引用本文：王赜坤，陈松涛：优化的 AES 算法在光传送网中的应用与实现[J]. 光通信技术，2019，43(5)：47-50.

优化的 AES 算法在光传送网中的应用与实现

王赜坤[1]，陈松涛[2]

（1.武汉华夏理工学院，武汉 430223；2.烽火通信科技股份有限公司，武汉 430073）

摘要：为了解决光传送网络(OTN)中数据传输的安全问题，提出了将高级加密标准(AES)算法应用于 OTN 系统中以保护数据的安全性。描述了 OTN 系统的加密技术，介绍了 AES 加密算法。为降低 AES 算法的运算复杂度、减少运算时间和节约资源，提出了一种将加密运算过程进行合并优化的解决方案。通过仿真和网络测试，验证了该优化方案的可行性和通信的安全性。

关键词：光传送网；加密；AES 算法；合并优化

中图分类号：TN929.11　**文献标识码**：A　**文章编号**：1002-5561(2019)05-0047-04

DOI：10.13921/j.cnki.issn1002-5561.2019.05.012　**开放科学(资源服务)标识码(OSID)**：

Application and implementation of optimized AES algorithm in optical transport network

WANG Zekun[1], CHEN Songtao[2]

(1. Wuhan Huaxia Institute of Technology, Wuhan 430223, China;
2. Fiberhome Telecommunication Technologies Co., Ltd., Wuhan 430073, China)

Abstract: In order to achieve the security problem of data transmission in optical transport network(OTN), the application of advanced encryption standard (AES) algorithm in OTN system to protect data security is proposed. The encryption technology of the OTN system is described, and the AES encryption algorithm is introduced. In order to reduce the computational complexity of the AES algorithm, reduce the computation time and save resources, a solution for merging and optimizing the encryption operation process is proposed. The feasibility of the optimization scheme and the security of communication are verified by simulation and network test.

Key words: OTN; encryption; AES algorithm; merge and optimization

0 引言

随着无线通信技术、移动互联网技术的飞速发展，网络数据流量呈现出“爆炸性”的增长，业务的丰富性对传送网能力和性能的要求更高。光传送网(OTN)技术因能满足各种新型业务的需求，而成为传送网发展的主要方向。然而，未经加密的数据很容易受到恶意的攻击，所有传送网上的数据都应有条件地进行加密，保证这些数据信息对接收者以外的人不可见[1,2]。目前，OTN 系统中的加密多为软件加密方式，这种方式易遭攻击者采用分析程序进行跟踪、反编译等手段进行破译。同时高级加密标准(AES)算法因其算法简单、密钥装载快和需要的内存空间少等优点被用在各种领域。因此，本文利用优化的 AES 算法，采用现场可编程阵列(FPGA)硬件加密的方式，对 OTN 系统中的业务信息 OPUk 净荷进行加密，既能保证数据传输的安全性，又可以提高数据处理的效率性。

收稿日期：2019-01-22。

基金项目：武汉华夏理工学院科研基金项目(批准号：18034)资助。

作者简介：王赜坤(1981-)，女，湖北武汉人，硕士，讲师，主要从事光通信技术、集成电路设计等方向的科学研究与教学工作；主持参与湖北省教育厅项目 3 项，校级教科研项目、质量工程项目等多项。

1 AES 算法简介

AES 算法具有 128bit 的固定明文长度，密钥长度可以独立指定 128bit、192bit 和 256bit，相应的迭代轮数为 10 轮、12 轮和 14 轮[3]。密钥长度为 128bit 的算法(AES-128)因迭代次数比较少、运算速度比较快，故适用于低功耗与高速率的应用场景，如对温度（要求

65℃的外围环境）和功耗要求比较苛刻的外置机房、边缘城域的接入点等，同时更适用于高速率信号的传输，如 100G 及以上的 OTN 系统传输。

虽然在安全程度上，密钥长度 256bit 比 128bit 的安全，但 256bit 比 128bit 大概需要多花 40%的运算时间，用于多出的 4 轮密钥生成以及对应的短进程优先（SPN）操作。此外，生成 256bit 的密钥也比生成 128bit 的密钥占用更多的资源。故本文主要讨论密钥长度为 128bit 的密码算法，其基本运算过程包含如下 4 个步骤：字节代替（SubBytes）、行移位（ShiftRows）、列变换（MixColumns）和轮密钥加（AddRoundKey）[4]。

AES 算法有 5 种工作模式：电子密码本、加密块链、加密反馈、输出反馈和计数器模式。相较于其它 4 种工作模式，计数器是完全的流模式，可以并行加密，实现简单（仅需要加密算法），而且能对任意长度的明文进行加密。故 OTN 中采用计数器模式下的 AES 算法对业务信息数据进行加密。计数器模式下，AES 加密的并不是明文，而是计数器的值，密文由计数器加密结果与明文相异或产生。

2 OTN 中 AES 算法的加密和密钥传输

2.1 OTN 中 AES 算法加密原理

在 OTN 系统中，客户业务信息加密是基于加密单板的物理端口[5]，通过标准 AES-128 加密算法对承载客户业务信息的 OPUk 净荷区进行加/解密。加密后的 OPUk 数据可在 OTN 中透明传输，不影响 OTN 的网络监视管理。同时可透传客户业务时钟，不影响时间时钟同步特性。在客户侧业务映射到 ODUk 信号过程中的介入处理流程中实现业务加密功能，如图 1 所示。在源端，客户业务信息首先映射到 OPUk 层进入加密流程，加密模块对 OPUk 净荷数据加密，即对 OTU 帧结构中第 17~3824 列数据进行加密。加密完成之后把装载密文净荷的 OPUk 信号按照 ITU-T G.709 定义成帧封装流程，映射成 ODUk 低阶。增加低阶 ODUk 层开销字节，复用到 ODUk 高阶，再增加高阶 ODUk 层开销字节，封装 OTUk 开销字节。之后增加帧头字节、前向纠错（FEC）字节，形成完整的 OTUk 帧信号，最后通过电/光转换模块变成光信号经光纤传输到宿端。在宿端，光线路模块将接收到的光信号转换成电信号，并恢复 OTUk 帧格式。从 OTUk 帧信号中解调出高阶 ODUk 和解复用出低阶 ODUk，之后通过解密模块解出密文、恢复出明文净荷，即从 OPUk 信号中解出客户业务信息。

2.2 OTN 加密模块结构

OTN 加密模块的作用是实现客户业务信息的通信安全性，保证数据传输的完整性和机密性，其结构如图 2 所示。

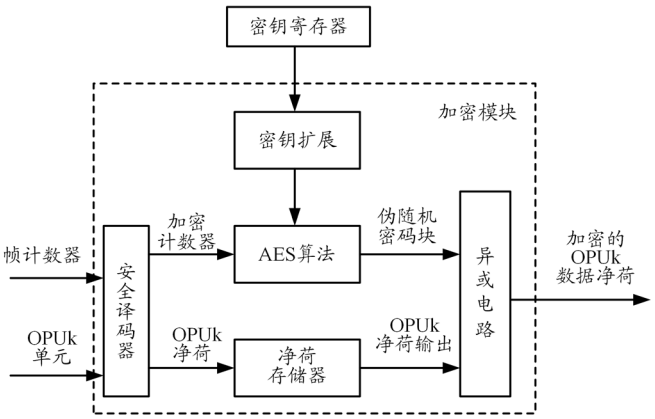


图 2 OTN 加密模块结构

图 2 中，安全译码器模块将帧计数器中的值作为 AES 算法的输入，并取出 OPUk 单元中的 OPUk 净荷（payload），输入到 OPUk 净荷存储器。AES 算法模块对计数器中的值进行运算，生成伪随机密码块。最后，伪随机密码块和相应的 OPUk 有效净荷被发送到异或电路模块，得到加密后的 OPUk 业务信息数据。

在 OTN 系统中，OTUk 帧由 3 个部分组成：OTUk 开销、ODUk 帧（帧开销和 OPUk 净荷）和 OTUk FEC。加密模块只对 OTUk 帧中的 OPUk 净荷区数据进行连续加密，而不对 OTUk 帧头（即不包括开销、FEC 区域）进行加密。OTUk 帧共 4 行 4080 列，以字节为单位，总共有 4×4080=16320 字节，其中第 17~3824 列为 OPUk 净荷区信息，数据最大长度为 4×3808=15232 字节，若净荷数据长度超过该值则需要进行分片加密。本文采用 AES 算法对帧计数器值进行加密，输入 128bit 固定

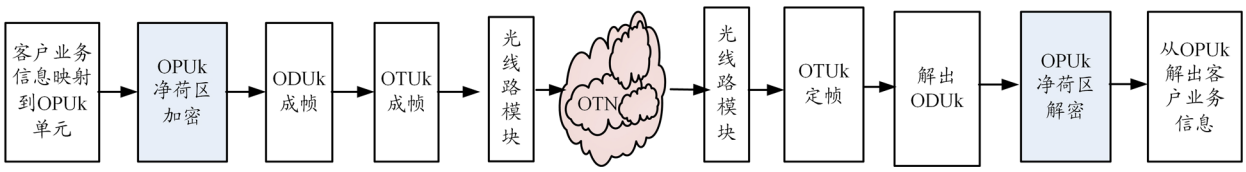


图 1 OTN 中 AES 算法加密原理框图

长度的密钥，因此，需要先将 32 位的帧计数器值进行复制，得到 128bit 的数据，然后将该数据作为 AES 算法的输入，通过加密运算生成 128bit 的伪随机密码块，最后伪随机密码块与用户净荷数据异或得到密文。

2.3 在 ODUk 上传送密钥信息

OTN 系统中的密钥交换和传输，目前还没有规定相关的标准。ITU-T G.709 定义了 ODUk 开销中的一些保留字节(RES 字节)。通过 RES 字节，安全信道可以在 ODUk 通道甚至边界建立(64 字节 OTUk 帧开销结构可参阅文献[1])。本文密钥的传输仅选择 OTUk 帧的 ODUk 层开销第 4 行第 9~14 列共 6 个 RES 字节，连续传输 3 帧即可，通过 FPGA 模块处理密钥的编码，然后与 OTUk 成帧芯片 Framer 的并行开销口进行对接，将密钥的字节插入 RES 开销位，伴随 OTUk 信号流进行传递。因此，密钥的传输没有增加额外的带宽消耗。相比 AES-256 的密钥需要 6 帧数据，本文采用 AES-128 的密钥总共 16 个字节，传输密钥仅需连续 3 帧数据即可完成密钥的传输，在密钥的传输上更加高效，减少了因密钥传输引入的数据时延。

3 OTN 中 AES 算法的优化

由于 OTN 系统对通信安全和通信延时有非常高的要求，故下文将对 AES 算法进行优化，以减少 AES 算法运行时间和资源的占用，同时保证算法在严苛条件的情况下也能执行。

由 AES-128 加密流程可知，加密运算的迭代过程前 9 轮完全相同，各轮变换都会涉及到多项式在有限域 GF(2^8)上的乘法。经前 9 轮加密运算的输出为：

$$\begin{aligned}\begin{bmatrix} s''''_{0,c} \\ s''''_{1,c} \\ s''''_{2,c} \\ s''''_{3,c} \end{bmatrix} = &\left\{ \{02\}\otimes \begin{bmatrix} S[s_{0,c}] \\ S[s_{1,c+1}] \\ S[s_{2,c+2}] \\ S[s_{3,c+3}] \end{bmatrix} \right\} \oplus \\ &\left\{ \{01\}\otimes \begin{bmatrix} S[s_{1,c+1}] \\ S[s_{2,c+2}] \\ S[s_{3,c+3}] \\ S[s_{0,c}] \end{bmatrix} \right\} \oplus \left\{ \{03\}\otimes \begin{bmatrix} S[s_{2,c+2}] \\ S[s_{3,c+3}] \\ S[s_{0,c}] \\ S[s_{1,c+1}] \end{bmatrix} \right\} \oplus \\ &\left\{ \{01\}\otimes \begin{bmatrix} S[s_{3,c+3}] \\ S[s_{0,c}] \\ S[s_{1,c+1}] \\ S[s_{2,c+2}] \end{bmatrix} \right\} \oplus \begin{bmatrix} k_{0,c} \\ k_{1,c} \\ k_{2,c} \\ k_{3,c} \end{bmatrix} \end{aligned} \tag{1}$$

其中，$s_{r,c}$ 为输入的数据，$k_{r,c}$ 为每一轮的轮密钥，$s''''_{r,c}$ 为经过前 9 轮加密运算之后的数据，$S[s_{r,c}]$ 为字节代替运算后的数据。r=0,1,2,3；c=c,c+1,c+2,c+3，表示行右循环分别移动 0,1,2,3 个字节。在进行列变换运算时，每一列都与一个固定的多项式相乘，{01}、{02}和{03}为相乘的多项式的系数。为了降低整个运算的复杂度，在加密过程中，对每一轮运算的过程进行合并，列变换与轮密钥的异或如式(2)所示。

$$\begin{bmatrix} s''''_{0,c} \\ s''''_{1,c} \\ s''''_{2,c} \\ s''''_{3,c} \end{bmatrix} = \begin{bmatrix} s'''_{0,c} \\ s'''_{1,c} \\ s'''_{2,c} \\ s'''_{3,c} \end{bmatrix} \oplus \begin{bmatrix} k_{0,c} \\ k_{1,c} \\ k_{2,c} \\ k_{3,c} \end{bmatrix} \tag{2}$$

其中，$s'''_{r,c}$ 为迭代过程前 9 轮中每一轮经过列变换后的数据。

最后一轮加密没有列变换，在合并中可以直接去掉列变换矩阵如式(3)所示，即生成伪随机密文块。

$$\begin{aligned}\begin{bmatrix} s'''''_{0,c} \\ s'''''_{1,c} \\ s'''''_{2,c} \\ s'''''_{3,c} \end{bmatrix} &= \begin{bmatrix} s''_{0,c} \\ s''_{1,c} \\ s''_{2,c} \\ s''_{3,c} \end{bmatrix} \oplus \begin{bmatrix} k_{0,c} \\ k_{1,c} \\ k_{2,c} \\ k_{3,c} \end{bmatrix} = \begin{bmatrix} s'_{0,c} \\ s'_{1,c+1} \\ s'_{2,c+2} \\ s'_{3,c+3} \end{bmatrix} \oplus \begin{bmatrix} k_{0,c} \\ k_{1,c} \\ k_{2,c} \\ k_{3,c} \end{bmatrix} = \\ &\begin{bmatrix} S[s''''_{0,c}] \\ S[s''''_{1,c+1}] \\ S[s''''_{2,c+2}] \\ S[s''''_{3,c+3}] \end{bmatrix} \oplus \begin{bmatrix} k_{0,c} \\ k_{1,c} \\ k_{2,c} \\ k_{3,c} \end{bmatrix} \end{aligned} \tag{3}$$

其中，$s'''''_{r,c}$ 为经过 10 轮加密运算之后得到的伪随机密文，$s''_{r,c}$ 为行移位运算后的数据，$s'_{r,c}$ 为字节代替运算后的数据，$S[s''''_{r,c}]$ 为前 9 轮加密运算之后的数据在第 10 轮中经过字节代替、行移位运算后的数据。

由于 $\{01\}\otimes S[x]=S[x]$，$\{02\}\otimes S[x]=2S[x]$，故定义 2 张表，令 $T_1[x]=S[x]$、$T_2[x]=2S[x]$，$T_1[x]$ 是 S 盒所构成的查找表，$T_2[x]$ 是 S 盒中每个元素在有限域 GF(2^8)中乘以 2 得到的查找表。由于{03}可以用($\{02\}\oplus x$)表示，故式(1)中 $\{03\}\otimes S[x]=\{02\}\otimes S[x]\oplus S[x]$，则 $T_3[x]=T_2[x]\oplus T_1[x]$，式(3)进行 S 盒查找表变换为：

$$\begin{bmatrix} s'''''_{0,c} \\ s'''''_{1,c} \\ s'''''_{2,c} \\ s'''''_{3,c} \end{bmatrix} = \begin{bmatrix} T_1[s''''_{0,c}] \\ T_1[s''''_{1,c+1}] \\ T_1[s''''_{2,c+2}] \\ T_1[s''''_{3,c+3}] \end{bmatrix} \oplus \begin{bmatrix} k_{0,c} \\ k_{1,c} \\ k_{2,c} \\ k_{3,c} \end{bmatrix} \tag{4}$$

根据上面合并优化过程可知，加密过程中只需要

2 张表（$T_1[x]$，$T_2[x]$）和简单的异或电路即可完成。算法合并优化后得到的伪随机密码块再与 OPUk 中的净荷数据相异或，得到加密后的客户业务信息。这样将加密中关于乘法的复杂运算简化为查表，可以减少运算的许多中间环节和运算时间。

4 OTN 的 AES 算法功能仿真实验

本文采用 Xilinx ISE 作为开发环境进行测试，选择 Xilinx 10 系列 FPGA 进行编译，使用 Modelsim 软件执行程序的功能仿真，并观察数据。在加密仿真图中，“text_in”为明文即 OPUk 中的净荷数据，“text_out”为密文即 OPUk 中加密之后的净荷数据。“ld”为帧计数器的值，作为 AES 算法运算的输入与“key”进行加密运算；当“done”变为高电平时，完成对明文（text_in）的加密，变为密文（text_out）。在解密仿真过程中，将加密得到的密文（text_out）作为解密运算的明文（text_in）数据，且保持帧计数器的值与密钥均不变进行解密运算，当解密运算完成时得到的解密数据（text_out）与加密的明文（text_in）完全相同，这证明了算法的正确性。同时，本文是采用 FPGA 的硬件方式加密，整个加密流程在黑盒（只能看到输入或输出等信息，无法看到加密运算中的信息）内完成，具有抗窃取性和攻击性特点，其安全性高于业界普遍使用的软件加密方式。

为了更进一步做比较，证明本文设计思路的合理性和优化措施的有效性，我们采用未优化的设计方案[6]实现 AES 算法加密，经 FPGA 仿真比较可知：优化后的设计算法在资源上节省了约 35%的逻辑单元，处理速度提高了约 20%。

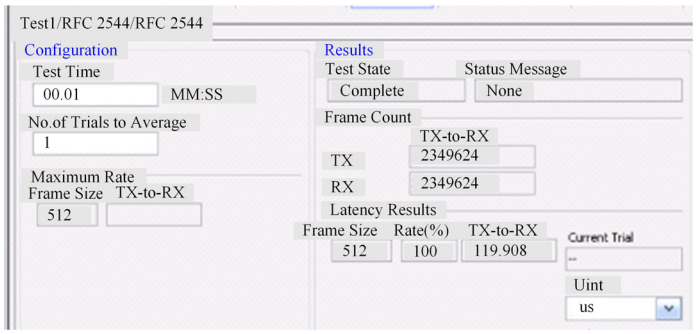
由于加密模块仅对 OPUk 净荷数据加密，不会对原信号频谱产生影响，因此，信号频谱图和星座图相位均保持不变。

为准确测量加密引入的时延效果，本文对 512 字节固定字长报文进行数据传输时延测试。未加密的数据传输时延结果如图 3（a）所示，时延为 119.908μs；加密后的数据传输时延结果如图 3（b）所示，时延为 120.113μs。两相比较，加密引入的时延仅有 208ns，优于业界普遍加密过程时延引入微秒级别。

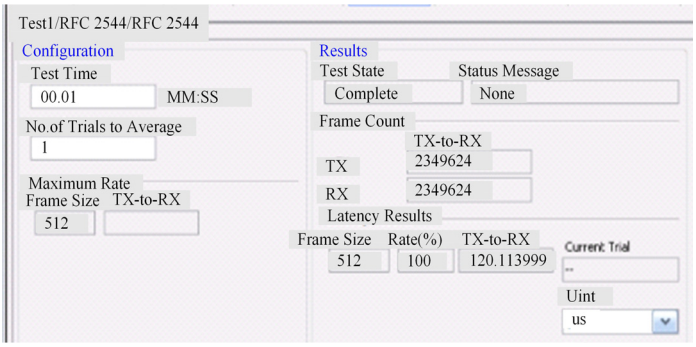
以上测试表明：AES-128 算法是一种适用于 OTN 系统的低时延加密算法，且数据加密后，系统传输性能依然良好。

5 结束语

本文对 AES 算法的原理以及优化的方法进行了研究，并将其应用于 OTN 系统中，同时利用 ITU-T G.709 协议的 RES 字节传送密钥。通过 AES 算法对连续比特流中 OPUk 净荷数据的加密，保证了数据传输的完整性和机密性。经实验测试，验证了优化后的 AES 算法相比于未优化的 AES 算法在资源的占用上少了近 1/3，运算速度上提高了约 1/5，具有低时延特性，达到了预期效果。



（a）加密前数据时延



（b）加密后数据时延

图 3 加密前后数据帧时延测试

参考文献：

[1] ITU-T. Interface for the Optical Transport Network: G.709/Y.1331 [S]. Geneva: ITU-T, 2016.

[2] ITU-T. Optical transport network (OTN): Linear protection, ITU-TRec G.873.1-2014[S]. Geneva: ITU-T, 2014.

[3] 威廉·斯托林斯. 密码编码学与网络安全：原理与实践[M]. 北京：电子工业出版社，2015：96-130.

[4] PENDLI V, PATHURI M, YANDRATHI S, et al. Improvising performance of advanced encryption standard algorithm [C]// Proc. Of the 2016 Second International Conference on Mobile and Secure Services (Mobi Sec Serv), February 26-27, Gainesville, USA. New York: IEEE, 2016.

[5] ITU-T. Characteristics of optical transport network hierarchy equipment functional blocks, ITU-T Rec G.798-2012[S]. Geneva: ITU-T, 2012.

[6] 王亮. 针对 AES 加密算法的研究及其 FPGA 实现 [D]. 上海：上海师范大学，2013.