

高吞吐率 XTS-AES 加密算法的硬件实现

李子磊, 刘政林, 霍文捷, 邹雪城

(华中科技大学 电子科学与技术系, 湖北 武汉 430074)

摘 要: 基于 XTS-AES 算法提出了一种具有并行全流水结构的硬件实现方法. 设计通过展开数据通路的方式, 提高了吞吐率; 同时还通过采用内部流水线结构优化关键路径的方式, 提高了电路的时钟频率和整体工作性能. 在 UMC 90 nm CMOS 工艺条件下, 所设计的 XTS-AES 模块的吞吐率比目前已知 XTS-AES 的最高吞吐率提高了 52.28%. 分析结果表明, 该硬件模块完全满足现阶段高速加密存储的需要.

关键词: 高速存储; 高吞吐率; 并行全流水结构; XTS-AES 加密算法

中图分类号: TN4

文献标识码: A

文章编号: 1000-7180(2011)04-0095-04

A High-Throughput Hardware Implementation of XTS-AES Encryption Algorithm

LI Zi-lei, LIU Zheng-lin, HUO Wen-jie, ZOU Xue-cheng

(Department of Electronic Science & Technology,

Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract: This paper proposes a new hardware implementation method for XTS-AES Algorithm that has a full parallel pipelined structure. The proposal scheme increases throughput by unrolling the data path. Meanwhile, it also improves the circuit clock frequency and overall performance by using inner pipelined structure to optimize the critical path. Compared with the currently known highest throughput XTS-AES implementation, the new XTS-AES module increases the throughput by 52.28% in UMC 90 nm CMOS technology. The result indicates that this hardware module fully meets the need of high-speed encrypted storage at present.

Key words: high-speed storage; high-throughput; fully parallel pipelined structure; XTS-AES Encryption algorithm

1 引言

近年来, 磁盘和磁带驱动器上个人敏感数据的丢失成为一个日益严重的威胁. 针对这个问题, IEEE 1619 安全存储委员会(SISWG)2007 年批准了“静止状态数据(data at rest)”在磁盘和磁带存储应用上的草案, 并于 2008 年 4 月正式公布了 XTS-AES 算法标准, 该算法主要用于以数据单元(包括扇区、逻辑磁盘块等)为基础结构的存储设备中静止状态数据的加密^[1], 能满足高性能 RAID 应用, 固态硬盘驱动器, 串行 SCSI(从 G1 到 G3)和光纤通道的需要. 随着该算法的日益推广, 越来越多的外国厂商

制作了高性能的 IP, 如 Elliptic^[2] 和 IP Cores^[3], 他们制作的 XTS-AES IP 核的吞吐率分别达到了 16 Gb/s 和 13.7 Gb/s.

本文针对 XTS-AES 算法适用于高吞吐率的通信环境, 设计了一种并行全流水结构的 XTS-AES 硬件加密模块, 通过对数据通路进行展开以及对关键路径进行优化, 使得 XTS-AES 硬件加密模块提高了吞吐率和性能, 能够满足固态硬盘等高速加密存储的需要.

2 XTS-AES 加密算法描述

文献[4]首先提出了可调整的分组密码的概念,

收稿日期: 2010-04-06; 修回日期: 2010-06-21

基金项目: 国家自然科学基金项目(60973034); 2007 新世纪优秀人才支持计划(NCET-07-0328)

跟传统的分组密码相比,除了密钥和明文这两个输入以外,可调整的分组密码多了一个输入,这个输入被称作调整值(Tweak).

XTS-AES是一种可调整的分组密码,它适用于对那些被分为固定长度数据单元的数据流信息进行加密.这些数据单元又被分为 $m+1$ 个数据块,前 m 个数据块 $P_0 \cdots P_{m-1}$ 的大小为128 bit,最后一个数据块 P_m 的大小为 b bit(b 在0到127之间).数据单元最多包含128-bit,数据块的个数为 $2^{128}-2$.

2.1 单个128-bit数据块的XTS-AES加密算法

可以用下式表示:

$$C \leftarrow \text{XTS-AES-blockEnc}(\text{Key}, P, i, j) \quad (1)$$

其中Key为256-bit或512-bit XTS-AES加密密钥(两个128-bit Key或两个256-bit Key之和);

P 为128-bit明文数据块;

i 为128-bit调整值;

j 为128-bit数据块在数据单元中的位置值.

C 为128-bit密文数据块.

具体如图1所示.

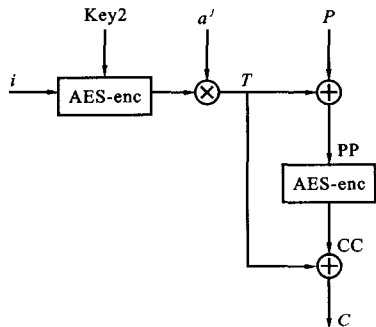


图1 单个128-bit数据块的XTS-AES加密算法示意图

其中AES-enc为标准AES算法,Key2为调整值密钥,Key1为数据密钥,模乘操作中为 $\text{GF}(2^{128})$ 域中对应于多项式 x 的本原.计算的步骤顺序如下:

$$(1) T \leftarrow \text{AES-enc}(\text{Key}_2, i) \otimes \alpha^j$$

$$(2) PP \leftarrow P \oplus T$$

$$(3) CC \leftarrow \text{AES-enc}(\text{Key}_1, PP)$$

$$(4) C \leftarrow CC \oplus T$$

其中的模乘定义如下:

调整值经过AES加密算法得到的密文被转变成字节阵列 $a_0[k], k=0, 1, 2 \cdots 15$,经过和 α 的 j 次方模乘后,得到的字节阵列 $a_j[k], k=0, 1, 2 \cdots 15, i$ 从0到 j ,

$$a_{i+1}[0] = (2(a_i[0] \bmod 128)) \mid (135 \mid a_i[15] / 128)$$

|)

$$a_{i+1}[k] = (2(a_i[k] \bmod 128)) \mid a_i[k-1] / 128, k=1, 2, \cdots 15^{[1]}$$

2.2 数据单元的XTS-AES加密算法

在数据单元的加密算法中,算法的应用跟最后一个数据块的大小 b 有关:

① 当 $b=0$ 时,即数据单元的大小是128的整数倍,这时对 m 个128 bit数据块进行128 bit数据块的XTS-AES加密算法,如下式所示:

for $q=0$ to $q=m-1$, do ,

$$C_q \leftarrow \text{XTS-AES-blockEnc}(\text{Key}, P_q, i, q) \quad (2)$$

② 当 $0 < b < 128$ 时,前 $m-1$ 个数据块进行128 bit数据块的XTS-AES加密算法,如下式所示^[1]:

for $q=0$ to $q=m-2$, do ,

$$C_q \leftarrow \text{XTS-AES-blockEnc}(\text{Key}, P_q, i, q) \quad (3)$$

对最后两个数据块应用密文窃取模式得到相应的密文,具体如图2所示.

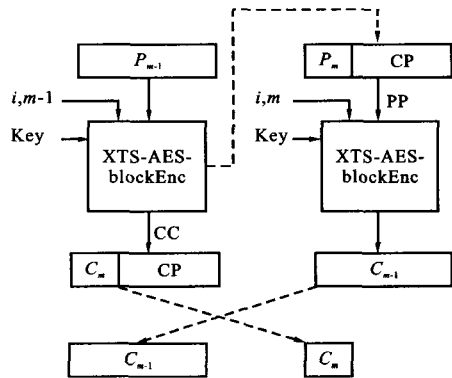


图2 密文窃取模式示意图

其中XTS-AES-blockEnc为单个128-bit数据块的XTS-AES加密算法,经过这样一个变换,整个数据单元中的明文全部被加密成为等长度的密文.

鉴于该算法的特点和应用环境,发现吞吐率是影响该算法应用的最主要因素,因此如何设计出高吞吐率的XTS-AES加密算法是难点所在.

3 XTS-AES算法的并行全流水结构实现

3.1 XTS-AES数据通路设计

XTS-AES算法主要应用于RAID应用和固态硬盘(SSD)等高性能环境,吞吐率是该算法最为重要的一项指标,因此对XTS-AES算法展开成流水结构来换取该算法的高数据吞吐率是比较有效的一个方法.

本文所设计的XTS-AES模块主要由流水AES模块和模乘模块以及输入输出FIFO组成.这里选

取的数据单元为一个标准扇区的大小(512 Bytes),即一个数据单元包含 32 个 128-bit 数据块. 数据通路如图 3 所示. 流水 AES 核主要负责对调整值和明文数据进行 AES 加密,为了节省面积开销,通过多路器对明文和密钥进行选择,这样就只需设计一个流水 AES 核;模乘模块主要负责完成加密后的调整值在 $GF(2^{128})$ 域中和 α 的 j 次方的模乘操作,共在 32 个周期里进行 32 次模乘操作,并且模乘后的结果需要寄存;由于 XTS-AES 加密算法所在的高速加密时钟和外部低速时钟属于不同的时钟域,就用到了明文 FIFO 和密文 FIFO. 在流水 AES 核的数据通路上,每次模乘后的结果最后都需要和流水 AES 核的密文进行异或才能得到最后的密文,因此采用了 2 组 32 选 1 多路器,一组用来选择和输入明文进行异或的模乘结果,另一个用来选择和流水 AES 核输出密文进行异或的模乘结果. 当 XTS-AES 计算开始后,调整值和输入明文以 128 位数据为单位,首先在密钥 Key2 的作用下完成调整值的 AES 加密,明文数据顺序移入明文 FIFO,然后和每次模乘的结果进行异或操作后进入流水 AES 核,经过 21 个周期,流水线正式建立,以后每过一个周期就会得到一组经 AES 加密后的密文,该密文和相应的模乘值异或后得到的最终密文再送入 FIFO,这样就经过了一个数据单元的 XTS-AES 加密.

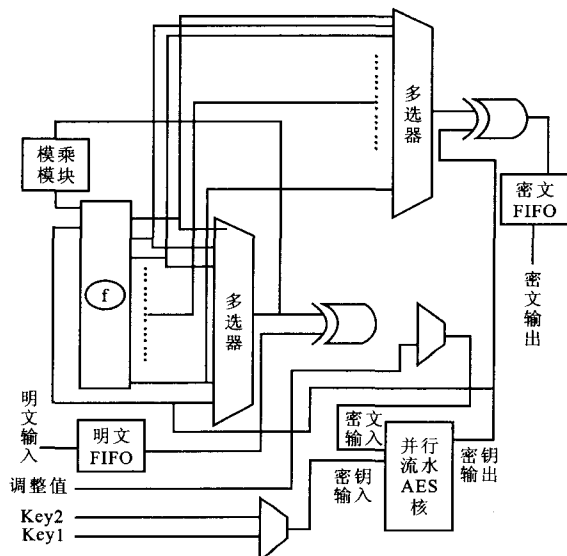


图3 整体电路数据通路

3.2 并行全流水 AES 加密算法模块的设计

AES 算法在结构上属于密钥迭代的分组密码算法^[5],它的数据分组长度固定为 128 比特,密钥长度为 128 比特,192 比特,256 比特,相应轮数为 10 轮、12 轮或 14 轮^[6]. 为了能达到最大的吞吐率,本

文采用并行全流水进行设计,并以 128 bit 为单位进行数据处理和密钥输入,这样就可以在开始时每一个周期输入一个新的数据,在流水线建立后,相应的每一个周期就可以得到一个结果. 因为 128-bit AES 变换是由轮函数通过 10 轮迭代实现的,对于流水线的设计至少需要 11 个时钟周期完成流水线的建立时间. 设基本 AES 结构的最小时钟周期为

$$T_{\text{basic}} = t_{\text{round}} + t_l \quad (4)$$

t_{round} 表示逻辑电路的延迟时间, t_l 表示寄存器的建立时间和传输延迟时间之和,则速度为

$$\text{Throughput}_b = 128 / (Nr \times T_{\text{basic}}) \quad (5)$$

那么 k 级流水线结构的速度为

$$\text{Throughput}_{\text{pipeline}} = k \cdot \text{Throughput}_b \quad (6)$$

但为了达到最大的速度/空间比,本文设计了一种具有内部流水线结构的 AES,在轮间 10 级流水线结构的基础上,再把每一轮的操作进行分割,即在每一轮的 ByteSub 和 ShiftRow 之间再添加寄存器,这些分割得到的组合逻辑电路部分称为流水线站,这样就把每一轮操作分为 2 级流水线站,分割之后电路可以处理多个数据分组,提高系统在单位时间内处理的数据量. 由于算法本身的结构,这个流水线站不是随意划分的,是根据电路中最大功能部件的延时决定^[7].

由于 ByteSub 使用的是传统的查找表,通过综合后的 report,发现这一步组合逻辑的延时达到了 1.1 ns,而整个一轮操作组合逻辑的延时为 2.02 ns,故 ByteSub 操作的延时时间在一轮延时中的比例占到了 54.42%,因此,导致时钟频率达不到很高,鉴于这个原因,在 ByteSub 和 ShiftRow 中增加了一级流水线,这样其速度可以达到 10 级流水线的速度的 2 倍,但面积只有很小的增加^[8]. 由于 ByteSub 这级流水线站的延时和剩下的一级流水线站的延时基本相等,那么 $2Nr$ (这里 $Nr = 10$) 个时钟周期内,并行处理的分组数为 $2k$ 个,令

$$\tau = t_l / t_{\text{round}}$$

那么内部流水线的最小时钟周期为

$$T_{\text{sp}} = (0.5 + \tau) / (1 + \tau) \cdot T_b$$

其速度为

$$\text{Throughput} = k \cdot (1 + \tau) / (0.5 + \tau) \text{Throughput}_b$$

在非反馈模式下,内部流水线结构的电路面积与流水线级数 k 成正比. 虽然面积也随流水线站数的增加而增加,但比例很小,增加的部分主要是寄存器的面积,由于 τ 很小,所以可以看出,其速度为同

级流水线结构的 2 倍,其面积却与同级流水线的面积相当。

由于内部流水线结构是通过提高时钟频率的方式来提高处理速度的,当流水线站数 r 增加时,空间的相应增加极小,因而它具有最高的速度 / 空间比。

因此最终设计了带 2 级流水线站的 10 级全流水 AES 结构。

3.3 模乘模块的设计

在 XTS-AES 算法中,模乘模块是比较重要的一环,这一步需要进行大量的模运算和取整运算,因此必须对该操作予以简化,以降低逻辑电路的复杂度。把调整值经过 AES 算法加密得到的结果(这里叫 $EncTweak_0$)经过移位和异或就可以得到模乘的结果 T 。算法简化如下:

调整值在 $Key2$ 的作用下经过 AES 算法得到的结果为 $EncTweak_0$,由于是第一次加密,所以位置值 j 为 0,所以在这一步不进行模乘操作,所以 $T_0 = EncTweak_0$ 。在进行第二次加密时,先把 $EncTweak_0$ 的高位和低位进行互换,即把高八位换到第八位,低八位换到高八位,依次类推。换位完成后判

断第 127 位的值,如果为 0,则左移一位,如果为 1,则先左移一位,得到的结果低八位再和十六进制数 87 相异或。最后得到的结果为 T_1 。这样,根据明文在数据单元的位置,就可以得到每次模乘操作的结果,简化算法公式如下:

```
Cin=0;
for (j=0; j<16; j++)
{Cout=(T[j]>>7)& 1;
T[j]=(T[j]<<1)+Cin) & 0xFF;
Cin=Cout;}
if (Cout)
T[0]^=0x87;
```

4 性能分析与比较

使用 Verilog 硬件描述语言分别实现了带 2 级流水线站的 10 级全流水 AES 电路和整体 XTS-AES 电路,并使用 Synopsys Design Compiler 在 UMC 90 nm 标准 CMOS 单元库下综合,将综合结果与已有的 XTS-AES IP 核相比后,得到如表 1 所示的结果。

表 1 性能比较

算法	最大频率/MHz	最大吞吐率/(Gb/s)	等效门数	实现工艺/nm
带 2 级流水线站的 XTS-AES 电路	262	33.53	203×10^3	90
Elliptic CLP-47 ^[2]	250	16	410×10^3	未知
IP Cores XTS3 ^[3]	215	13.7	140×10^3	90

由以上综合结果可知,在吞吐率方面,采用 2 级流水线站的 XTS-AES 电路比文献[2]提高了 52.28%,比文献[3]则提高了 59.15%,获取了较好的吞吐率性能;在运算效能上,本文以吞吐率除以面积得到单位面积的吞吐率作为参考指标,新的设计与文献[2]中设计相比提高了 76.37%,比文献[3]中设计提高了 40.75%,较好地处理了面积和速度之间的折中关系。在运行的时钟频率方面,由于对传统的流水结构的 AES 核和整体电路的关键路径进行了改进和优化,因此比文献[2]中设计的时钟频率提高了 4.58%,比文献[3]中的则提高了 17.94%。综合结果表明本文设计的 XTS-AES 电路具备较高的吞吐率和整体性能,能够应用于高速储存的需要。

5 结束语

本文提出了一种 XTS-AES 硬件实现方法,通

过对 10 轮流水 AES 增加两级流水线站,优化模乘模块和数据通路,最终提高了整个电路的吞吐率。本文所设计的 XTS-AES 模块的吞吐率比目前已知 XTS-AES 的最高吞吐率提高了 52.28%,因而本文所提出的设计完全能满足磁盘加密设备对高吞吐率的要求以及可以作为在线存储、近线存储和离线存储的数据安全方案。

参考文献:

[1] IEEE Std 1619—2007. Institute of Electrical and Electronics Engineers. The XTS—AES Tweakable Block Cipher[S]. IEEE, 2008.

[2] Elliptic Technologies Inc. CLP—47 Configurable XTS—AES Core Product Brief[EB/OL]. [2010—01—12]. www.elliptictech.com.

[3] IP Cores. Inc. XTS3 Family of Cores, IEEE P1619 XTS—AES Cores[EB/OL]. [2010—09—25]. www.ipcores.com.

给网络可信度评估模块. 系统评估的内容主要有: 系统安全性评估、系统可靠性评估、系统可维度评估等量指标. 通过自律管理者使系统的可信度维持在一个稳定的范围内.

3.4 反馈调节机制

两个自律元素之间的信息交互, 是整个系统运行及参数自主调节的关键. 当系统安全性评估的结果偏低, 而网络连接可信度评估结果相对较高时表明把攻击信息误判为可疑信息的比例高, 需要增大可疑信息区间的下确界. 当系统安全性评估的结果偏高, 而网络连接可信度评估结果相对较低时表明把可信信息误判为可疑信息的比例高, 需要减小可疑信息区间的上确界. 系统平稳运行的时候, 安全性评估曲线与网络连接可信度评估曲线应基本相一致, 如果差别过大时就需要对可信度阈值进行调节, 其调节过程可以用图 6 来模拟.

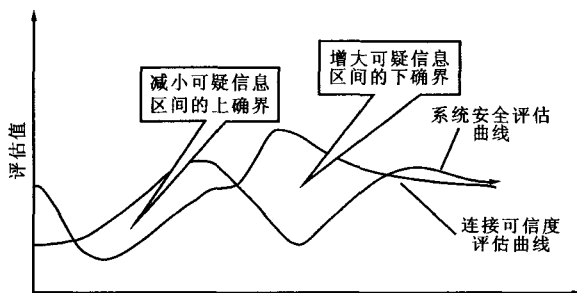


图 6 阈值调节过程示意图

4 结束语

本文借鉴了自律计算的思想以及自动控制中的

反馈的机制, 构建了一个入侵容忍系统模型, 具备较好的自适应性, 能动态调整各个子模块的运行参数, 使系统对已知和未知的攻击具有容忍能力.

将自律计算与入侵容忍相结合的研究还处于起步阶段, 相关的方法和理论还需要进一步的研究和完善. 计算机系统评估技术在此模型中也起到了关键的作用, 计算机系统评估技术与传统的入侵容忍技术相结合将是一个很好的研究方向.

参考文献:

- [1] 崔竞松, 王丽娜, 张焕国. 一种并行容侵系统研究模型—RC 模型[J]. 计算机学报, 2004, 27(4): 500—506.
- [2] 殷丽华, 方滨兴. 入侵容忍系统安全属性分析[J]. 计算机学报, 2006, 29(8): 1505—1512.
- [3] 李冰洋, 王慧强. 自律入侵容忍系统的模型构建与量化分析[J]. 计算机应用研究, 2009, 26(5): 1883—1886.
- [4] 邓林, 余刘琅. 基于入侵容忍原理的服务器防护技术[J]. 微电子学与计算机, 2008, 25(6): 143—146.
- [5] 廖备水, 李石坚. 自主计算概念模型与实现方法[J]. 软件学报, 2008, 19(4): 779—802.

作者简介:

吴庆涛 男, (1975—), 博士, 副教授, 硕士生导师. 研究方向为网络信息安全.

华彬 男, (1983—), 硕士研究生. 研究方向为网络安全.

郑瑞娟 女, (1980—), 博士, 讲师. 研究方向为网络信息安全、可信计算.

(上第 98 页)

- [4] Moses Liskov, Ronald L Rivest, David Wagner. Tweakeable Block Ciphers[C]// CRYPTO 2002, LNCS 2442, 2002. Berlin Heidelberg: Springer—Verlag, 2002: 31—46.
- [5] 王海洋, 陈弘毅. 一种小面积的高吞吐率 AES 协处理器设计[J]. 微电子学与计算机, 2009, 26(6): 12—16.
- [6] 余国义, 李皓, 刘国希, 等. 一种适用于 DSP 的安全模块的设计[J]. 微电子学与计算机, 2009, 26(7): 1—4.
- [7] 华漫. AES 算法分析及其硬件实现设计[D]. 成都: 西南交通大学, 2004: 36—38.

- [8] 卢坚. 高级加密标准研究及其 FPGA 实现[D]. 成都: 西南交通大学, 2002: 41—44.

作者简介:

李子磊 男, (1985—), 硕士研究生. 研究方向为信息安全、数字 IC 设计.

刘政林 男, (1968—), 教授. 研究方向为集成电路设计、无线传感器网络、嵌入式系统安全等.