文章编号:1001-5132（2009）01-0012-05

AES 算法 EAX 加密认证模式优化技术研究

刘 丽[1]，戴振祥[1]，沈戎芬[1]，何加铭[2]

（1.宁波教育学院 信息与艺术学院，浙江 宁波 315010；2.宁波市通信芯片与射频技术重点实验室，浙江 宁波 315040）

摘要： 通过深入分析 AES 算法的 EAX 加密认证模式，提出一种新加密认证模式 PEAX (Parallel-EAX)，其中的加密和认证模式分别采用 RCTR(Random CTR)和 POMAC(Parallel OMAC)模式，使 EAX-AES 算法处理过程完全并行化，从而在软件和硬件实现上都达到满意的性能. PEAX 保持 EAX 的优点，并能使认证处理和加密处理都基于相同的加密核心单元进行的，有效地降低实现过程的复杂度. 最后并通过 Visual Studio 2005 软件仿真，验证 PEAX 模式优化的性能.

关键词： AES; EAX-AES; PEAX; 加密; 认证

中图分类号： TP39 **文献标识码：** A

实际中需要处理的消息通常是任意长的，且要求密文尽量不确定，因此引出了如何利用分组密码处理任意长度消息的问题，而解决此问题的技术就是分组密码工作模式，所以它的研究始终伴随着分组密码的研究历史.

NIST 规定认证加密模式方案有 CCM(Counter with CBC-MAC)、EAX(Encryption with Authentication for Transfer)、CWC(Carter Wegman with Counter)和 GCM(Golois Conter Mode)[1-4]. EAX 和 CCM 模式在信息长度大于 64 bit 情况下，几乎具有差不多的速度性能. EAX 模式则是模式 CCM 的优化，支持 on-line、头信息预处理、无限制的信息长度等新功能. 另一方面在硬件实现上，CWC 和 GCW 速度的提升完全依赖于认证过程的可并行化处理. 因此本文对 EAX 模式进行优化，提出一种称为 PEAX (Parallel-EAX)的模式，使其处理过程完全并行化，在软件和硬件实现上都达到满意的性能. PEAX 还具有 EAX 的另一个优点，就是认证处理和加密处理都是基于相同的加密核心单元进行的，有效地降低实现过程的复杂度，而 GCW 需 2 个独立的核心单元(Hash 函数和加密核心单元)[5-7]. 本文将针对 EAX 模式的缺点从这两方面对它进行优化，提出的新模式算法优点改进了 OMAC 算法，使数据处理可以完全并行化；并对加密 CTR 模式进行优化，提高其安全性能.

1 EAX-AES 算法原理

1.1 EAX-AES 加密认证的过程

EAX-AES 加密认证的过程如图 1 所示，其具体算法描述如下:

(1) $N' \leftarrow OMAC_K^0(N)$：以 0 为初始值，应用 OMAC 模式对 Nonce 进行认证，认证的结果是 N' 作为 CTR_K 加密模块的初始向量(IV).

收稿日期：2008-03-12. 宁波大学学报（理工版）网址: http://3xb.nbu.edu.cn
基金项目：国家自然科学基金（60372026）；宁波市自然科学基金（2008A610029）.
第一作者：刘 丽（1982），女，重庆人，硕士，主要研究方向: 无线通信. E-mail: lili_dou@126.com

(2) $H' \leftarrow OMAC_K^1(H)$：以 1 为初始值，应用 OMAC 模式对头信息(Header)进行认证，认证的结果为 H'.

(3) $C \leftarrow CTR_K^N(M)$：以 N' 为初始值，应用 CTR 模式对消息(Message)进行加密，加密后密文为 C.

(4) $C' \leftarrow OMAC_K^2(C)$：以 2 为初始值，应用 OMAC 模式对密文 C 进行认证，认证的结果为 C'.

(5) $Tag \leftarrow N' \oplus H' \oplus C'$：产生完整的认证标识 Tag.

(6) $T \leftarrow Tag$ 的低 τ 位：产生最终的认证标识 T. 最终的密文为 $C \| T$.

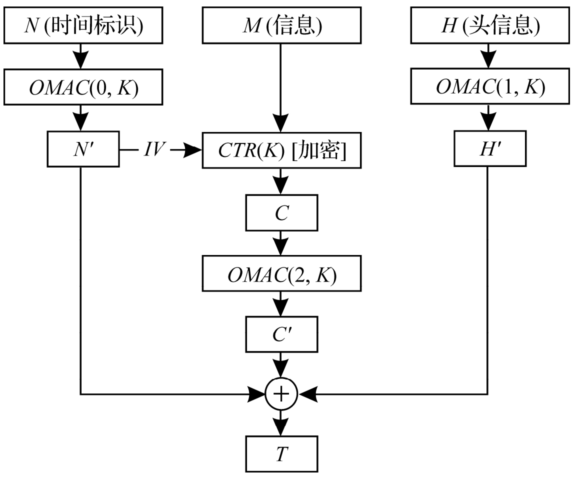

图 1　EAX 模式的加密认证过程

1.2　EAX-AES 解密认证过程

EAX-AES 解密认证的过程如图 2 所示，其算法描述如下:

(1) 如果 $|CT|$ 长度小于 τ，则认证错误并返回.

(2) $C \| T \leftarrow CT$ 从接受的信息中恢复 C 和 T，T 的长度为 τ.

(3) $N' \leftarrow OMAC_K^0(N)$：以 0 为初始值，应用 OMAC 模式对接受到的 Nonce 进行认证，认证的结果 N' 作为 CTR_K 解密模块的初始值.

(4) $H' \leftarrow OMAC_K^1(H)$：以 1 为初始值，应用 OMAC 模式对接受的 Header 进行认证，认证的结果为 H'.

(5) $C' \leftarrow OMAC_K^2(C)$：初始值为 2，应用 OMAC 模式对接受到密文 C 进行认证，认证结果为 C'.

(6) $Tag' \leftarrow N' \oplus H' \oplus C'$：根据接受的数据产生完整的认证标识 Tag'.

(7) $T' \leftarrow Tag'$ 的低 τ 位：产生最终的认证标识 T'.

(8) 如果计算出来的 T' 与接受的 T 不等，则认证错误，并返回.

(9) $M \leftarrow CTR_K^N(C)$：以 N' 作为初始值，应用 CTR 模式对消息(Message)进行解密，解密后的明文为 M.

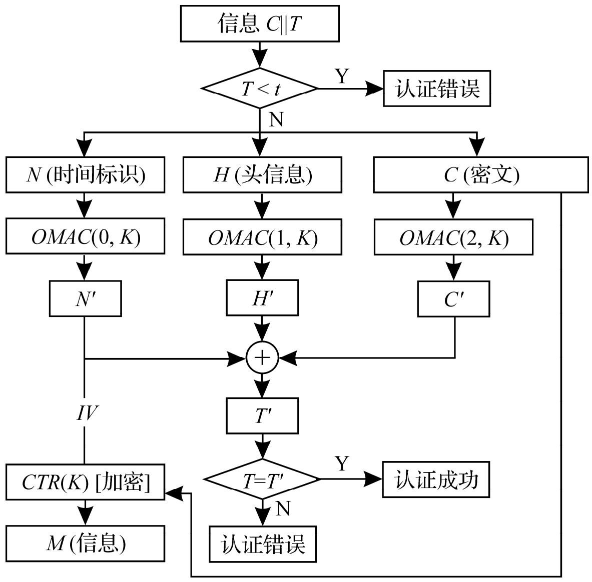

图 2　EAX 模式的解密认证过程

2　PEAX-AES 加密认证模式及优化

2.1　OMAC 算法优化——POMAC(Parallel-OMAC)

我们知道 OMAC 采用的是 CBC 模式，而 CBC 模式不利于硬件上的并行处理，降低了数据的处理速度. 因此首先考虑用 ECB 模式替换 CBC 模式，但 ECB 模式由于其自身的缺点，分组之间是完全独立，安全性能比 CBC 差. 如果采用 ECB 模式，认证码生成公式将会变为:

$$Y[i] \leftarrow E(K, M[i]) \oplus Y[i-1],$$

其最终的认证码是各个分组密文的异或. 如果输入的分组信息是相同的，则密文也是相同的，2 个密文异或后会等于 0，相当于这个明文对认证码的

值不存在任何影响. 所以单纯 ECB 模式不可行.

如果把 OMAC 的认证码生成公式:

$Y[i] \leftarrow E(K, M[i]) \oplus Y[i-1]$,

改换成:

$Y[i] \leftarrow E(K, M[i]) \oplus Y[1-N_r]$,

其中, N_r 为加密模块的轮数(128 bit 的 AES, 轮数 $N_r = 10$), 则认证过程可以进行并行处理. 这种改变在安全性能上和原来的方法没有任何区别, 不管是 $M[i] \oplus Y[i-1]$ 还是 $M[i] \oplus Y[1-N_r]$ 都只是为了引入反馈环路, 使单个的分组明文和密文不再独立而已. 因此新的 POMAC(Parallel-OMAC)模式就变成:

(1) $L \leftarrow E(K, 0^n)$:如果 $MSB(L) = 0$, 则 $L.u \leftarrow L \ll 1$, 否则 $L.u \leftarrow (L \ll 1) \oplus Cons$; 如果 $MSB(L.u) = 0$, 则 $L.u^2 \leftarrow L.u \ll 1$, 否则 $L.u^2 \leftarrow (L.u \ll 1) \oplus Cons$, ($Cons = 0 \times 0 \cdots 087$).

(2) (I)初始化: $Y[-N_r+1] \leftarrow E(K, 0^n), Y[-N_r+2] \leftarrow E(K, 1^n), \cdots, Y[0] = E(K, (N_r-1)^n)$

(II) 把信息 M 分组: $M[1], M[2], \cdots, M[m]$ (每个分组的长度为 n).

for $i \leftarrow 1$ to $m-1$ do

$Y[i] \leftarrow E(K, M[i] \oplus Y[i-N_r])$.

(III) 如果 $Y[M]$ 的长度等于 n, 则 $X[m] \leftarrow M[m] \oplus Y[m-1] \oplus L.u$, 否则,

$(X[m] \leftarrow M[m]10^{n-1-M[m]\text{的长度}}) \oplus$

$Y[m-1] \oplus L.u^2$.

$T \leftarrow E(K, X[m]), Tag \leftarrow T$ 的最低 t 比特.

POMAC 的初始化过程中需要计算 $Y[-N_r+1]$, $Y[-N_r+2], \cdots, Y[0]$ 的值. 在硬件实现上, 由于 $E_k(\bullet)$ 是并行处理, 因此初始化所需的时间只是 1 次完整的 $E_k(\bullet)$ 的计算时间. 另外, 由于在 POMAC(或 OMAC)算法的第 1 步中, 需要计算出 $L = E(K, 0^n)$ 的值, 所以初始化的过程并没有引进其他多余的运算时间.

图 3 和图 4 显示非并行处理和并行处理在框架上和性能上的区别. 假设 1 次加密运算需要的时间为 T_e, 加密算法中每轮时间都相等的, 为 T_e / N_r. 现对长度为 M 个分组的信息进行认证处理, 则 OMAC 需要的时间为 $M \times T_e$, 而 POMAC 需要的时间为 $T_e + M \times T_e / N_r$. 因此对于长信息在硬件实现上, POMAC 要比 OMAC 大约快上 N_r 倍.

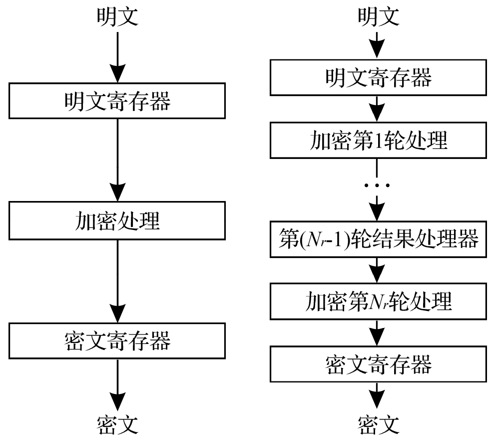

图 3 非并行处理(左)和并行处理的框图

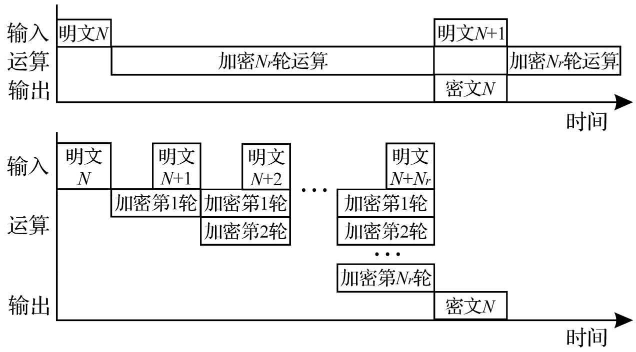

图 4 非并行处理(上)和并行处理(下)的性能比较

2.2 CTR 加密模式的优化——RCTR (Radom-CTR)

EAX 的加密模式使用 CTR 模式. 在 CTR 模式中, T_i 和 T_{i-1} 具有简单的线性关系(递增 1), 从而导致它的安全性完全决定于加密算法 $E_k(\bullet)$. 从概率论的角度上看, 由于 T_i 和 T_{i-1} 具有很强的线性关系 $T_i = Incr(T_{i-1})$, 必然会导致输出 $E_k(\bullet)$ 也具有某种相关性, 这种特征破坏了 $E_k(\bullet)$ 的安全性能. 尽管对攻击着来说, 明文是未知的, 但由于明文间的相关性却是已知的, 如果加密算法 $E_k(\bullet)$ 存在某种安全缺陷, 明文之间的相关性更容易导致整个加密被破解. 最好的方法是应用 LFSR 序列发生器产生一系列均匀分布的随机数作为 T_i 的值, 其中 LFSR 序列发生器的初始值设为 CTR 的初值. LFSR 序列

发生器实现起来很简单, 不会增加大的代价, 确能把 CTR 的安全性能提高了很多，LFSR 的初值最好要和密钥相关.

线性反馈移位寄存器(Linear Feedback Shift Register, LFSR)是基于本原多项式来产生伪随机序列的. 针对 128 bit 的加密模块, 我们选取本原多项式为: $X^{128}+X^{7}+X^{2}+X+1$，其结构如图 5.

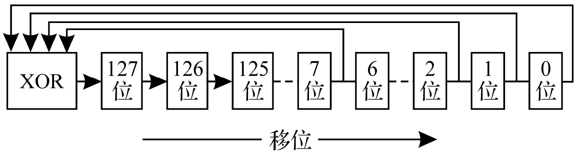

图 5　$LFSR=X^{128}+X^{7}+X^{2}+X+1$ 的结构图

RCTR(Radom-CTR)模式的描述如下:

(1) 把 IV(初始化向量)作为 LFSR 的初始值

(2) $T_i=LFSR$ 产生的新随机数, $i=1,2,\cdots,t$; $O_i=E_k(T_i), i=1,2,\cdots,t; C_i=P_i\oplus O_i, i=1,2,\cdots,t-1; C_t=P_t\oplus MSBu(O_t)$.

2.3　PEAX(Parallel-EAX)模式分析

PEAX 模式的加密认证过程如图 6 所示，其算法描述如下:

(1) $N'\leftarrow POMAC_K^0(N)$: 以 0 为初始值，应用 POMAC 模式对 Nonce 进行认证，认证的结果是 N' 作为 RCTR_K 加密模块的初始向量(IV). EAX 模式 $N'\leftarrow OMAC_K^0(N)$: 以 0 为初始值，应用 OMAC 模式对 Nonce 进行认证，认证的结果 N' 作为 CTR_K 加密模块的初始向量(IV).

(2) $H'\leftarrow POMAC_K^1(H)$: 以 1 为初始值，应用 POMAC 模式对头信息(Header)进行认证，认证的结果为 H' . EAX 模式“ $H'\leftarrow OMAC_K^1(H)$ ”: 以 1 为初始值，应用 OMAC 模式对头信息(Header)进行认证，认证的结果为 H' .

(3) $C\leftarrow RCTR_K^N(M)$: 以 N' 为初始值，应用 RCTR 模式对消息(Message)进行加密，加密后的密文为 C . EAX 模式“ $C\leftarrow CTR_K^N(M)$ ”: 以 N' 为初始值，应用 CTR 模式对消息(Message)进行加密，加密后的密文为 C .

(4) $C'\leftarrow POMAC_K^2(C)$: 以 2 为初始值，应用 POMAC 模式对密文 C 进行认证, 认证的结果为 C' . EAX 模式“ $C'\leftarrow OMAC_K^2(C)$ ”: 以 2 为初始值，应用 OMAC 模式对密文 C 进行认证, 认证结果为 C' .

(5) $Tag\leftarrow N'\oplus H'\oplus C'$: 产生完整的认证标识 Tag . EAX 模式相似.

(6) $T\leftarrow Tag$ 的低 τ 位产生最终的认证标识 T. 最终的密文为 $C\,\|\,T$. EAX 模式相似.

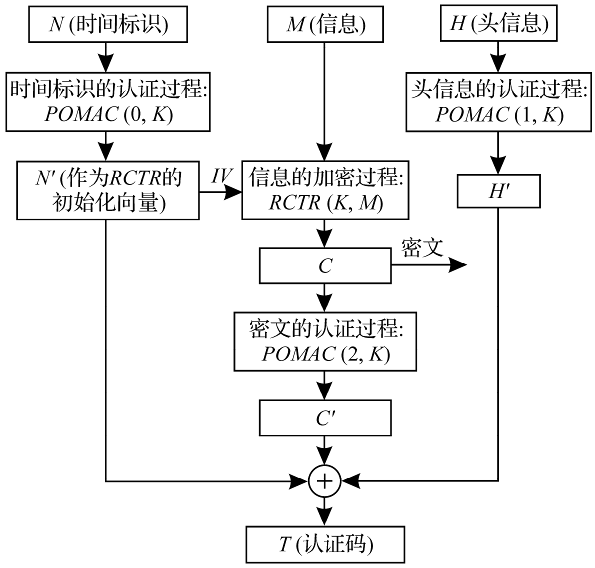

图 6　PEAX 模式加密认证过程

3　PEAX-AES 的测试结果与分析

表 1 列出了 EAX 和本文提出的 PEAX 模式的软件仿真结果(基于 Brian Gladman 的程序). 对于 PEAX 模式, 提供了 3 种比较方案: POMAC＋EAX 模式(只对 OMAC 进行并行优化)、RCTR＋EAX 模式(只对 CTR 进行优化)和 PEAX(OMAC 和 CTR 都进行优化). 由表 1 比较可知, PEAX 模式性能较好.

4　结语

通过分析 EAX 模式原理分析, 分别对其加密模式 CTR 和认证模式 OMAC 进行优化, 提出一种改进的加密认证模式方法(PEAX 模式), 该模式包含 RCTR 加密模式和 POMAC 认证模式, 一方面通过在 RCTR 模式中新加入 LFSR 序列发生器减小密

表 1 EAX 和 PEAX 模式的性能(速度)比较

信息长度/byte	EAX 模式/(byte·10^{-6} s)	POMAC+EAX 模式/(byte·10^{-6} s)	RCTR+EAX 模式/(byte·10^{-6} s)	PEAX 模式/(byte·10^{-6} s)
16	194.87	126.98	195.67	127.10
20	207.31	111.36	208.13	113.45
40	131.47	86.26	134.50	87.24
44	120.22	78.37	121.24	79.45
64	97.02	69.88	98.12	70.03
128	81.19	59.86	82.76	60.02
256	73.05	56.50	74.15	57.43
552	69.16	53.53	70.70	54.05
576	67.90	53.06	68.8	53.85
1 024	66.19	52.61	67.31	53.01
1 500	65.78	52.16	66.01	52.80
8 192	64.54	52.16	65.48	52.47
平均	66.37	52.41	67.26	52.54

注：仿真环境 CPU 为 AMD64 1.8 GMHz，内存为 1 G.

钥间的相关性，提高 EAX 模式的安全性能；另外在 OPMAC 模式中将迭代结构改变成并行处理结果，从而可提高软硬件处理速度，并由理论分析得以证明，最后并通过仿真证明了该方法的优越性.

参考文献：

[1] Federal Information Processing Standards Publication 197. Specification for AES197 [EB/OL]. [2001-11-26]. http://csrc.nist.gov/publications/tips/.

[2] Daemen J, Rjjmen V. AES proposal: Rijndael [EB/OL]. [2001-02-28]. http://csrc.nist.gov/CryptoToolkit/aes/rijndael/.

[3] Bellare M, Rogaway P, Wagner D. The EAX Mode of Operation[C]//Proceedings of the 11th Workshop on Fast Software Encryption, Lecture Notes in Computer Science, Springer Verlag, 2004.

[4] Bellare M, Rogaway P. Robust computational secret sharing and a unified account of classical secret-sharing goals[C]//Proceedings of the 14th ACM Conference on Computer and Communications Security, 2007.

[5] 马杏弛，刘亮. 高级加密标准——Rijndael 算法介绍与探讨[J]. 装备指挥技术学院学报, 2003(1):96-100.

[6] 吴文玲，冯登国. 分组密码工作模式的研究现状[J]. 计算机学报, 2006, 29(1):21-36.

[7] 张清华. Rijndael 算法的高效实现及其性能分析[J]. 计算机应用, 2004(2):78-92.

Research on Optimizing Encryption-authentication Mode of EAX in AES Algorithm

LIU Li[1], DAI Zhen-xiang[1], SHEN Rong-fen[1], HE Jia-ming[2]

(1.Information and Arts Institute, Ningbo College of Education , Ningbo 315010, China;
2.The Ningbo Key Lab of Communication Chips & RF Technology, Ningbo 315040, China)

Abstract: The security issue has been remaining central in the research field of AES. Based on the thorough study on the cipher mode of AES, this paper proposes a new authentication-encryption mode called PEAX (Parallel-EAX), which consists of RCTR(Random CTR) and POMAC(Parallel OMAC) mode. The newly introduced authentication-encryption mode enables the parallel processing of encryption and authorization, and can achieve better performance both in software and hardware implementation. Without sacrificing the advantages of EAX, both encryption and authentication processing in PEAX are conducted on the basis of a single cipher core unit such as AES, resulting with lowest complexity in implementation. The simulation result indicates that the proposed PEAX functions sufficiently well in the Visual Studio 2005 development environment and is up to the technical expectation.

Key words: AES; EAX-AES; PEAX; encryption; authentication

CLC number: TP39 **Document code:** A

（责任编辑 章践立）