文章编号:1001 - 9081(2005)01 - 0135 - 03

工作于 CBC 模式的 AES 算法可重配置硬件实现

刘 航,戴冠中,李晖晖,慕德俊
(西北工业大学 自动化学院,陕西 西安 710072)
(lhang_2k@hotmail.com)

摘 要:分组加密算法的工作模式选择对于敏感信息的安全至关重要。文中采用可重配置硬件设计了一个工作于 CBC 模式的 AES 核,并对关键单元的硬件设计进行了优化。仿真和实验测试结果表明,设计的 AES 核能够稳定地工作于 CBC 模式,实现对敏感信息的高速加密处理。

关键词:高级加密标准;密码分组链接;可重配置硬件实现
中图分类号: TP393 **文献标识码**:A

Reconfigurable hardware implementation of AES algorithm in CBC mode

LIU Hang, DAI Guan-zhong, LI Hui-hui, MU De-jun
(College of Automation, Northwestern Polytechnical University, Xi'an Shaanxi 710072, China)

Abstract: The operation mode of block cipher algorithm is vital to sensitive information security. In this paper, an Advanced Encryption Standard (AES) core which works in Cipher Block Chaining (CBC) is developed in reconfigurable hardware. The design of the key unit is optimized based on the analysis of AES. Simulation and experimental results show that in CBC mode the core can work steadily and encrypt/decrypt sensitive information in high speed.

Key words: advanced encryption standard; cipher block chaining; reconfigurable hardware implementation

0 引言

AES[1]作为新的加密标准,能够为敏感信息提供比 DES、3DES 更加安全可靠的保护,已经成为被广泛推荐使用的分组加密算法。与此同时,针对不同应用,NIST(National Institute of Standards and Technology)为 AES 算法定义了 5 种可供选择的操作模式:CBC(Cipher Block Chaining,密码块链接)、ECB(Electronic Code Book,电子密码本)、CFB(Cipher FeedBack,密码反馈)、OFB(Output FeedBack,输出反馈)和 CTR(Counter,计数器)。其中,CBC 模式能够防止从具有相同数据的明文产生相同的密文,使得对于敏感信息的加密处理更加安全,是目前 IPSec 协议规范草案中推荐使用的加密工作模式之一。[2,3]

然而,AES 算法的计算复杂性将导致关键计算处理设备性能的下降,因此,迫切需要硬件加密芯片为高速、安全的信息保密服务提供支持。FPGA(Field Programmable Gate Array)等可重配置硬件具有开发成本低,设计周期短,编程灵活,易于调整,能够根据需要重新配置硬件功能,并且可以提供比软件实现更快的处理速度,是目前 AES 算法硬件实现研究中的热点之一[4-8]。

但是,由于工作于 CBC 模式下的 AES 算法在硬件实现时难以采用完全展开、流水线等并行处理技术,不能对同一数据块的不同数据分组进行并行加密处理,无法显著提高计算处理设备的吞吐量[9,10]。因此,研究工作于 CBC 模式的 AES 算法的可重配置硬件实现,对于改善关键处理设备的安全处理性能和实时性具有重要意义。

本文在分析 AES 算法各部件及其工作模式的基础上,对 AES 算法硬件实现中的关键单元进行了优化,采用硬件描述语言设计了一个能工作于 CBC 模式的 AES 核,仿真和实验测试结果表明,本文设计的 AES 核在 CBC 工作模式下可以稳定的工作,数据吞吐量达 610Mbps,可以用于实现对网络数据包加密等计算密集型任务的加速处理。

1 AES 算法与 CBC 工作模式

AES 算法是一个典型的密钥迭代分组加密算法。其数据

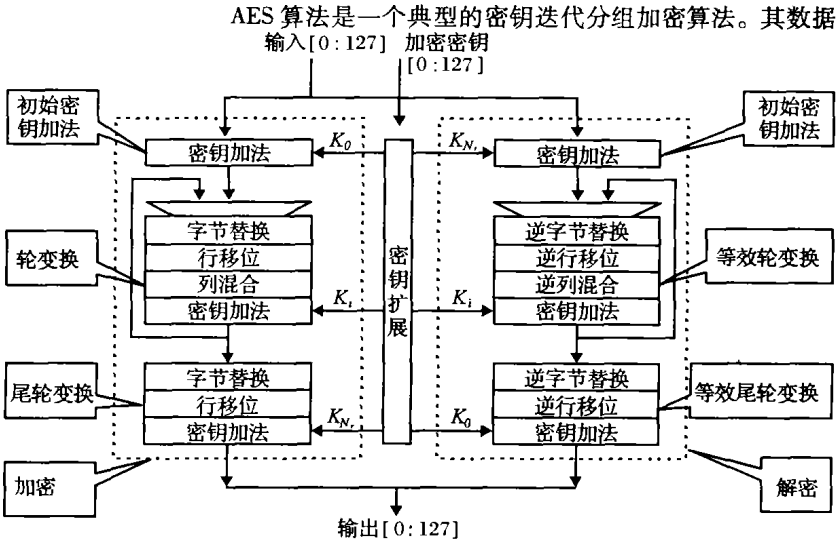

图 1 AES 算法结构图

收稿日期:2004 - 07 - 08;**修订日期**:2004 - 12 - 05 **基金项目**:国防基础研究项目(J1300B005)

作者简介:刘航(1973 -),男,讲师,博士研究生,主要研究方向:智能信息处理、网络与信息安全; 戴冠中(1937 -),男,教授,博士生导师,主要研究方向:自动控制、信息安全.

分组长度为 128 位，并且支持 128 位、196 位和 256 位密钥。如图 1 所示，AES 算法由加密、解密和密钥扩展等三个模块构成。

加密模块由一个初始密钥加法 AddRoundKey、N_r-1 次轮变换 Round 和一个尾轮变换 FinalRound 构成。其中，128 位数据分组构成的 4 × 4 状态矩阵 State 和轮密钥 ExpandedKey[i]组成初始密钥加法和所有轮变换的输入。

为了实现方便，解密模块一般采用与加密模块相同的步骤次序：一个初始密钥加法、N_r-1 次等价轮变换 EqRound 和一个等价尾轮变换 EqFinalRound，只是在 EqRound 和 EqFinalRound 中的每一步被改成 Round 和 FinalRound 中相应步骤的逆，并相应改变密钥选择次序即可。

密钥扩展模块（KeyExpansion）实现加解密密钥（CipherKey）的扩展，导出用于初始密钥加法的一个轮密钥 ExpandedKey[0]和 N_r 个用于轮变换的密钥 ExpandedKey[i]。

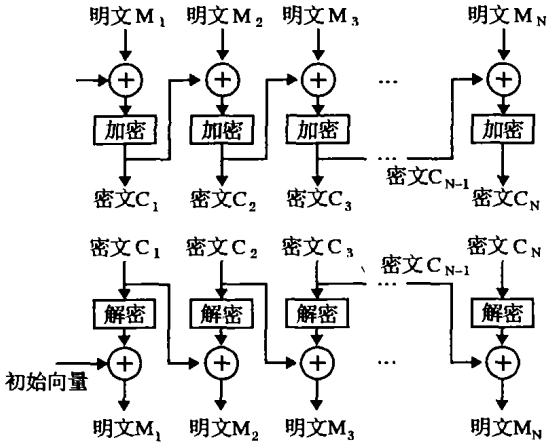

图 2 CBC 工作模式

为了防止从具有相同数据的明文产生相同的密文，使得对于敏感信息的加密处理更加安全，提高密码分析的复杂度，CBC 工作模式被广泛采用，并且是目前 IPSec 协议规范草案中推荐使用的工作模式之一。如图 2 所示，在 CBC 工作模式下，每个数据块的第一个 128 位明文分组 M_1 在加密前先与一个随机产生的初始向量 Ⅳ 进行异或，然后再对该结果进行加密处理得到第一个密文分组 C_1，该数据块的后续明文分组 M_2、M_3、…、M_N 则必须先与前一个分组的加密结果 C_1、C_2、…、C_{N-1} 进行异或运算后再进行加密处理；解密过程中，第一个密文分组 C_1 解密后与初始向量 Ⅳ 进行异或运算得到第一个明文分组 M_1，其他密文分组 C_2、C_3、…、C_N 在解密后则必须与前一个密文分组 C_1、C_2、…、C_{N-1} 进行异或运算后才能得到正确的明文 M_2、M_3、…、M_N。由此可见，在 CBC 模式下无法实现对同一数据块不同数据分组的并行加密或解密处理，难以像 ECB 模式那样使用展开和流水线等技术获得高性能的处理。

2 工作于 CBC 模式的 AES 算法可重配置硬件实现

2.1 AES 算法的 CBC 模式实现结构

AES 算法中的行移位单元仅涉及状态矩阵各字节的移位操作，直接的信号换位可以缩短电路延迟时间；而密钥加法单元也仅需要一次逻辑异或运算即可实现。相对而言，字节替换单元、列混合单元以及密钥扩展模块的实现较为复杂，是决定 AES 算法硬件实现性能的关键。因此，针对 CBC 模式的特点，优化 AES 算法中字节替换单元、列混合单元以及密钥扩展单元的设计成为本文的重点。与此同时，考虑到 CBC 模式与 ECB 模式的共同之处，本文设计的 AES 核可以进行工作模式的选择：当工作于 CBC 模式时，随机产生的初始向量或前一密文分组被用来进行加密前或解密后的异或运算；当工作于 ECB 模式时，AES 核强制采用 128 全“1”作为每次加密前或解密后进行异或运算的初始向量，从而使得该 AES 核能够满足敏感信息存储和传输等不同应用场合的加密需求，其实现结构示意图如图 3。

在该结构中，数据加密和解密模块被单独实现，每个加密或解密的轮运算步骤均在 1 个时钟周期内完成。因此，一个 128 位数据分组经过 11 个时钟周期可以完成加密或解密计算，进一步的优化还可以使其具有并行处理不同数据块中两个数据分组的加密或解密运算的能力，有效提高数据吞吐量。

2.2 字节替换单元实现

SubBytes 和 InvSubBytes 的计算结果仅与当前进行的是加密还是解密运算有关，与密钥无关，因此，AES 算法中的字节替换单元可以采用查表操作代替有限域上的仿射变换 f、f^{-1} 和乘法逆运算而加以实现，使 AES 算法硬件实现的电路迟延减小。

又根据实验测试，当使用硬件描述语言的（with…select…when…）语句进行 S 盒设计时，Quartus II 环境下的编译结果表明：一个用于对 32 位数据进行置换的 S 盒需要使用 4 × 256

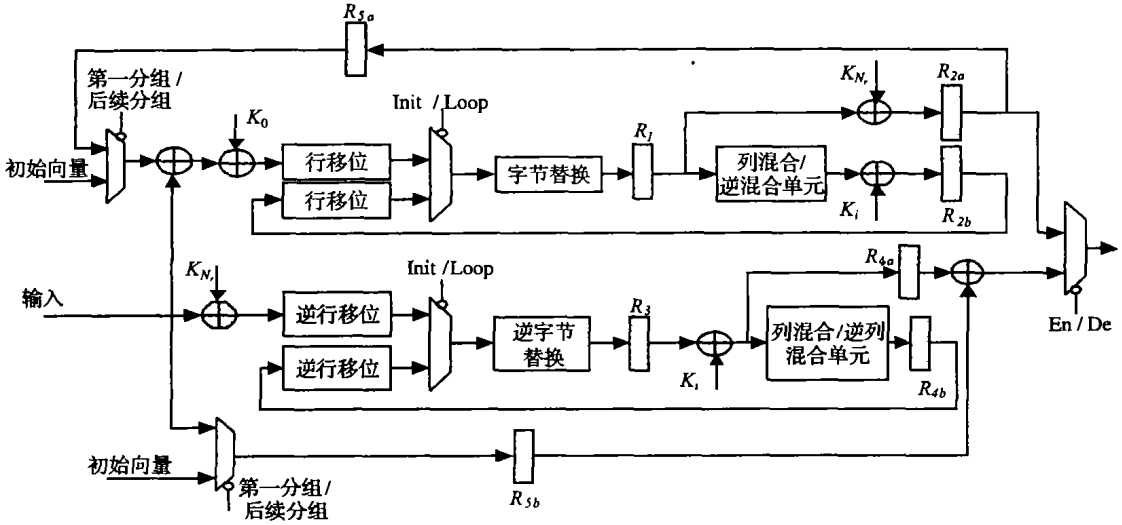

图 3 工作于 CBC 模式的 AES 算法实现结构图

位的查找表，占用 832 个逻辑单元，约占 ALTERA EP20KE200EFC-484-2x 芯片逻辑单元资源的 10%。

为了达到优化的芯片运算处理性能，本文使用 Altera 芯片中的 ESB(Embeded System Block)，分别实现用于加密运算的 S_{RD} 和解密运算的 S_{RD}^{-1}。由于 ESB 专门用于设计类似高速 RAM、CAM 和 ROM 等器件，在电路性能方面，访问延迟仅有 4.7ns 左右，在逻辑单元的占用和访问延迟时间方面均具有优势。

2.3 列混合单元实现

MixColumns 变换将状态矩阵各列视为 $GF(2^8)$ 上次数小于 4 的多项式，并被同一固定多项式

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

进行模 $x^4 + 1$ 的乘法运算：

$$b(x) = c(x) \cdot a(x) \pmod{x^4 + 01} \quad (1)$$

由于 $GF(2^8)$ 中的每一个元素都可以表示为 2 的不同幂次之和，乘以任意常数的乘法都可以通过反复调用 xtime(乘以 0x02 的函数)来实现。因此，AES 算法中的 MixColumns 单元用下面给出的方法来实现[1]：

$$\begin{aligned} &t = a_0 \oplus a_1 \oplus a_2 \oplus a_3 \qquad u = a_0 \\ &v = a_0 \oplus a_1 \quad v = xtime(v) \quad b_0 = a_0 \oplus v \oplus t \\ &v = a_1 \oplus a_2 \quad v = xtime(v) \quad b_1 = a_1 \oplus v \oplus t \\ &v = a_2 \oplus a_3 \quad v = xtime(v) \quad b_2 = a_2 \oplus v \oplus t \\ &v = a_3 \oplus u \quad v = xtime(v) \quad b_3 = a_3 \oplus v \oplus t \end{aligned} \quad (2)$$

与 MixColumns 相比，InvMixColumns 变换更为复杂，其系数矩阵由{0x09、0x0B、0x0D、0x0E}组成，硬件上的实现会占用较多的逻辑单元，并造成硬件时延的增加，导致整体性能降低。为了降低系统的资源占用率和电路延迟，必须尽可能实现电路复用。又由于 MixColumns 单元中的模乘多项式 $c(x)$ 与 InvMixColumns 单元的模乘多项式 $d(x)$ 具有如下关系：[1]

$$d(x) = (04x^2 + 05)c(x) \pmod{x^4 + 01} \quad (3)$$

因此，一个相对简单的预处理步骤和一个 MixColumns 环节将简化 InvMixColumns 单元的实现。

$$\begin{aligned} &u = xtime(xtime(a_0 \oplus a_2)) \\ &v = xtime(xtime(a_1 \oplus a_3)) \\ &b_0 = a_0 \oplus u, \quad b_1 = a_1 \oplus v \\ &b_2 = a_2 \oplus u, \quad b_3 = a_3 \oplus v \end{aligned} \quad (4)$$

其中，状态矩阵每一列中 4 个字节的 MixColumns 和 InvMixColumns 运算结果 b_0、b_1、b_2、b_3 都可以看作先将输入列的 4 个字节 a_0、a_1、a_2、a_3 分别进行循环向上移位 0 个、1 个、2 个和 3 个字节，再进行与式(2)或者式(4)中 b_0 相同的计算过程。一个使用 1 字节 MixColumns 和 InvMixColumns 基本复用运算模块的 4 字节 MixColumns 和 InvMixColumns 单元，如图 4 所示。

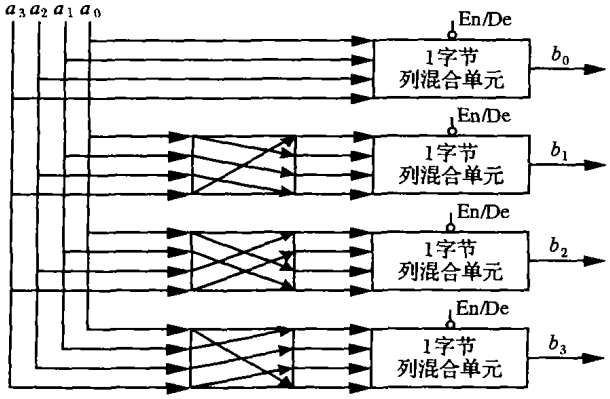

图 4　4 字节的 MixColumns 和 InvMixColumns 运算单元

2.4 密钥扩展模块的实现

加解密过程中每一轮都需要一个 128 位的轮密钥，直接实现的密钥扩展模块将使加解密模块长期处于等待轮密钥的状态，导致处理性能降低；与此同时，由于对使用同一密钥进行加密或解密的数据分组，只需要一次密钥扩展即可获得用于加密和解密的运算轮密钥。因此，为了保证轮密钥能够在进行数据加密和解密前获得，确保加解密效率，借鉴文献[10]的方法，设计实现了密钥扩展计算模块(图 5 所示)，并将由该模块提前计算出的轮密钥结果存储于存储器中，供加解密模块的使用。

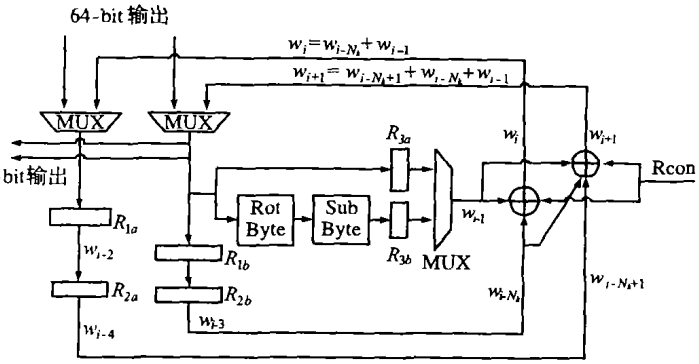

图 5　密钥扩展计算模块

在该密钥扩展计算模块中，初始加解密密钥以 64 位为单位被输入。每一次轮密钥计算的第 1、2 个周期分别输入前一轮的轮密钥，并在第 2 个周期进行 SubByte(RotByte(Ki))的计算，于是便可在第 3、4 个时钟周期分别计算获得当前轮的轮密钥；作为下一轮密钥扩展的输入，当前轮的轮密钥在第 3、4 个周期被输入，两个时钟周期后即可获得新一轮的轮密钥，依此类推。采用该方法，花费 22 个时钟周期即可完成一个 10 轮的 AES 加解密密钥扩展处理。

3 仿真及测试结果

为了测试本文设计的 AES 核的性能和可靠性，我们使用 ALTERA EP20KE200EFC-484-2x 作为实验载体分别进行了软件仿真实验和实验板测试。这个 128 位密钥长度的 AES 核既可工作于 CBC 模式，又可工作于 ECB 模式；静态时序分析表明，该 AES 核在 CBC 工作模式下正常工作情况时的最大工作频率可以达到 52.6MHz，由此计算得到其最高数据吞吐量可以达到 610Mbps。进一步的测试实验采用自制电路测试板对该 AES 核的正确性、稳定性和可靠性进行测试，并在 50MHz 工作频率下并获得了正确、稳定、可靠的测试结果，考虑到静态时序分析软件在电路参数和仿真结果估计上的保守性，我们有理由相信本文设计的 AES 核能够稳定的工作于 CBC 模式，并达到 610Mbps 以上的加解密处理速度。

4 结语

本文基于 FPGA 设计、实现并测试了一个能工作于 CBC 和ECB两种模式的AES核。并对关键单元的硬件设计进行

(下转第 140 页)

中,其实例为 I。

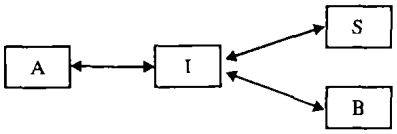

图5 系统入侵模型

3.3 协议安全属性的 CTL 描述

(1) 认证性

通信的发起者和通信的响应者之间能够相互认证对方的身份。如果 B 要认证 A 的身份,那么在任何情况下 A 作为发起者开始与 B 会话的次数 $A\_Session$ 不小于 B 作为响应者与 A 完成会话的次数 B_Count_auth, 用 CTL 描述为: $AG(A\_session \geq B\_count\_auth)$, 如果存在 $A\_session < B\_count\_auth$, 则说明存在入侵者冒充 A 向 B 进行认证。同样,我们可以用 $AG(B\_session \geq A\_count\_auth)$ 来描述初始者 A 对响应者 B 的身份认证。

(2) 密钥的保密性

保密性要求如果一个密钥被通信双方接收作为会话密钥,则除非该密钥泄露,否则它未被入侵者 I 得知。用 CTL 描述为:

$$\sim E((A\_session\_K_{ab} = K_{ab} \ \&\ B\_session\_K_{ab} = K_{ab})) \ U \ I\_get\_K_{ab} = K_{ab})$$

(3) 密钥的新鲜性

新鲜性要求新的会话密钥的值不能与曾用过的某一密钥重合。用 CTL 描述为:

$$AG(\ Used\_K_{ab} = K_{ab} \text{ -> } (Session\_K_{ab} \sim = K_{ab}))$$

3.4 模型的检验结果与分析

将上述协议模型和协议安全要求输入到 SMV 中, 发现协议模型并不符合安全协议的要求。经分析该协议不能抵抗重传攻击。设攻击者 I 存储一条旧提出提出报文 $\{K_{ab}, B\}_{K_{as}}$, 并从中破译旧的会话密钥 K_{ab}, 则 I 可以做如下攻击:

$M1$ $A \rightarrow I:A, \{N_a, A\}_{K_{as}}$; I 截获 A 发给 B 的提出报文 $M1$。

$M2$ $I \rightarrow S:I, A, \{N_I, C\}_{K_{is}}, \{N_a, A\}_{K_{as}}$; I 冒充 B 把提出报文 $M2$ 发给 S。

$M3$ $S \rightarrow I:\{K_{is}, A\}_{K_{is}}, \{N_I, N_a, \{K_{as}, I, N_a\}_{K_{is}}\}_{K_{is}}$; S 误认为是 I—A—S 间的正常通信,故把提出报文 $M3$ 发给 I。

$M3'$ $I \rightarrow A:\{K_{ab}, B\}_{K_{as}}, \{N_a, N_i, \#\#\#\}_{K_{ab}}$; I 冒充 S 把报文 $M3'$ 发给 A,其中 ### 可为任意值。

$M4$ $A \rightarrow I:\{\#\#\#\}_{K_{bs}}, \{N_i\}_{K_{ab}}$; I 冒充 B 截获报文 $M4$。这样,I 成功地使 A 相信,A 和 $B(I)$ 之间获得一轮新的会话密钥 K_{ab}。以后,I 就可以冒充 B,利用 K_{ab} 进一步获取 A 的机密。

5 结语

论文对密码协议模型检测的方法给出了一些理论上的探讨,并用 SMV 检测工具给出了一个实际分析的例子。笔者认为模型检测的优点是能够自动进行,当系统模型不满足时,模型检验器会自动生成不满足系统规格的反例,据此可以进一步修改模型以满足所需求的属性。同时符号模型检测还缓解了组合爆炸问题,因而对复杂的密码协议的分析检测也是一种行之有效的方法。

参考文献:

[1] LOWE G. Breaking and Fixing the Needham-Schroeder Public-Key Protocol using FDR[A]. Proc. of TSCAS'96. Berlin: Springer Verlag[C], 1996. 147 - 166.

[2] HOARE CAR. Communicating Sequential Processes[M]. Prentic-Hall, 1985.

[3] DANG Z, KEMMERER R. Using the ASTRAL Model Checker for Cryptographic Protocol Analysis[A]. DIMACS Workshop on Design and Formal Verification of Security Protocols[C], 1997.

[4] MARRERO W, CLARKE E, JHA S. A Model checker for Authentication Protocols[A]. DIMACS Workshop on Design and Formal Verification of Security Protocols[C], 1997.

[5] MITECHELL J, MITECHELL M, STERN U. Automated Analysis of Cryptographic Protocols using Mur[A]. Proc. of the IEEE Symposium on Security and Privacy[C]. USA: IEEE Computer Society Press, 1997. 141 - 151.

[6] SMV[EB/OL]. http://www-cad.eecs.berkeley.edu/~kenmcmil.

(上接第 137 页)

了优化。仿真和实验测试结果表明,本文设计的 AES 核能够实现对敏感信息的高速加密处理。在此基础上,进一步的优化可以使其具有不同数据块中两个数据分组加密或解密运算的并行处理能力,再次提高数据吞吐量,适应安全性要求提高对于关键处理设备处理能力的挑战。

参考文献:

[1] DAEMEN J, RIJMEN V. The Design of Rijndael: AES - The Advanced Encryption Standard[M]. Springer-Verlag Berlin Heidelberg, 2002.

[2] IP Security Protocol(ipsec)[EB/OL]. http://www.ietf.org/html.charters/ipsec-charter.html.

[3] The AES-CBC Cipher Algorithm and Its Use with IPsec[EB/OL]. http://ietf.org/rfc/rfc3602.txt

[4] DANDALIS A, PRASANNA VK, ROLIM JDP. A Comparative Study of Performance of AES Final Candidates Using FPGAs[A]. Workshop on Cryptographic Hardware and Embedded Systems[C], 2000.

[5] ELBIRT AJ, YIP W, CHETWYND B, *et al.* An FPGA Implementation and Performance Evaluation of the AES Block Cipher Algorithm Finalists[A]. The Third Advanced Encryption Standard (AES3) Candidate Conference[C], April 2000.

[6] ELBIRT A, PAAR C. An FPGA-Based Performance Evaluation of the AES Block Cipher Candidate Algorithm Finalists[A]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems[C], 2001.

[7] GAJ K, CHODOWIEC P. Comparison of the Hardware Performance of the AES Candidates Using Reconfigurable Hardware[A]. The Third Advanced Encryption Standard (AES3) Candidate Conference[C], April 2000.

[8] MCLOONE M, MCCANNY J. High Performance Single-Chip FPGA Rijndael Algorithm[A]. Workshop on Cryptographic Hardware and Embedded Systems[C], 2001.

[9] BELLOWS P, FLIDR J, LEHMAN T, *et al.* GRIP: A Reconfigurable Architecture for Host-Based Gigabit-Rate Packet Processing[A]. Proceedings 10th Annual IEEE Symposium on Field-Programmable Custom Computing Machines. FCCM 2002[C], 2002.

[10] CHODOWIEC P, GAJ K, BELLOWS P, *et al.* Experimental Testing of Gigabit IPSec-Compliant Implementation of Rijndael and Triple DES Using SLAAC-1V FPGA Accelerator Board[A]. Proc. Information Security Conference[C], 2001.