计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2019.20180745
56(7):1488-1497,2019

面向数字型的轻量级保形加密算法研究

刘波涛[1,2,3] 彭长根[1,2,3] 吴睿雪[1,3] 丁红发[3,4] 谢明明[3]
[1](贵州大学计算机科学与技术学院 贵阳 550025)
[2](中国电子科技集团大数据研究院有限公司 贵阳 550081)
[3](公共大数据国家重点实验室(贵州大学) 贵阳 550025)
[4](贵州财经大学信息学院 贵阳 550025)
(teslal0505@foxmail.com)

Lightweight Format-Preserving Encryption Algorithm Oriented to Number

Liu Botao[1,2,3], Peng Changgen[1,2,3], Wu Ruixue[1,3], Ding Hongfa[3,4], and Xie Mingming[3]
[1](*College of Computer Science and Technology, Guizhou University, Guiyang* 550025)
[2](*Big Data Research Institute Co., Ltd., China Electronic Technology Group Corporation Ltd., Guiyang* 550081)
[3](*State Key Laboratory of Public Big Data (Guizhou University), Guiyang* 550025)
[4](*College of Information, Guizhou University of Finance and Economics, Guiyang* 550025)

Abstract The Internet of things (IoT), which has been widespread and large-scale applied, arises more and more security and privacy issues. Lightweight encryption is an important measurement for ensuring confidentiality for devices of IoT, in where the computing, storage and energy resources are always limited. However, the shallow application of lightweight block cipher will change the format of ciphertext tremendously due to confusion and diffusion operations. These changes make the ciphertext inconsistent with plaintext in expressive form and format, and lead to require extra storage, computation and redisplay resources. Lightweight format-preserving encryption algorithm can ensure data confidentiality while maintaining the format consistency between ciphertext and plaintext, and these features benefit to IoT greatly. Aiming at the problems that the traditional format-preserving encryption algorithm performs inefficiently, consumes many resources, and cannot encrypt length numeric data, a lightweight format-preserving encryption algorithm oriented to number is proposed in this work. Firstly, a numeric typed permutation table is constructed by using lightweight block cipher algorithm; then the numerical plaintext is added to the key of lightweight block cipher in one-to-one correspondence, and the modulo 10 operation is performed; at last, replacement cryptographic operation is performed to obtain the numerical ciphertext by using the proposed the numeric typed replacement table. The proposed algorithm preserves the format for any numerical data with arbitrary

收稿日期:2018-11-05;**修回日期**:2019-02-18
基金项目:国家自然科学基金项目(U1836205,61662009,61772008);贵州省科技计划项目(黔科合重大专项字[2018]3001,黔科合重大专项字[2018]3007,黔科合重大专项字[2017]3002,黔科合基础[2017]1045,黔科合支撑[2016]2315);衡阳师范学院智能信息处理与应用湖南省重点实验室开放基金项目(IIPA18K02)
This work was supported by the National Natural Science Foundation of China (U1836205, 61662009, 61772008), the Science and Technology Foundation of Guizhou Province (Guizhou-Science-Contact-Major-Program [2018] 3001, Guizhou-Science-Contact-Major-Program [2018] 3007, Guizhou-Science-Contact-Major-Program [2017] 3002, Guizhou-Science-Contact-Foundation [2017] 1045, Guizhou-Science-Contact-Support [2016] 2315), and the Open Fund Project of Hunan Provincial Key Laboratory of Intelligent Information Processing and Application for Hengyang Normal University (IIPA18K02).
通信作者:彭长根(peng_stud@163.com)

length, and it's also consistent with the original lightweight block cipher in terms of efficiency and security. By comparing with traditional format-preserving encryption, the experimental result shows that the proposed algorithm is more security, more efficient and more lowly lower resource-consuming. It is suitable for secure storage and data marking of numerical data in resource-constrained environment devices of IoT.

Key words lightweight block cipher; format-preserving encryption; numeric data; Internet of things (IoT); data marking

摘 要 物联网的大规模普及应用引发了诸多安全和隐私问题.轻量级加密是资源受限环境下物联网设备保证数据机密性的主要手段,然而直接应用轻量级分组密码加密会因为编码序列的混淆扩散使密文格式发生巨大变化,在表现形式和格式上与明文不一致,需要额外的存储、计算、回显资源.轻量级保形加密算法可以在实现机密性的同时,保持密文数据与明文数据在格式上具有一致性,在物联网领域具有更大的优势.针对现有保形加密算法存在实现效率不高、资源消耗较大及不能加密较长数字型数据的问题,提出一种面向数字型的轻量级保形加密算法.首先利用轻量级分组密码算法构造数字型置换表,数字型明文与轻量级分组密码的加密密钥进行一一对应相加、取模 10 操作,再利用数字型置换表进行置换加密操作,得到数字型密文数据.算法实现了对任何长度数字型数据加密前后的格式不改变,分析表明该算法在效率、安全性方面与原轻量级分组密码算法保持一致.同时,实验结果表明:相比传统的保形加密算法,该算法具有高安全、高效、低资源,适用于资源受限环境下物联网设备的数据加密存储及数据遮蔽.

关键词 轻量级分组密码;保形加密;数字型数据;物联网;数据遮蔽

中图法分类号 TP309

信息化时代,数字数据在生活当中是无处不在,例如身份证号、手机号、信用卡号及客户号等都是数字重要的应用,而这些数据是包含了个人身份识别信息以及商业财产机密等,为了防止存储在物联网终端设备中这些有价值的数字数据被恶意攻击者窃取利用,是需要对其进行加密处理来实现防护.

由于物联网终端设备自身存在计算能力弱及存储空间非常有限的特点[1],而终端设备的数字型数据利用轻量级分组密码(例如 PRESENT[2] 密码、SKINNY[3] 密码、SFN[4] 密码等)直接加密保护,得到加密后的密文数据类型与长度发生变化,这些数据的结构改变导致数据无法存回终端设备,从而造成额外的成本开销与损失.为了解决该问题,采用保形加密(format-preserving encryption, FPE)进行对数据加密保护,加密后的密文与明文具有相同的数据格式,从而不需要破坏终端设备数据存储结构[5].

保形加密是一类新型、特殊的加密技术,广泛应用于数据遮蔽领域,其相关的研究一直备受密码学者的关注.从 1981 年就开始进行探索研究[6],到 2002 年,由 Black 和 Rogaway[7] 提出 FPE 的基本模型与方法,并且也是首次给出了其安全性,所提出其中包括 3 种传统型的算法:Prefix 型算法,Cycle-Walking 型算法和 Generalized-Feistel 型算法,其在一定程度上解决了整数上的 FPE 问题.随后,Bellare 等人[8] 基于 Cycle-Walking 型算法进行了拓展,提出了一种基于非平衡 Feistel 网络的整数 FPE 算法;Jia 等人[9] 基于 Generalized-Feistel 型算法,提出了一种基于 Type-2 Feistel 网络的整数 FPE 算法;Bellare 等人[10] 在 CCS 2017 上提出了基于身份的保型加密算法.对于保形加密算法安全性分析而言,Bellare 等人[11] 在 CCS 2016 上提出基于 Feistel 网络的保形加密算法的消息恢复攻击,Durak 等人[12] 在 CRYPTO 2017 上针对 FF3 保形加密算法进行攻击;Hoang 等人[13] 在 CRYPTO 2018 上提出基于 Feistel 网络的保形加密算法的已知明文消息恢复攻击;在实际应用中,Zou 等人[14] 实现将 Prefix 型保形加密算法进行加密保护中文名字等.

现有的保形加密算法中存在实现效率不高、资源较大的问题,而传统的 Prefix 型保形加密算法相比其他传统的 Cycle-Walking 型和 Generalized-Feistel 型保形加密算法操作过程简单,对数据加解

密操作速度较快,是一个良好的保形加密算法.但是传统的 Prefix 型保形算法也有 2 方面的不足:1)利用传统的高级加密标准(advanced encryption standard, AES)[15]分组密码算法作为置换表的构建算法,从而造成置换表的建立会耗费大量时间与软硬件资源,无法适用于物联网终端设备.2)主要是进行实现整型字段的加密处理,从而无法很好地实现数字类型数据的加密;且在实现整型字段数据时,该方法也只能处理消息空间整数集小的数据,对常用的信用卡号、电话号码、身份证号及各种验证码等较长的数字型数据不能够很好保形加密处理.并且国内外尚未发现公开发表有关物联网设备终端数据保形加密算法研究成果.

本文提出一种面向数字型的轻量级保形加密算法,先通过轻量级分组密码算法构造数字型置换表,要加密数字型明文与轻量级分组密码的加密密钥进行一一对应相加、取模 10 操作,再进行数字型置换表置换加密操作,得到数字型密文数据.该算法做到了保持加密前后的数据格式不改变,实现了数据遮蔽,并且对任何长度的数字型数据,进行高效、安全的加密保护,节省了软件实现的开销与硬件实现的成本,对于保护物联网终端设备数据存储的安全,无疑具有重要意义.

1 PRESENT 密码

在 2007 年 PRESENT 轻量级分组密码是由国际加密硬件与嵌入式系统国际会议上被提出的,成为了 IOS-29192 标准.在轻量级分组密码中,PRESENT 密码的提出是一个里程碑性研究,该密码高效的、低资源设计优点已被其后很多轻量级分组密码所模仿与采用,例如 GIFT 密码[16]与 RECTANGLE 密码[17],其 S 盒有良好的密码特性,LED[18]与 QTL[19]等多个密码采用该 S 盒.

PRESENT 密码的分组长度是 64 b,密钥长度有 80 b 和 128 b 两种,加密变换需要 31 轮迭代运算,采用 SP(substitution-permutation)网络结构,包含密钥扩展函数(*keySchedule*)与加密轮函数,轮函数是轮密钥加变换(*addRoundKey*)、S 盒变换(*sBoxLayer*)及 P 置换(*pLayer*) 3 个模块构成,如图 1 所示(⊕为异或符).

轻量级 PRESENT 密码描述为:设 64 b 中间变量为 *state*,80 b 和 128 b 初始密钥为 k,64 b 轮密钥为 rk.加密轮函数的 3 个模块具体描述:

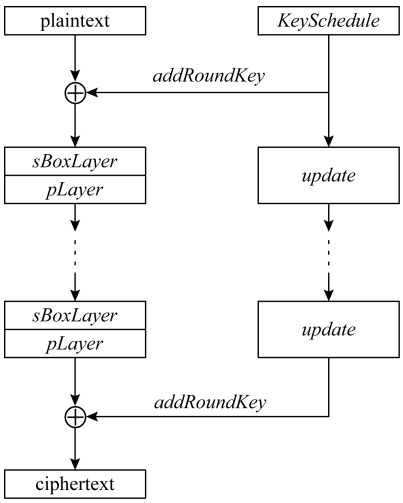


Fig. 1 The encryption process of PRESENT

图 1 PRESENT 加密流程图

1) 轮密钥加变换

轮密钥加变换是 *state* 与 rk 一一异或运算操作:

$$F_2^{64}\rightarrow F_2^{64}, state\leftarrow state\oplus rk^i, (1\leqslant i\leqslant 32). \quad (1)$$

2) S 盒变换

S 盒变换是将 *state* 划分为 16 个 4 b 数据,每 4 b 数据经过一个 4×4 的 S 盒替换运算.

$$F_2^4\rightarrow F_2^4, state_{j[3:0]}\leftarrow \mathrm{S}(state_{j[3:0]}), \quad (0\leqslant j\leqslant 15). \quad (2)$$

3) P 置换

P 置换变换是 *state* 进行每 1 b 移位运算操作,置换层的各比特通过公式置换(b 代表每 1 b),该操作在硬件实现过程不需要消耗资源:

$$\begin{aligned} b_j &\leftarrow b_{4\times j},\\ b_{j+16} &\leftarrow b_{4\times j+1},\\ b_{j+32} &\leftarrow b_{4\times j+2},\\ b_{j+48} &\leftarrow b_{4\times j+3}, (0\leqslant j\leqslant 15). \end{aligned} \quad (3)$$

轮函数结构如图 2 所示.

密钥扩展函数具体描述为(以 80 b 密钥为例):

1) 循环移位变换

循环移位变换是 k 进行向左循环移 61 b:

$$k\leftarrow k<<<61. \quad (4)$$

2) 常数加变换

常数加变换是 k 进行异或一个 5 b 轮常数(*round_count*):

$$k\leftarrow k\oplus round\_count. \quad (5)$$

密钥扩展函数中的 S 盒与加密轮函数共用一个 S 盒,而 S 盒替换只对 80 b 密钥中 4 b 进行一次替换.

轻量级 PRESENT 密码自被提出来,密码学者一直对其进行一系列的安全分析,最新的 PRESENT

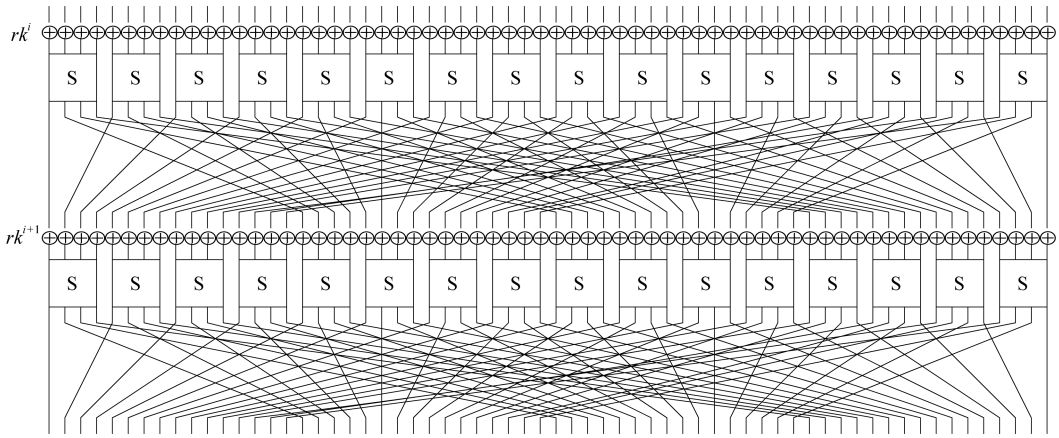


Fig. 2 The round function structure of PRESENT

图 2 PRESENT 密码的轮函数结构

密码分析结果如表 1 所示,经过 10 年的不断的考验,PRESENT 密码是满足当前安全应用的需求.

Table 1 Results of Cryptanalysis of PRESENT

表 1 PRESENT 密码的攻击分析

Attack Types	Rounds	Time	Data
MLC[20]	26	2^{72}	2^{66}
TDC[21]	26	2^{76}	$2^{63.16}$
FFT-MLC[22]	27	2^{74}	2^{62}
BC[23]	31	$2^{79.34}$	2^{22}

表 1 中攻击类型:多维线性攻击(multidimensional linear cryptanalysis, MLC)、截断差分攻击(truncated differential cryptanalysis, TDC)、FFT-多维线性攻击(fast Fourier transform multidimensional linear cryptanalysis, FFT-MLC)、Biclique 攻击(Biclique cryptanalysis, BC).

2 构造面向数字型的轻量级保形加密算法

在构造新的保形加密算法中,效率和安全性永远是值得探讨和研究的,其保证算法的安全性条件下做到构建一个高效、低资源的保形加密算法.由于传统的 Prefix 型保形加密算法相比其他 Cycle-Walking 型和 Generalized-Feistel 型保形加密算法更具有简洁、高效的特点,本文将改进于传统的保形 Prefix 型算法来实现一种高效、低资源的面向数字型保形加密算法.

对于数字型的数据,明文消息空间数据集为 $X=0,1,\cdots,9$,为了保持密文消息数据集与明文消息数据集相同,算法中构建的置换表的数据集也为 $X=0,1,\cdots,9$.一种高效、低资源的面向数字型保形加密算法的构造分为 6 个步骤进行实现:

1) 利用轻量级分组密码加密

将 0~9 这 10 个数字加载至寄存器,利用轻量级分组密码对这 10 个数字进行加密 E_P 操作,轻量级分组密码密钥为 k,得到元组 I:

$$I=(E_P(0,k),E_P(1,k),\cdots,E_P(9,k)). \quad (6)$$

其中,每一个分量元组 $I_j=(E_P(j,k))(0\leqslant j\leqslant 9)$,这些分量元组长度与轻量级分组密码算法的分组长度相同,一般为 64 b.

2) 元组数据排序

将步骤 1 中每一个分量元组 $I_j=(E_P(j,k))$进行二进制数从高位到低位大小判断排序,得到一个数字型置换表序列 T.

3) 明文相加、取模 10

将明文 M 加载至寄存器中,M 与轻量级分组密码的加密密钥进行一一对应相加、取模 10 操作:

$$(M+k) \bmod 10. \quad (7)$$

4) 明文加密置换

将步骤 3 运算的结果数据,进行步骤 2 得到的置换表置换操作,得到密文 C,这个置换加密为 E_T,保形加密具体流程如图 3 所示.

$$C=E_T((M+k) \bmod 10). \quad (8)$$

5) 构造逆置换表

在正置换表的基础上,构造一个逆的置换表用于解密密文,该逆置换表也是一个数字型置换表序列 T^{-1}.

6) 密文解密置换

将 C 先进行逆置换表置换操作,然后与轻量级分组密码密钥进行一一对应相减取模 10 运算,得到 M,具体流程如图 4 所示.

$$M=(E_{T^{-1}}^{-1}(C)-k) \bmod 10. \quad (9)$$

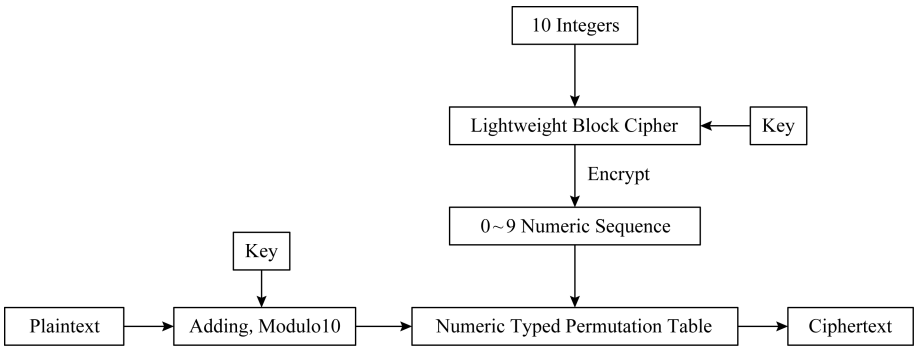


Fig. 3 The encryption process of lightweight format-preserving algorithm

图 3 轻量级保形算法加密流程

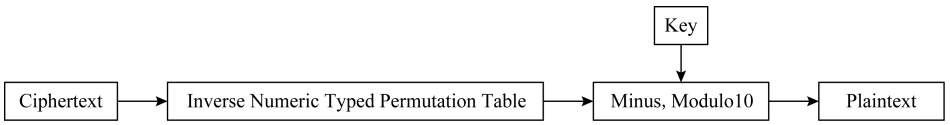


Fig. 4 The decryption process of lightweight format-preserving algorithm

图 4 轻量级保形算法解密流程

3 面向数字型的轻量级保形加密算法实现

一种高效、低资源的面向数字型保形加密算法的具体实现,构造置换表的轻量级分组密码是采用第 1 节描述的 PRESENT 密码.

首先将十进制(decimal, DEC)的 0～9 这 10 个整数用二进制 4 b 数据表示,并加载至寄存器,在寄存器中分别进行高位填补 60 个二进制数 0,从而得到 10 个 64 b 数据,作为构造数字型置换表的待加密数据,如表 2 所示.

将填补后这 10 个 64 b 二进制数作为 PRESENT 密码算法的 10 个待加密数据,选取 PRESENT 密码的加密密钥,选择为 PRESENT 密码的 80 b 密钥长度,此次加密密钥任选为十六进制(hexadecimal, HEX)的 642A032F5010040760CB.将这 10 个待加密数据用这个加密密钥分别进行 PRESENT 密码加密操作,得到 10 个密文数据,将密文数据进行对应 0～9 数字大小的分布标记,这个分布序号标记列表作为一个数字型正置换表,如表 3 数据变换过程所示.

Table 2 The Supplement of Integer

表 2 整数填补数据表

Integer(DEC)	64 b Number(HEX)
0	0000 0000 0000 0000
1	0000 0000 0000 0001
2	0000 0000 0000 0002
3	0000 0000 0000 0003
4	0000 0000 0000 0004
5	0000 0000 0000 0005
6	0000 0000 0000 0006
7	0000 0000 0000 0007
8	0000 0000 0000 0008
9	0000 0000 0000 0009

Table 3 The Construction of Numeric Typed Permutation Table

表 3 数字型正置换表构造

Plaintext(HEX)	Key(HEX)	Ciphertext(HEX)	Sequence(DEC)
0000 0000 0000 0000	642A 032F 5010 0407 60CB	4663 2034 C503 F4C1	1
0000 0000 0000 0001	642A 032F 5010 0407 60CB	D8CA BDD2 B86A E310	8
0000 0000 0000 0002	642A 032F 5010 0407 60CB	AE9E 5605 AA44 80AE	7
0000 0000 0000 0003	642A 032F 5010 0407 60CB	8E8F 9E3A EADE 0A11	6
0000 0000 0000 0004	642A 032F 5010 0407 60CB	55D3 E7AA E8F8 8142	2
0000 0000 0000 0005	642A 032F 5010 0407 60CB	6E75 681E 2B58 702C	4
0000 0000 0000 0006	642A 032F 5010 0407 60CB	F55D FD45 946D 1025	9
0000 0000 0000 0007	642A 032F 5010 0407 60CB	3372 31A1 3183 1EAF	0
0000 0000 0000 0008	642A 032F 5010 0407 60CB	56DF B9EF A8D0 3B92	3
0000 0000 0000 0009	642A 032F 5010 0407 60CB	72A7 1460 DF66 018E	5

从表 3 得到分布标记“1876249035”10 个数字的序列,作为数字型正置换表 T,如表 4 所示:

Table 4 The Numeric Typed Permutation Table

表 4 数字型正置换表

x	$T(x)$	x	$T(x)$
0	1	5	4
1	8	6	9
2	7	7	0
3	6	8	3
4	2	9	5

将要加密数字型明文数据加载至寄存器,在这里选取银行卡号作为要加密数字型明文数据,银行卡号为 6212262402033357571.在应用设备当中,将要加密的真实银行卡号数字数据与 PRESENT 轻量级分组密码的加密密钥(642A032F5010040760CB)进行一一对应相加、取模 10 运算操作,具体为

(6+6)%10=2,(2+4)%10=6,(1+2)%10=3,
(2+10)%10=2,(2+0)%10=2,(6+3)%10=9,
(2+2)%10=4,(4+15)%10=9,(0+5)%10=5,
(2+0)%10=2,(0+1)%10=1,(3+0)%10=3,
(3+0)%10=3,(3+4)%10=7,(5+0)%10=5,
(7+7)%10=4,(5+6)%10=1,(7+0)%10=7,
(1+12)%10=3.

当加密密钥比要加密的真实数字型明文数据长,将取加密密钥的对应前面部分数据;当加密密钥比要加密的真实数字型明文数据短,首先选择密码算法那种较长的密钥,如果那种较长的密钥还是比加密的真实数字型明文数据短,则选择较长的密钥前后部分异或操作进行扩充数据,从而扩充到相加一一对应的密钥长度,在实例当中,80 b 加密密钥比要加密的真实银行卡数字数据长,进行舍弃最后一个为“11”数.

将一一对应相加、取模运算得到数字型数据为 2632294952133754173,再一一进行数字型正置换表置换加密操作,例如数字“2”通过数字型正置换表置换为数字“7”,数字“6”通过数字型正置换表置换为数字“9”,依次进行全部置换操作,置换后,为了保证算法的安全,将数字型正置换表进行删除操作,置换加密后得到,数字型密文数据为 7967752547866042806,该密文数据与真实的银行卡号具有相同的数据类型,这样完成了银行卡号数据的加密保护.

进行加密后的数字型密文数据解密恢复操作,在构造数字型正置换表序列的基础上来构造逆置换表序列,将数字型正置换表序号进行取反、重排操作,得到数字型逆置换表序列 T^{-1},如表 5 所示:

Table 5 The Inverse Numeric Typed Permutation Table

表 5 数字型逆置换表

x	$T^{-1}(x)$	x	$T^{-1}(x)$
0	7	5	9
1	0	6	3
2	4	7	2
3	8	8	1
4	5	9	6

将加密后的数字型密文数据(7967752547866042806)一一进行逆置换表置换恢复操作,例如,数字“7”通过数字型逆置换表置换为数字“2”,数字“9”通过数字型逆置换表置换为数字“6”,依次进行全部置换操作,置换后,为了保证算法的安全,将数字型逆置换表进行删除操作,置换恢复得到数据为 2632294952133754173.然后在应用设备当中,将置换恢复得到数据与 PRESENT 轻量级分组密码的加密密钥(642A032F5010040760CB)进行一一对应相减、取模 10 运算操作,具体为

(2−6)%10=6,(6−4)%10=2,(3−2)%10=1,
(2−10)%10=2,(2−0)%10=2,(9−3)%10=6,
(4−2)%10=2,(9−15)%10=4,(5−5)%10=0,
(2−0)%10=2,(1−1)%10=0,(3−0)%10=3,
(3−0)%10=3,(7−4)%10=3,(5−0)%10=5,
(4−7)%10=7,(1−6)%10=5,(7−0)%10=7,
(3−12)%10=1.

将一一对应相减、取模 10 运算得到数字型数据为 6212262402033357571,加密后数字型密文数据得到恢复,恢复为真实数字型明文银行卡号.

4 算法的安全性

设计新的加密算法时,算法安全性是首要进行考虑的.面向数字型的轻量级保形加密算法,其置换表的构造取决于轻量级分组密码,而本保形加密算法的安全性是基于轻量级分组密码本身的安全性,且构造过程中是不会降低轻量级分组密码本身的安全性的,因此轻量级保形加密算法有着轻量级分组密码的安全性,另外与密钥相加、取模操作进一步提升了算法的安全.

本保形算法中采用轻量级 PRESENT 密码进行构造置换表,轻量级 PRESENT 密码自被提出

来,密码学者一直对其进行一系列的安全性分析,在文中第1节,已经给出了最新的轻量级PRESENT密码相关安全性分析结果,也表明目前轻量级PRESENT密码是足够安全.对于保形Prefix型加密算法而言,构造的不同加解密置换表之间具有随机性是确保保形加密算法本身的安全性.本保形加密算法的置换表构造过程中,将进行不同的置换表之间随机性测试,在测试实验中,轻量级PRESENT分组密码进行加密及密文排序处理分析,只需改变1 b的加密密钥,这1 b数据不同的加密密钥对应得到的置换表之间元素差别是非常大,测试结果如表6所示,通过表6给出的数据可以证明了利用轻量级PRESENT密码构造的置换表具有随机性,表明PRESENT密码构造置换表是安全的.

Table 6 The Randomness of Permutation Table

表6 置换表的随机性

Key(HEX)	Permutation Table(DEC)
0000 0000 0000 0000 0000	3278045916
0000 0000 0000 0000 0001	1342680975
0000 0000 0000 0000 0002	3547980126
0000 0000 0000 0000 0003	1354798260
0000 0000 0000 0000 0004	9301427658
0000 0000 0000 0000 0005	7624895103
0000 0000 0000 0000 0006	8314605729
0000 0000 0000 0000 0007	0973265814
0000 0000 0000 0000 0008	5027613498
0000 0000 0000 0000 0009	4056189327

对于传统的保形Prefix型加密算法而言,其置换表中的元素是0～9这10个不同的数字时,从而这数字型置换表的空间是为10!=3628800,由于这个置换表的空间不是足够大,从而明文容易被暴力攻击成功.为了解决传统的保形Prefix型算法存在的这个不足,本保形算法是对传统的保形Prefix型算法进行了改进,该改进方法是保证保形加密算法高效性的前提下使得改进后的算法具有高安全性,具体改进为:在算法中添加了将保形加密的明文与轻量级PRESENT密码的加密密钥进行一一对应相加、取模10运算操作,利用轻量级PRESENT密码的加密密钥进行保形加密算法的2次控制作用,真正实现了在保形Prefix型加密算法中加密密钥与保形加密明文的结合,从而进一步实现对保形加密的明文进行安全性保护.这种改进方法,在当密钥不知道的情况下是可以保护保形加密的明文数据,在置换加密操作前进行了提前变换保护,以抵抗暴力攻击明文.

定理1. 当明文与密钥对应相加、取模10操作,再经过置换表置换加密,保形Prefix型算法可以抵抗暴力攻击.

证明. 设一个保形加密的明文为A,一个加密密钥为B,一个运算后结果为C,则:

$$C=(A+B) \bmod 10, \tag{10}$$

其中,C是未知的(由于C的下一步操作是进行置换表加密操作,从而C数据是不进行公开),加密密钥B是进行严格保密,也是未知的,另外取模方程运算是包含多个解,因此暴力攻击破解的计算复杂度到达$10^{3628800}$,暴力破解明文A变为一个求解困难性问题,保形Prefix型算法可以有效地抵抗暴力攻击. 证毕.

另一方面,在保形Prefix型算法中,当输入一串连续性的数字明文,通过置换表加密之后,加密后的结果也是会出现一串连续性的数字密文.例如设输入一串连续的数字明文为0000000000,当置换表为"1876249035"进行加密情况下,则加密输出的密文结果为1111111111,从而出现明显的明文与密文之间关联关系特征暴露给攻击者,导致了保形加密算法安全性漏洞等问题.而这种保形加密的明文与轻量级PRESENT密码的加密密钥进行一一对应相加、取模10运算的方法,能够有效防止保形Prefix型加密算法出现这个漏洞性的问题.因为保形加密的明文与轻量级的加密密钥进行相加、取模10运算操作,是利用了加密密钥与保形加密明文的结合,进行了加密密钥的控制(实际应用设备要求用户设置口令密码不能简单为一串连续性数字),从而不会出现保形Prefix型算法的安全漏洞性问题.本保形加密算法的改进从这2个重要的方面保证算法的安全性,使设计的新保形加密算法具有很高的安全性能.

5 算法实现性能

面向数字型的轻量级保形加密算法,对于其软硬实现性能,关键在于创建置换表的分组密码选取,当置换表创建完成之后,接下来就是简单地相加、取模10及置换表置换操作,这些简单的运算操作对于整个保形加密算法实现性能影响是非常小的.本实

验的运行环境是处理器为 Intel Core i3-3220 CPU @3.30 GHz,内存为 8 GB,固态硬盘为 500 GB,操作系统为 Windows 7.

将算法进行 C 语言实现,在 Microsoft Visual C++ 6.0 软件上进行运行、编译.选择一个整型数字 0 扩充为 64 b 构造数字型置换表的待加密数据"0000000000000000(HEX)",通过 PRESENT 轻量级分组密码进行加密,加密花费时间为 0.001 s.将对应数字 0～9 这 10 个数字扩充为 10 个 64 b 构造数字型置换表的加密数据,通过 PRESENT 轻量级分组密码进行加密,加密总需花费时间为 0.01 s,由于对这 10 条密文进行大小分布标记不需要记录花费时间,构造置换表所需的时间就为 0.01 s.置换表构建创建之后,保形加密的明文数据进行与加密密钥进行相加、取模 10 及进行置换表置换操作,这些操作花费时间是微不足道的,我们在实验中对 500 条 19 位银行卡数字与加密密钥相加、取模 10 之后,再进行置换表(1876249035)置换运算,得到保形加密后的密文数据,花费的时间为 0.001 s.通过软件实验测试,整个轻量级保形加密算法加密 500 条 19 位银行卡数字需要花费时间为 0.011 s,得出面向数字型的轻量级保形加密算法具有实现速度快的特点.

面向数字型的轻量级保形加密算法是采用 PRESENT 轻量级分组密码构造置换表,而传统的保形 Prefix 型算法是采用 AES 密码构造置换表,分组密码构造置换表是保形加密算法实现性能的关键之处.

软件资源消耗方面,以下给出了 PRESENT-80 密码与 AES-128 密码在 8 b 处理器与 32 b 处理器软件实现资源的比较结果,如表 7 所示.通过 AES-128 密码与 PRESENT-80 密码占用的 ROM 与 RAM 数据结果对比,从而使用 PRESENT 密码构造置换表来实现保形加密,做到了保形加密软件实现轻量化.

Table 7 Comparison of PRESENT-80 and AES-128 in Software Implementation

表 7 PRESENT-80 密码与 AES-128 密码软件资源对比

Processor	Ciphers	Block Length/b	Key Length/b	ROM Size/B	RAM Size/B
8 b Processor	AES	128	128	2 742	127
	PRESENT	64	80	1 562	83
32 b Processor	AES	128	128	2 444	232
	PRESENT	64	80	1 760	280

硬件资源消耗方面,在 Xilinx ISE Design Suite 13.2 平台实现算法的 FPGA 测试,在 Synopsys Design Compiler version B-2016.06 平台实现算法的 ASIC 测试.在进行 FPGA 测试实验中,PRESENT-80 密码占用资源面积为 10 265 个 Slices,而 AES-128 密码占用资源面积为 20 173 个 Slices,在进行 ASIC 测试实验中,PRESENT-80 密码占用的面积资源为 1 570 个等效门(gate equivalents, GEs)及吞吐率为 200 Kbps,而 AES-128 密码占用面积资源为 3 400 个 GEs 及吞吐率为 12.4 Kbps.在硬件测试中,PRESENT 轻量级分组密码相比 AES 密码是具有低资源、高吞吐率的优点.轻量级保型加密算法在测试中,分别占用面积为 12 101 个 Slices 与 1 912 个 GEs(小于 RFID 实现安全组件 2 000 个 GEs),而传统的保形 Prefix 型加密算法占用资源分别为 21 440 个 Slices 与 3 628 个 GEs.轻量级保形加密算法相比传统的保形 Prefix 型加密算法,在占用面积资源上缩减了一半左右.通过这些数据分析,使用 PRESENT 密码构造置换表来实现保形加密,大大减少了加密硬件资源消耗.

在软硬件实现过程中,相比基于 AES 密码的传统保形 Prefix 型加密算法,本算法基于 PRESENT 密码实现了保形加密的轻量化,从而节省了软件实现的开销与硬件实现的成本,是能够适用于物联网终端设备数字型数据保形加密.

6 总 结

本文提出了一种面向数字型的轻量级保形加密算法,该算法是基于轻量级 PRESENT 分组密码来进行构造置换表,相比传统的保形 Prefix 型算法基于 AES 密码构造置换表,在软件平台实现中节省了 ROM 与 RAM 软件消耗的资源,在硬件平台实现中降低了实现成本,同时提高了吞吐率.另外,在保证算法具有高效的前提下,本算法相比传统的保形 Prefix 型算法,增加了与轻量级密码加密密钥一一

对应相加、取模 10 运算,大大提高了算法的安全性,并且解决了传统的保形 Prefix 型算法存在安全不足性的问题以及不能加密处理较长的数字型数据.在算法中,将轻量级 PRESENT 密码换为其他轻量级分组密码来构造置换表,也是可以达到较好的效果,所以本算法是具有通用价值.

未来进行探究保形加密算法中有关密钥管理的问题以及如何设计更高效、低资源的轻量级保形加密算法在资源受限的物联网设备终端进行运用.

参 考 文 献

[1] Cao Zhenfu. New development of information security—For the 60th anniversary of Journal of Computer Research and Development [J]. Journal of Computer Research and Development, 2019, 56(1): 131-137 (in Chinese)
(曹珍富. 信息安全的新发展——为《计算机研究与发展》创刊六十周年而作[J]. 计算机研究与发展, 2019, 56(1): 131-137)

[2] Bogdanov A, Knudsen L R, Leander G, et al. PRESENT: An ultra-lightweight block cipher [G] //LNCS 4727: Proc of the 9th Int Workshop on Cryptographic Hardware and Embedded Systems (CHES 2007). Berlin: Springer, 2007: 450-466

[3] Beierle C, Jean J, Kölbl S, et al. The SKINNY family of block ciphers and its low-latency variant MANTIS [G] // LNCS 9815: Advances in Cryptology (CRYPTO 2016). Berlin: Springer, 2016: 123-153

[4] Li Lang, Liu Botao, Zhou Yimeng, et al. SFN: A new lightweight block cipher [J]. Microprocessors & Microsystems, 2018, 60: 138-150

[5] Liu Zheli, Jia Chunfu, Li Jingwei. Research on the format-preserving encryption techniques [J]. Journal of Software, 2012, 23(1): 152-170 (in Chinese)
(刘哲理, 贾春福, 李经纬. 保留格式加密技术研究[J]. 软件学报, 2012, 23(1): 152-170)

[6] Institute for Computer Sciences and Technology, National Bureau of Standards. Guidelines for implementing and using the NBS data encryption standard, FIPS PUB 74 [R]. Gaithersburg, MD: Institute for Computer Sciences and Technology, National Bureau of Standards, 1981

[7] Black J, Rogaway P. Ciphers with arbitrary finite domains [G] //LNCS 2271: Cryptographers' Track at the RSA Conf (CT-RSA 2002). Berlin: Springer, 2002: 114-130

[8] Bellare M, Ristenpart T, Rogaway P, et al. Format-preserving encryption [G] //LNCS 5867: Proc of the 16th Int Workshop on Selected Areas in Cryptography (SAC 2009). Berlin: Springer, 2009: 295-312

[9] Jia Chunfu, Liu Zheli, Li Jingwei, et al. A new integer FPE scheme based on feistel network [G] //LNEE 155: Proc of the 2nd Int Conf on Electric and Electronics (EEIC 2012). Berlin: Springer, 2012: 637-644

[10] Bellare M, Hoang V T. Identity-based format-preserving encryption [C] //Proc of the 24th ACM SIGSAC Conf on Computer and Communications Security. New York: ACM, 2017: 1515-1532

[11] Bellare M, Hoang V T, Tessaro S. Message-recovery attacks on feistel-based format preserving encryption [C] //Proc of the 23rd ACM SIGSAC Conf on Computer and Communications Security. New York: ACM, 2016: 444-455

[12] Durak F B, Vaudenay S. Breaking the FF3 format-preserving encryption standard over small domains [G] //LNCS 10402: Advances in Cryptology (CRYPTO 2017). Berlin: Springer, 2017: 679-707

[13] Hoang V T, Tessaro S, Ni T. The curse of small domains: New attacks on format-preserving encryption [G] //LNCS 10991: Advances in Cryptology (CRYPTO 2018). Berlin: Springer, 2018: 221-251

[14] Zou Junwei, Wang Peng, Luo Hong. Improved prefix based format-preserving encryption for Chinese names [J]. China Communications, 2018, 15(3): 78-90

[15] Dworkin M J, Barker E B. Nechvatal J R, et al. Advanced encryption standard (AES), FIPS PUB 197 [R]. Gaithersburg, MD: Institute for Computer Sciences and Technology, National Bureau of Standards, 2001

[16] Banik S, Pandey S K, Peyrin T, et al. GIFT: A small present towards reaching the limit of lightweight encryption [G] //LNCS 10529: Proc of the 19th Int Workshop on Cryptographic Hardware and Embedded Systems (CHES 2017). Berlin: Springer, 2017: 321-345

[17] Zhang Wentao, Bao Zhenzhen, Lin Dongdai, et al. RECTANGLE: A bit-slice lightweight block cipher suitable for multiple platforms [J]. Science China Information Sciences, 2015, 58(12): 1-15. doi: 10.1007/s11432-015-5459-7

[18] Guo Jian, Peyrin T, Poschmann A, et al. The LED block cipher [G] //LNCS 6917: Proc of the 13th Int Workshop on Cryptographic Hardware and Embedded Systems (CHES 2011). Berlin: Springer, 2011: 326-341

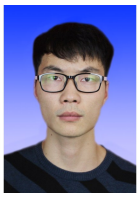
[19] Li Lang, Liu Botao, Wang Hui. QTL: A new ultra-lightweight block cipher [J]. Microprocessors & Microsystems, 2016, 45: 45-55

[20] Cho J Y. Linear cryptanalysis of reduced-round PRESENT [G] //LNCS 5985: Cryptographers' Track at the RSA Conf (CT-RSA 2010). Berlin: Springer, 2010: 302-317

[21] Blondeau C, Nyberg K. Links between truncated differential and multidimensional linear properties of block ciphers and underlying attack complexities [G] //LNCS 8441: Advances in Cryptology (EUROCRYPT 2014). Berlin: Springer, 2014: 165-182

[22] Zheng Lei, Zhang Shaowu. FFT-based multidimensional linear attack on PRESENT using the 2-bit-fixed characteristic [J]. Security & Communication Networks, 2016, 8(18): 3535-3545

[23] Faghihi Sereshgi M H, Dakhilalian M, Shakiba M. Biclique cryptanalysis of MIBS-80 and PRESENT-80 block ciphers [J]. Security & Communication Networks, 2016, 9(1): 27-33



Liu Botao, born in 1991. Master. His main research interests include data privacy and embedded system.

Peng Changgen, born in 1963. Professor, PhD supervisor in the College of Computer Science and Technology and master supervisor in the College of Mathematics and Statistics at Guizhou University. His main research interests include data privacy, cryptography and big data technology and security.

Wu Ruixue, born in 1995. Master. Her main research interests include privacy and data security.

Ding Hongfa, born in 1988. PhD candidate. His main research interests include privacy and data security.

Xie Mingming, born in 1993. Master. His main research interests include privacy and data security.