AES 缓存碰撞攻击在美国太空安全挑战赛中的应用

雷思磊，仇 婕，马婷婷
（酒泉卫星发射中心，甘肃 兰州 732750）

摘 要： 通过对美国太空安全挑战赛中 AES 破解挑战的分析，提出了使用 AES 缓存碰撞时间攻击的破译思路，分析了 AES 缓存碰撞时间攻击的原理，将密钥可能空间缩小至 6 700 万，成功利用密钥已知部分推测出未知部分，最终实现该攻击，验证了 AES 缓存碰撞时间攻击的实践可行性。

关键词： AES；缓存碰撞；太空安全
中图分类号： TN915.08　　**文献标识码：** A　　**DOI：** 10.19358/j.issn.2096-5133.2022.03.006
引用格式： 雷思磊，仇婕，马婷婷. AES 缓存碰撞攻击在美国太空安全挑战赛中的应用[J].信息技术与网络安全，2022，41(3)：32-37.

Application of AES cache collision attack in American space security challenge

Lei Silei，Qiu Jie，Ma Tingting
（Jiuquan Satellite Launch Center，Lanzhou 732750，China）

Abstract： Through the analysis of the AES cracking challenge in the US space security challenge，this paper puts forward a decoding idea of using AES cache collision time attack，which reduces the possible space of the secret key to 67 million by analyzing the principle of AES cache collision time attack. This paper has successfully realized the attack by inferring the unknown part with the known part of the secret key，verifying the practical feasibility of the AES cache collision time attack.

Key words： AES；cache collision；space security

www.pcachina.com

0 引言

国家太空安全是国家安全在空间领域的表现。随着太空技术在政治、经济、军事、文化等各个领域的应用不断增加，太空已经成为国家赖以生存与发展的命脉之一，凝聚着巨大的国家利益，太空安全的重要性日益凸显。而在信息化时代，太空安全与信息安全紧密地结合在一起。2020 年 9 月 4 日，美国白宫发布了首份针对太空网络空间安全的指令——《航天政策第 5 号令》，其为美国首个关于卫星和相关系统网络安全的综合性政策，标志着美国对太空网络安全的重视程度达到新的高度[1]。随后美国组织了“黑一颗卫星(Hack A Sat)”[2]太空安全挑战赛，该赛事涵盖多个领域的知识，本文使用 AES 缓存碰撞攻击给出了该赛事其中一个挑战的解决方案。

1 美太空安全挑战赛介绍

从 2020 年起，美国已经连续两年举办太空安全挑战赛。该赛事分为资格赛、决赛两个阶段，资格赛采用积分制，其中积分前 8 名的参赛队进入决赛，最终角逐出前三名。以 2021 年为例，在 6 月 26 日至 27 日为期 30 小时的资格赛中，有 75 个以上国家近 3 000 名竞争对手组成的 1 000 多支队伍参加。资格赛由 5 个类别的 24 个挑战组成，侧重于不同的技能组合，包括卫星操作、逆向工程和射频通信等。12 月 11 日至 12 日决赛期间，8 个团队参加了持续 24 小时的决赛，决赛项目是攻防对抗式的夺旗赛，包括争夺虚拟地面站、通信子系统、“平卫星”物理卫星硬件，以及用于模拟和测试命令的数字孪生软件。各团队被要求同时操作和防御自己具有脆弱性的系统，同时攻击对手相同的系统来得分。系统中存在许多可利用的漏洞，团队必须修补或以其他方式减轻自己漏洞所产生的影响，以防止系统被攻击，同时保持系统正常运行[3]。

在2020年第一届太空安全挑战赛中，在“载荷模块(Payload Module)”这个类别下有一个挑战是“密码泄露(Leaky Crypto)”，主办方给出的挑战信息如下：

有一颗卫星在通信过程中，会对每个消息使用电码本(Electronic Codebook，ECB)模式的高级加密标准AES-128算法[4-6]进行加密。攻击者由于不知道密钥，无法正常加密，那么当攻击者向卫星发送消息后，卫星就会发现消息不对，从而丢弃该伪造消息。攻击者已经截获了一段密文，需要参赛者给出对应的明文，并且已知如下一些信息：

(1)知道密钥的前6个字节是“0xc35c7a9947fe”。

(2)知道截获密文对应明文的前几个字节是“flag{ksjkglhwsjd”。

(3)提供了100 000行尝试攻击数据，每个数据包括两部分：明文、加密明文时间。部分数据如下所示：

ed74cd51ab28e765d1e98965e86ba749,10584

ea0c914e1ff97edbe3bd46228f53771e,10656

3e8d4ed226d2ffea86fcf8be22af3e33,10704

3bd7f7bb63745fc45940220e417a116d,10632

99c5843576a344f0d8ec990795912a38,10632

34239b24eef47313a14e0a55767810dd,10632

7b9ae4bdb410862d12de1d94cd40853a,10656

f54c322e63f3b169fdfb4b161decc9b6,10680

91359a6820ecadafd2e616aa2c0baa42,10680

c4ce053a2df7f883d24905856df2b180,10656

从给出的挑战信息可知，这是已知加解密算法、部分明文、全部密文，并且有大量明文及其加密时间的数据，要求得出加密密钥的挑战。本文利用AES缓存碰撞攻击方法给出解决方案。

2 AES加密过程介绍

AES是美国联邦政府采用的一种区块加密标准，是一种对称加密算法[7-8]，对称加密即加密与解密用的是同样的密钥。AES因其运算速度快、安全性高的优点，被用来替代原先的数据加密标准(Data Encryption Standard，DES)，且已经被广泛使用[9-12]。AES加密、解密基本过程如图1所示。其中图1左边是加密过程，右边是解密过程，加密、解密是对称的。从图中可以发现，加密、解密需要进行多轮计算，密钥长度不同，导致计算轮数不同，根据密钥长度不同AES分为AES-128、AES-192[13]、AES-256[14]。对于AES-128，加密、解密过程有10轮运算。

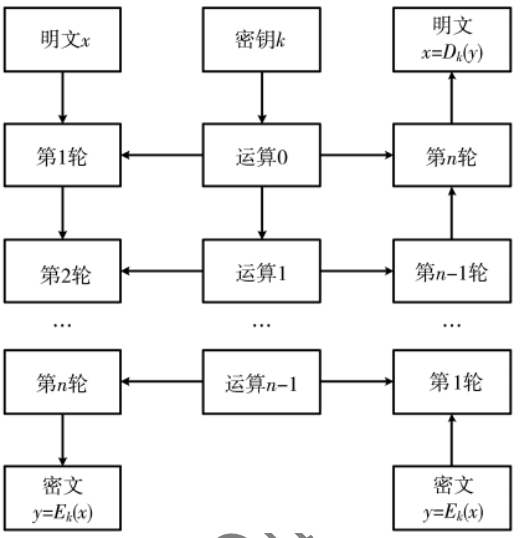

图1 AES加密、解密基本过程

其中每一轮运算的过程如图2所示。每一轮对16个字节进行运算，由四层组成：字节代换层(Byte Substitution Layer)、ShiftRows层、MixColumn层、密钥加法层(Key Addition Layer)。

AES在软件层面和硬件层面都已经有了很多种实现方式。其中AES设计者提议的一种快速软件实现——查表法得到广泛应用。查表法的核心思想是将字节代换层、ShiftRows层和MixColumn层融合为查找表，每个表的大小是256条，每一条是4 B，一般称为T盒(T-box)或T表。加密过程4个表(Te)，解密过程4个表(Td)，共8个表。每一轮操作都通过16次查表产生。虽然一轮就要经历16次查表，但这也简化了伽罗瓦域(Galois field)上的计算操作和矩阵乘法操作，而且T表一般提前编写程序算出，然后作为一个常量数组，硬编码在AES的软件实现代码中，对于计算机而言是非常简单高效的查表和按位运算。于是，每一轮的计算可以使用式(1)~式(4)表示。从式(1)~式(4)中可以发现，查找A_0、A_4、A_8、A_{12}的时候使用的都是Te_0表，查找A_5、A_9、A_{13}、A_1的时候使用的都是Te_1表，查找A_{10}、A_{14}、A_2、A_6的时候使用的都是Te_2表，查找A_{15}、A_3、A_7、A_{11}的时候使用的都是Te_3表。将使用同一个Te表的那4个字节称为一个家族(family)。

$$\begin{pmatrix} D_0 \\ D_1 \\ D_2 \\ D_3 \end{pmatrix} = \mathrm{Te}_0(A_0)+\mathrm{Te}_1(A_5)+\mathrm{Te}_2(A_{10})+\mathrm{Te}_3(A_{15})+W_{k0} \quad (1)$$

www.pcachina.com

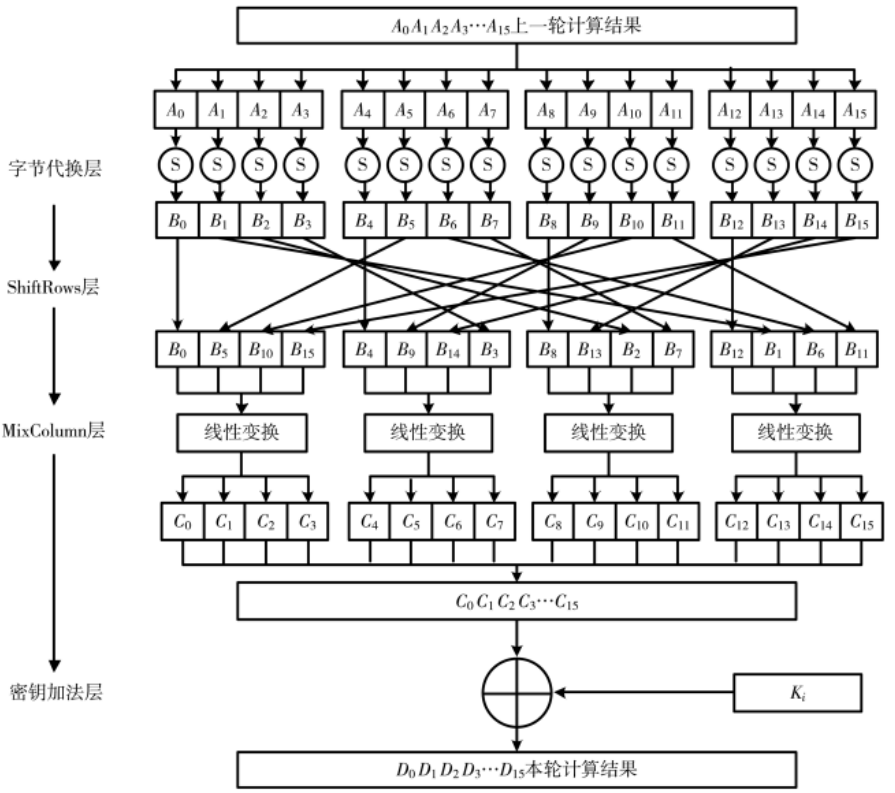

图 2 AES 每一轮的运算

$$\begin{pmatrix} D_4 \\ D_5 \\ D_6 \\ D_7 \end{pmatrix} = \mathrm{Te}_0(A_4)+\mathrm{Te}_1(A_9)+\mathrm{Te}_2(A_{14})+\mathrm{Te}_3(A_3)+W_{k1} \quad (2)$$

$$\begin{pmatrix} D_8 \\ D_9 \\ D_{10} \\ D_{11} \end{pmatrix} = \mathrm{Te}_0(A_8)+\mathrm{Te}_1(A_{13})+\mathrm{Te}_2(A_2)+\mathrm{Te}_3(A_7)+W_{k2} \quad (3)$$

$$\begin{pmatrix} D_{12} \\ D_{13} \\ D_{14} \\ D_{15} \end{pmatrix} = \mathrm{Te}_0(A_{12})+\mathrm{Te}_1(A_1)+\mathrm{Te}_2(A_6)+\mathrm{Te}_3(A_{11})+W_{k3} \quad (4)$$

3 AES 缓存碰撞攻击原理

假设明文为 P，大小为 16 B，第 i 个字节为 P_i，密钥为 K，大小也为 16 B，第 i 个字节为 K_i，对于第一轮计算，查找表的序号是 $P_i \oplus K_i$，如表 1 所示。假如 $P_0 \oplus K_0$ 等于 $P_4 \oplus K_4$，那么这两次查表将查找到的是同一个内存地址，根据处理器的工作原理，由于已经将 $P_0 \oplus K_0$ 对应的数据从内存读取到了缓存，因此再次读取 $P_4 \oplus K_4$ 对应的数据的时候，将直接从缓存读取对应数据，时间将大大节省。如果出现这种情况，那么由于已知 P_0、K_0、K_4，就可以计算得到对应的 P_4[15]。

表 1 第一轮查找表的序号

T 表	Te_0	Te_1	Te_2	Te_3
序号	$P_0 \oplus K_0$	$P_5 \oplus K_5$	$P_{10} \oplus K_{10}$	$P_{15} \oplus K_{15}$
	$P_4 \oplus K_4$	$P_9 \oplus K_9$	$P_{14} \oplus K_{14}$	$P_3 \oplus K_3$
	$P_8 \oplus K_8$	$P_{13} \oplus K_{13}$	$P_2 \oplus K_2$	$P_7 \oplus K_7$
	$P_{12} \oplus K_{12}$	$P_1 \oplus K_1$	$P_6 \oplus K_6$	$P_{11} \oplus K_{11}$

对于小样本量数据，还需考虑控制其他 family 中序号相等对加密时间的影响。例如，一条明文数据的加密中可能因为 $P_0 \oplus K_0 = P_4 \oplus K_4$，从而使得加密时间大大减少，但每组加密数据有 16 B，若其他 family 中不存在这种两个序号相等的情况，那么这条数据的加密时间不会因为一个 $P_0 \oplus K_0 = P_4 \oplus K_4$ 而减少太多；相反若一组数据中虽然 $P_0 \oplus K_0 ! = P_4 \oplus K_4$，但在其他 family 中如果存在多个序号相等的情况，例如在 Te_1 中 $P_5 \oplus K_5 = P_9 \oplus K_9 \cdots\cdots$ 则这组数据的加密时长也会大大缩短。但是对于大样本量数据，时间差计算的统计平均值，可以有效控制错误判断。比如：将 100 000 组数据，随机分成两组，其统计平

www.pcachina.com

均时间应该是一致的，或者说相差很小，但是，通过条件筛选出特定的一组（此处就是依据$P_0 \oplus K_0 = P_4 \oplus K_4$），那么针对第 4 字节就不再具有统计特性，但是对其他字节而言仍然具有统计特性。因此，对于挑战赛题中 100 000 组大样本数据不必再讨论其他 family 中序号相等对加密时间的影响。

从挑战赛题目可知，目前有100 000 组明文数据，其中包含每一组明文数据的加密时间，并且已知密钥的前 6 个字节，根据上述原理，尝试使用密钥的第 0 字节，推测密钥的第 4 字节，以验证上述分析是否成立。验证步骤如下：

（1）对 100 000 组数据，每一组数据的第 0 字节与已知密钥的第 0 字节进行异或运算，得到结果数组 Set_0；

（2）假设密钥第 4 个字节为 x，那么对100 000 组数据，每一组数据的第 4 个字节与 x 进行异或运算，得到结果数组 Set_1；

（3）对比 Set_0 和 Set_1，可以得到其中相同的元素的序号集，这个序号集称为碰撞数据集 CollisionSet，不相同的元素的序号集称为非碰撞数据集 non－CollosionSet；

（4）将测试数据中，non－CollosionSet 中对应明文的加密时间取均值 non－CollosionSetMeanTime，将测试数据中，CollisionSet 中对应明文的加密时间取均值 CollosionSetMeanTime；

（5）将 non－CollosionSetMeanTime 减去 CollosionSetMeanTime，得到一个时间差；

（6）对于密钥第 4 个字节的所有 256 种可能值，重复第（2）~（6）步，得到 256 个时间差；

（7）取出时间差绝对值最大的前 10 个对应的猜测值，其中应有密钥第 4 个字节对应的值。

按照上述步骤，编写程序，得到时间差的情况如图 3 所示，其中横轴是密钥第 4 字节的 0~255 可能的值，纵轴是计算出来的时间差，可见有一个明显的峰值，表示时间差很大。时间差最大的前 10 个可能值如下：

70　69　68　71　187　133　179　105　96　180

从挑战题目给出的已知密钥可知，密钥的第 4 字节是 0x47（十进制为 71），正是在上述推测出来最有可能的 10 个值中，因此，可以证明 AES 缓存碰撞攻击可以用于本挑战的解答。

4 AES 缓存碰撞攻击实现

将上述推测密钥第 4 字节的过程称为密钥推测函数 KeySearch，其函数定义原型如下：

```
def keysearch(l_ref, targets):
```

其中，l_ref 是已知密钥的某个字节，由该字节推测与这个字节在一个 family 中的其余字节。已知密钥的前 6 个字节，所以可由密钥第 0 字节推测第 4、8、12 字节，由密钥第 1 字节推测密钥第 5、9、13 字节，由密钥第 2 字节推测密钥第 6、10、14 字节，由密钥第 3 字节推测密钥第 7、11、15 字节；由密钥第 4 字节推测密钥第 8、12 字节；由密钥第 5 字节推测密钥第 9、13 字节。

```
keysearch(l_ref_0, [4,8,12])
```

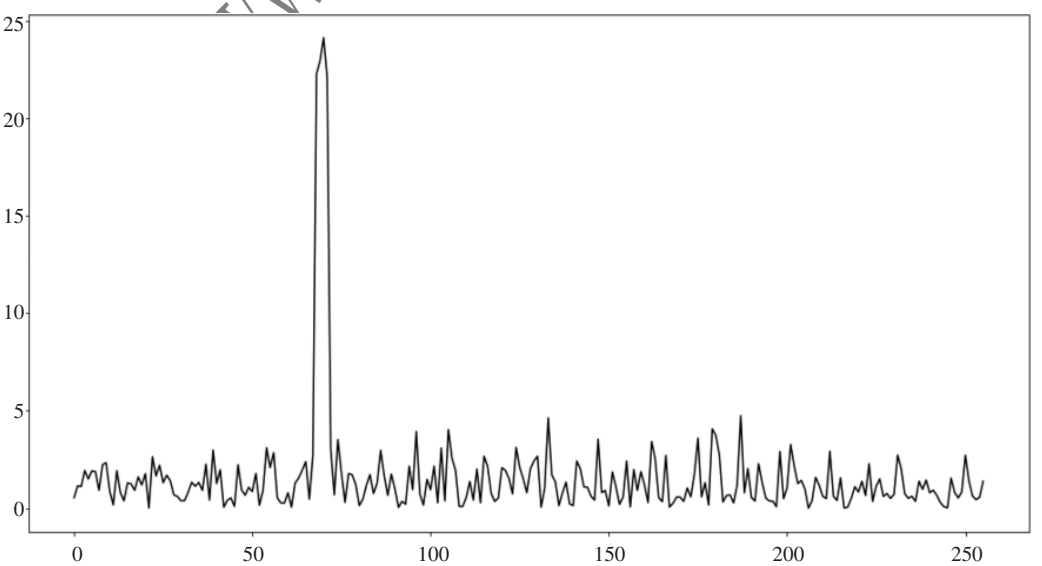

图 3　当密钥第 4 个字节依次取 0~255 时，分别对应的加密时间差

www.pcachina.com

```
keysearch(l_ref_4, [0,8,12])
keysearch(l_ref_1, [5,9,13])
keysearch(l_ref_5, [1,9,13])
keysearch(l_ref_2, [6,10,14])
keysearch(l_ref_3, [7,11,15])
```

第一轮推测结果如下，此处只取了前四个最有可能的结果。

```
Found candidate values for key byte 8
[38 37 36 39]
Found candidate values for key byte 9
[235 234 233 232]
Found candidate values for key byte 14
[31 29 28 30]
Found candidate values for key byte 7
[31 30 29 28]
Found candidate values for key byte 15
[155 154 152 153]
```

可知，推测出了密钥第 7、8、9、14、15 字节的可能值。还有密钥第 6、10、11、12、13 字节没有推测出来。继续使用推测出来的新的字节推测未知字节，如下：

```
keysearch(l_ref_7, [11])
keysearch(l_ref_8, [12])
keysearch(l_ref_9, [13])
keysearch(l_ref_14, [6,10])
keysearch(l_ref_15, [11])
```

第二轮推测结果如下，此处只取了前四个最有可能的结果。

```
Found candidate values for key byte 13
[220 221 223 222]
Found candidate values for key byte 10
[50 48 51 49]
```

可知，新推测出了密钥的第 10、13 字节的可能值。还有密钥第 6、11、12 字节没有推测出来。继续使用推测出来的新的字节推测未知字节，如下：

```
keysearch(l_ref_10, [2,6,14])
```

第三轮推测结果如下，此处只取了前四个最有可能的结果。

```
Found candidate values for key byte 14
[31 29 30 28] 24.61651797495142
```

可知，此次只是推测出了密钥第 14 字节的可能值，而这个结果前文已经得到了，所以密钥第 6、11、12 仍然还没有推测得到。这里将 family 的范围扩大，用密钥的任一字节使用 keySearch 推测其余 15 个字节的值。

```
keysearch(l_ref_0, range(16))
keysearch(l_ref_1, range(16))
keysearch(l_ref_2, range(16))
keysearch(l_ref_3, range(16))
keysearch(l_ref_4, range(16))
keysearch(l_ref_5, range(16))
keysearch(l_ref_7, range(16))
keysearch(l_ref_8, range(16))
keysearch(l_ref_9, range(16))
keysearch(l_ref_10, range(16))
keysearch(l_ref_13, range(16))
keysearch(l_ref_14, range(16))
keysearch(l_ref_15, range(16))
```

第四轮推测结果如下，此处只取了前四个最有可能的结果。

```
Found candidate values for key byte 12
[150 148 149 151] 25.77075236406381
Found candidate values for key byte 6
[13 14 15 12] 25.987920521583874
```

可知，新推测出了密钥第 6、12 字节的值，现在只剩下第 11 字节还没有推测出结果。但是整个解空间大小减少为 $4^9 \cdot 256 = 67\ 108\ 864$，因为已知部分明文，就可以使用暴力破解法，通过穷举所有的可能性对挑战赛给出的密文进行解密，对比已知的部分明文，最终确定正确的密钥。本文使用 Intel i7-10750H 处理器、32 GB 内存的计算机最终暴力破解使用时间为 9 min 21 s。

5 结论

随着太空安全得到日益广泛的重视，美国通过举办太空安全挑战赛，有效促进了太空安全薄弱环节的普及，展示了从地面站、通信链路、载荷到飞行器可能面临的各种攻击，选拔了太空安全领域的技术专家。本文运用 AES 缓存碰撞时间攻击解决了太空安全挑战赛中关于 AES 密码破译的挑战，对于全面理解美国太空安全挑战赛具有一定参考意义。

参考文献

[1] 苟子奕，韩春阳.美国太空领域的网络安全政策分析[J].国际太空，2021(1)：27-31.

[2] Hack a sat[EB/OL].https://www.hackasat.com/.

[3] 星际智汇.美空军部宣布“黑一颗卫星”挑战赛冠军[EB/OL].(2021-12-16).https://mp.weixin.qq.com/

www.pcachina.com

s/k93yA9-iUB5VB9hi-Ao-tA.

[4] BAHRAK B, AREF M R. Impossible differential attack on seven-round AES-128[J]. IET Information Security, 2008, 2(2): 28-32.

[5] LIBERATORI M, OTERO F, BONADERO J C, et al. AES-128 cipher. high speed, low cost FPGA implementation[C]//2007 3RD Southern Conference on Programmable Logic. IEEE, 2007: 195-198.

[6] ALMAZROOIE M, SAMSUDIN A, ABDULLAH R, et al. Quantum reversible circuit of AES-128[J]. Quantum Information Processing, 2018, 17(5): 1-30.

[7] 吕慧, 赵跃华. 基于掩码的AES算法抗二阶DPA攻击方法研究[J]. 电子设计工程, 2015, 23(20): 31-33.

[8] 黄灿英, 汪莹, 陈艳. 基于AES-128密码算法的嵌入式系统安全[J]. 沈阳工业大学学报, 2020, 42(5): 549-553.

[9] 卢军, 张国辉, 李国强. 一种基于数据分解的AES优化算法设计[J]. 单片机与嵌入式系统应用, 2019, 19(4): 22-25.

[10] 万刘蝉, 韦永壮. AES-128的密钥中比特检测及分析[J]. 桂林电子科技大学学报, 2016, 36(4): 338-341.

[11] TSAI K L, HUANG Y L, LEU F Y, et al. AES-128 based secure low power communication for LoRaWAN IoT environments[J]. IEEE Access, 2018, 6: 51-66.

[12] MAHAJAN P, SACHDEVA A. A study of encryption algorithms AES, DES and RSA for security[J]. Global Journal of Computer Science and Technology, 2013.

[13] HAN L, WU N, GE F, et al. Differential fault attack for the iterative operation of AES-192 key expansion[C]//2020 IEEE 20th International Conference on Communication Technology (ICCT). IEEE, 2020: 1156-1160.

[14] BIRYUKOV A, DUNKELMAN O, KELLER N, et al. Key recovery attacks of practical complexity on AES-256 variants with up to 10 rounds[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, 2010: 299-319.

[15] BONNEAU J, MIRONOV I. Cache-collision timing attacks against AES[C]//International Workshop on Cryptographic Hardware and Embedded Systems. Springer, Berlin, Heidelberg, 2006: 201-215.

(收稿日期: 2021-12-26)

作者简介:

雷思磊(1984-), 男, 硕士, 高级工程师, 主要研究方向: 信息安全。

仇婕(1990-), 女, 硕士, 工程师, 主要研究方向: 网络管理。

马婷婷(1990-), 女, 本科, 工程师, 主要研究方向: 光纤通信。

www.pcachina.com

《信息技术与网络安全》被美国EBSCO数据库收录

本刊收到美国《艾博思科(EBSCO)数据库》颁发的收录证书, 从2021年第(40)6期开始正式被EBSCO数据库全文收录。

此次成功收录是对我刊学术质量的认可, 通过国际国内双向学术交流, 增加了中文论文的国际展示度, 提升了国内作者及机构的国际学术地位与知名度, 也符合学术平台发展客观规律, 标志着期刊发展进入了一个新阶段!


EBSCO
INFORMATION SERVICES
CERTIFICATE OF INCLUSION 收录证书
This acknowledges that
Information Techology and Network Security
has been licensed for indexing and inclusion in EBSCO research database from 2021 no. 40(6) onwards.
兹证明
信息技术与网络安全
从2021年第40(6)期被收录于EBSCO学术数据库。
EBSCO INTERNATIONAL CHINA CO., LTD.
艾博思科印刷出版咨询(北京)有限公司
Valid through 有效日期: Dec 31, 2024
Ref. no. 备查号: MID-MDsJ

版权声明

经作者授权，本论文版权和信息网络传播权归属于《信息技术与网络安全》杂志，凡未经本刊书面同意任何机构、组织和个人不得擅自复印、汇编、翻译和进行信息网络传播。未经本刊书面同意，禁止一切互联网论文资源平台非法上传、收录本论文。

截至目前，本论文已经授权被中国期刊全文数据库（CNKI）、万方数据知识服务平台、中文科技期刊数据库（维普网）、JST 日本科技技术振兴机构数据库等数据库全文收录。

对于违反上述禁止行为并违法使用本论文的机构、组织和个人，本刊将采取一切必要法律行动来维护正当权益。

特此声明！

《信息技术与网络安全》编辑部

中国电子信息产业集团有限公司第六研究所

www.ncachina.com