }

四轮密码的逆序 Square 攻击与 Square 攻击相比无明显优势，唯一区别是穷尽后得到的密钥轮次不同。

五轮密码的逆序 Square 攻击较 Square 攻击，更好地利用了密码的结构特性。图 2 列出了五轮 Rijndael 密码的解密过程，详细标明了各轮解密过程中后密文的 Square 特性和五轮逆序 Square 攻击中所攻击的密钥字节及其相互关系。

由图 3 的描述可知，由 Plain(A, F, K, P)，RoundKey0(A, F, K, P) 和 RoundKey1(1) 可以得到 Cipher1(1)，而 $\sum_{i=0}^{256}$ Cipher1(1)=0，因此通过选取 256 组密文，对其 Plain(A, F, K, P) 进行加密，猜测 RoundKkey0(A,F,K,P) 和 RoundKey1(1) 来验证是否有 $\sum_{i=0}^{255}$ Cipher1(1)=0，求得 RoundKey0(A, F, K,P) 和 RoundKey1(1)，其复杂度是 $2^{40} + 3 \times 2^8$。

这里与 Square 攻击不同的是 RoundKey1(1)=A⊕S(H)，由图 3 的描述我们可以发现，一次攻击可以得到 6 个字节密钥 (A, F, K, P, H, I) 和关系 E⊕S(L), M⊕S(D)。故得到所有密钥的逆序 Square 攻击的计算量为 $2^{40} + 2^{24} + 2^{16}$。

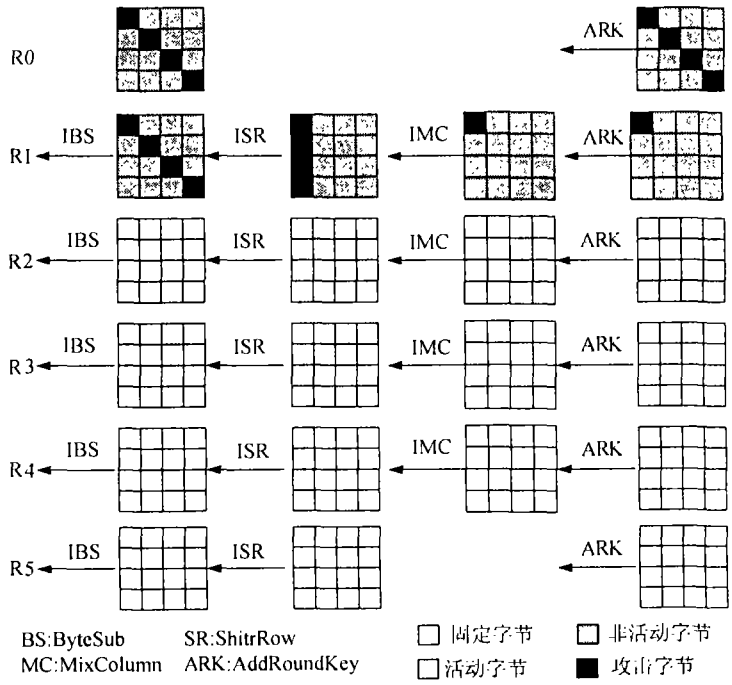

图 2　五轮 Rijndael 密码的逆序 Square 攻击中活动字节变换情况

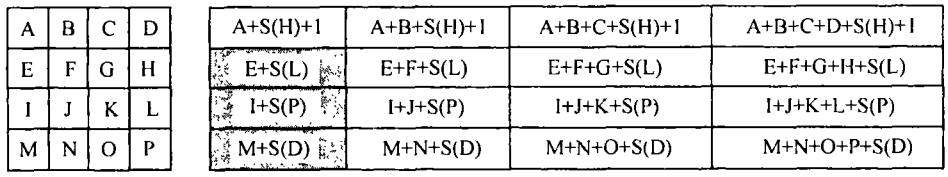

图 3 Rijndael 密钥扩展算法 (同色表示一次攻击使用的密钥)

六轮密码的逆序 Square 攻击不同于五轮的攻击在于选择明文量为 2^{32}，第一次攻击的密钥为 Roundkey0(A, F, K, P)、Roundkey1(1) 和 Roundkey6(A, E, I, M) 复杂度为 2^{72}．得到所有的密钥的攻击量为 $2^{72}+2^{56}+2^{48}$．

3.3 逆序 RotByte 对攻击的影响

逆序 RotByte 是指按照第 2 节的描述对密钥扩展算法进行变种，本小节中讨论两种攻击方法在变种后的密码攻击强度的变化．

由 3.1 节中对 Square 攻击的描述，可以看到五轮 (六轮) 密码的一次攻击得到的 5 个字节 (9 个字节密钥) 密钥之间，并无一种函数关系，因此在密钥扩展算法中改变 RotByte 的顺序，攻击强度不变，甚至如设计者所述，Square 攻击是独立于密钥扩展算法的一种攻击方法，对密钥扩展算法进行其它变形可能也不会影响密码的攻击强度．

在 3.2 节的描述中对于五轮 Square 攻击第一次攻击的密钥是 RoundKey0(A, F, K, P, A⊕S(H)⊕1)，若在密钥扩展算法中使用逆序 RotByte，第一次攻击的密钥为 RoundKey0(A, F, K, P, A⊕S(P))(如图 4)，显然攻击复杂度变为 2^{32}．

同样变种的六轮 Rijndael 的逆序 Square 攻击中第一次攻击的强度仅为 2^{64}．

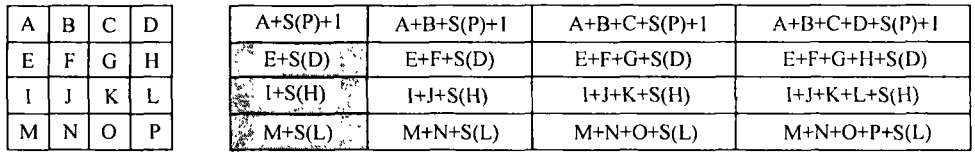

图 4 Rijndael 变种密钥扩展算法 (同色表示一次攻击使用的密钥)

4 两种攻击的强度的比较

显然，两种攻击方法都是依赖于原密码的 Square 特性，不同的是所攻击的密钥字节轮次不同，导致了攻击强度的变化．下面对两种攻击方法的攻击强度进行比较．比较结果如表 8 所示．

表 8 两种攻击算法在两种密钥扩展算法条件下的攻击强度

攻击方法			四轮	五轮变换		六轮变换	
				原密钥	变种密钥	原密钥	变种密钥
Square 攻击	一次攻击	复杂度	2^8	2^{40}	2^{40}	2^{72}	2^{72}
		密钥量 (比特)	8	32	32	32	32
	计算量		16×2^8	$3\times2^{40}+2^{32}$	$3\times2^{40}+2^{32}$	$2\times2^{72}+2^{64}$	$2\times2^{72}+2^{64}$
逆序 Square 攻击	一次攻击	复杂度	2^8	2^{40}	2^{32}	2^{72}	2^{64}
		密钥量 (比特)	8	48	32	48	32
	计算量		16×2^8	$2^{40}+2^{24}+2^{16}$	$2^{32}+2^{24}+2^{16}$	$2^{72}+2^{64}+2^{56}$	$2^{64}+2^{56}+2^{48}$

5 结论

逆序 Square 攻击与 Square 攻击一样，都是利用了密码的 Square 特性提出的攻击方法，因攻击前提不同，攻击强度有了明显的变化，对密码结构的依赖性也有了变化。六轮密码的逆序 Square 攻击中，一次攻击 9 个字节密钥，如果选择更简单的密钥扩展算法，攻击强度可能会下降更多。

参 考 文 献

[1] Daemen J, Rijmen V. AES proposal, Rijndael, In AES Round 1 Technical Evaluation CD-1, Documentation, NIST, August 1998, See http://www.esat.kuleuven.ac.be/~rijmen/rijndael/ or http://www.nist.gov/aes.

[2] Daemen J, Knudsen L, Rijmen V. The block cipher Square. Proc. of FSE'97, lecture notes in computer science 1267, 1997: 149–165.

[3] Ferguson N, Kelsey J, Stefan Lucks, Schneier B, Stay M, Wagner D, Whiting D. Improved cryptanalysis of Rijndael, AES Round 3 Technical Evaluation. NIST, August 1999, See http://www.esat.kuleuven.ac.be/~rijmen/rijndael/ or http://www.nist.gov/aes.

多　磊：　女，1976 年生，博士生，主要研究方向信息安全、分组密码的设计与分析.

李　超：　男，1966 年生，教授，博士生导师，主要研究方向代数编码与密码学、信息安全与扩频通信.