# 轻量级序列密码研究进展

王明兴[1,2]，苗三立[1]，朱明佳[1]

（1.中国电子信息产业集团有限公司第六研究所，北京 102209；2.密码科学技术国家重点实验室，北京 100878）

**摘　要：** 随着移动通信和物联网的快速发展，在硬件资源受限和功耗较低的应用场景中对密码保护的需求越来越大，密码算法的轻量化成为必然要求，近年来，轻量级序列密码的设计与分析是一个研究热点。首先介绍了四个新型的轻量级序列密码，它们共同的特点是内部状态小于密钥长度的两倍；其次介绍了每种算法的最新安全性评估结果；最后探讨了轻量级序列密码的发展趋势。

**关键词：** 小状态；轻量级序列密码；轮密钥函数；时间存储折中攻击

**中图分类号：** TP393　　**文献标识码：** A　　**DOI：** 10.19358/j.issn.2096-5133.2020.12.005

**引用格式：** 王明兴，苗三立，朱明佳. 轻量级序列密码研究进展[J].信息技术与网络安全，2020，39(12)：25-29.

## Research advances on lightweight stream cipher

Wang Mingxing[1,2], Miao Sanli[1], Zhu Mingjia[1]

(1.The 6th Research Institute of China Electronics Corporation, Beijing 102209, China;
2.State Key Laboratory of Cryptology, Beijing 100878, China)

**Abstract:** With the rapid development of mobile communication and the Internet of Things, the demand for cipher protection increases in the application scenarios with limited hardware resources and low power consumption, and then the lightweighting of cryptographic algorithm becomes an inevitable requirement. In recent years, design and analysis of lightweight stream cipher is a research hotspot. In this paper, we firstly introduce four new lightweight stream ciphers, which common characteristic is that their internal state is less than twice the length of the key; secondly, the latest security evaluation results of each stream cipher are introduced; finally, the development trend of the lightweight stream cipher is discussed.

**Key words:** small state; lightweight stream cipher; round key function; time memory data tradeoff



## 0 引言

序列密码是一种对称密码算法，即加密密钥和解密密钥是相同的。序列密码的经典结构包括两个部件：驱动部件和密钥流输出部件，驱动部件通过迭代运算更新内部状态，输出部件抽取一部分内部状态的比特值，经过复杂运算产生密钥流。加密时，密钥流和明文异或产生密文，解密时，密钥流和密文异或产生明文。通俗地讲，序列密码算法可以认为是分组密码的密钥扩展算法。

序列密码的一个结构特点是必然有内部状态，这使序列密码容易受到时间存储折中攻击(Time Memory Data Tradeoff，TMD)，为了抵抗这种攻击，通常要求序列密码的内部状态的长度至少是密钥长度的两倍，例如序列密码 Grain[1]、Grain-128a[2]系列密码算法，它们的内部状态是密钥长度的两倍，而 Trivium 算法[3]的内部状态大于密钥长度的两倍。这导致序列密码在硬件实现时所占用的硬件资源代价过大，例如 Trivium 算法的硬件面积是2 580 门，不适用于资源受限、低功耗、物联网等新的应用场景。

为了既能抵抗 TMD 攻击，又能减小内部状态的长度，研究人员最近几年提出了“小状态”轻量级序列密码研究。那些能够抵抗 TMD 攻击且内部状态的长度小于密钥长度的两倍的轻量级序列密码，称为小状态轻量级序列密码(Small State Lightweight Stream Cipher)。

本文介绍了四个小状态的轻量级序列密码算法，依次是 Sprout[4]、Fruit-80[5]、Plantlet[6]、Lizard[7]；主要介绍它们的设计理念和最新的分析结果，来发

现存在的问题，总结研究成果，促进小状态的轻量级序列密码的研究。

## 1 轻量级序列密码介绍

本文介绍的四个算法都是轻量级序列密码，都是试图在小状态的前提下，设计安全的密码算法，它们整体结构高度相似，都基于 Grain 算法的驱动结构，采用线性或者非线性反馈移位寄存器(Linear/Non-linear Feedback Shift Register，LFSR/NFSR)的串联结构，见图 1 虚线部分。LFSR 会影响 NFSR，反之，LFSR 不受 NFSR 的影响，后续的分析表明，这样的结构特点带有安全隐患。以下分别描述各个算法的设计特点。

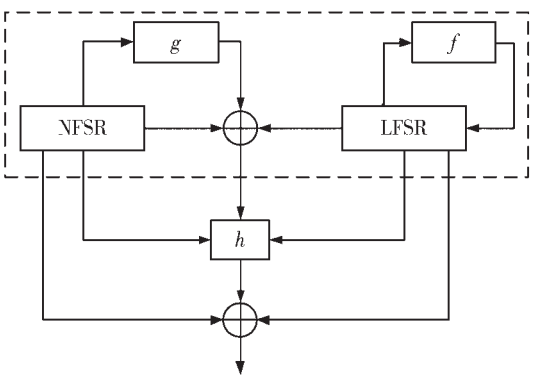


图 1 序列密码 Grain 算法简图

### 1.1 Sprout 算法

Sprout 算法是 ARMKNECHT F 等人[4]在 2015 年提出的第一个小状态的轻量级序列密码，其核心设计思想是，通过把内部状态分成 $2^{|K|}$ 等价类，使得 TMD 攻击每次至少考虑一个等价类，这里 $|K|$ 是密钥长度。实现这一思想的方法是密钥一直参与内部状态的迭代过程。而 Sprout 算法通过在 Grain 结构的基础上增设一个轮密钥函数，使得密钥不仅参与算法初始化过程，而且参与密钥流的产生过程，达到了密钥一直参与内部状态的更新的目的。

Sprout 算法的驱动部件是 40 bit 长的 LFSR 串联上一个 40 bit 长的 NFSR，以及轮密钥函数和计数器。密钥长度 80 bit，初始向量 70 bit，一个初始向量产生的密钥流不超过 $2^{40}$ bit。初始化过程中输出密钥流比特反馈参与移位寄存器的运算，初始化轮数 320 轮，初始化完成之后，每一拍输出 1 bit 密钥流。

### 1.2 Fruit-80 算法

GHAFARI V A 等人[8]于 2016 年提出了 Fruit 轻量级序列密码，Fruit 可以看作是 Sprout 算法的升级版本。随后的公开分析结果表明，该算法的主要结构是抵抗密钥恢复攻击的，但是前提条件是某些参数的选取要适当；于是 Fruit 的升级版算法 Fruit-v2 被提出[9]。设计者提出了四方面的改进：使用了新的轮函数，修改了 Sprout 的弱点；使用新的初始化方案来抵抗相关密钥攻击，防止密钥输出过程中线性移位寄存器的内部状态出现全零状态；增加了线性反馈移位寄存器的阶数，以产生周期更长的密钥流序列；减少了反馈函数和输出函数的项数。

2018 年 Fruit 设计者接受了文献[10]的建议，即在算法运算过程中初始向量 IV 一直参与混淆，而且还提出了限制使用每一个密钥产生的密钥流的长度的改进措施，最终提出了 Fruit-80 轻量级序列密码[5]。其 LFSR 长度是43 bit，NFSR 长度是37 bit，密钥长度是 80 bit，初始向量 70 bit，每个密钥和初始向量 IV 所产生的密钥流的长度不超过 $2^{43}$ bit，而且初始向量 IV 不能重复使用。

### 1.3 Plantlet 算法

轻量级序列密码 Plantlet 是 MIKHALEV V 等人[6]于 2017 年提出的，它也可以看作是 Sprout 算法的升级版。其密钥长度 80 bit，初始向量长度 90 bit，内部状态 101 bit，初始化轮数 320 轮，设计安全强度是 80 bit。

该算法硬件代价小，内部状态小，吞吐率高，密钥非易失性存储(Non-Volatile Memory，NVM)，在计算过程中可以持续读取。与 Sprout 算法和 Fruit 算法相比，它有两方面的改进：一是轮密钥不再同时依赖于密钥和当前状态，而是只依赖于密钥；二是线性反馈移位寄存器在初始化过程和密钥流产生过程中使用的是不同的线性反馈移位寄存器。

### 1.4 Lizard 算法

轻量级序列密码 Lizard 是 HAMANN M 等人[7]在 2017 年提出的面向蓝牙、无线局域网、超文本传输协议的应用场景的轻量级序列密码，密钥长度是 120 bit，初始向量 64 bit，内部状态是 121 bit，提供 80 bit 安全强度。初始化轮数 256 轮。

该算法突出的特点是，每一个密钥和初始向量对(K，IV)限制产生的密钥流长度是 $2^{18}$ bit，它的驱动部件是两个非线性反馈移位寄存器的串联结构，密钥在初始化过程中两次参与运算，Lizard 的能耗特别低，只有 2.1 μW。

不同算法的各项指标对比见表 1。不难发现，这四个轻量级序列密码的硬件面积比 Trivium 算法

表 1 不同算法各项安全指标汇总表

| 名称 | 密钥长度/bit | 安全强度/bit | 初始向量/bit | 内部状态/bit | 初始化轮数 | 硬件面积/门 |
|---|---|---|---|---|---|---|
| Sprout | 80 | 80 | 70 | 80 | 320 | 813 |
| Fruit | 80 | 80 | 70 | 80 | 160 | 960 |
| Plantlet | 80 | 80 | 90 | 101 | 320 | 996 |
| Lizard | 120 | 80 | 64 | 121 | 256 | 1 218 |
| Trivium | 80 | 80 | 70 | 288 | 1 152 | 2 580 |

的一半还要小，因为内部状态越小，硬件面积越小。算法的安全目标都是 80 bit。综合而言，Fruit 算法的运算效率和硬件代价都较为优越，非常适合资源受限的应用场景。

## 2 轻量级序列密码安全性分析结果

### 2.1 Sprout 算法分析结果

2015 年，LALLEMAND V 等人[11]宣称，Sprout 的密钥恢复攻击的时间复杂度是 $2^{70}$，他们联合使用分别征服攻击和猜测确定攻击技术，主要是利用了移位寄存器的尺寸小，以及受轮密钥非线性影响的更新函数与轮密钥本身的依赖关系不够密切的特点。

在当年的印度密码学年会上，BANIK S[12]指出，如果猜测 50 bit 的内部状态，剩下的内部状态的比特值可以使用 SAT 求解器恢复，基于这个假设，给出了一个区分攻击，随机选择 $2^{40}$ 个初始向量 IV，存储复杂度是 $2^{48}$。由于 LFSR 只影响 NFSR，反之不受影响，那么，平均存在 $2^{30}$ 个初始向量 IV 使得LFSR 的内部状态在密钥流阶段出现全零的情况，基于这一点，密钥恢复攻击的时间复杂度相当于 $2^{66.7}$ 次加密运算，而存储复杂度忽略不计。

在 SAC2015 会议上，ESGIN M F 等人[13]也指出，联合使用猜测确定技术和分别征服攻击，Sprout 算法的 TMD 的存储复杂度为 $2^{80-a}$，$a \leqslant 40$，需要执行 $2^{71-a}$ 次加密运算和 $2^a$ 次查表运算得到 $2^a$ bit 密钥流；猜测确定攻击的时间复杂度相当于 $2^{68}$ 次加密运算，存储复杂度忽略不计。

### 2.2 Fruit 算法分析结果

HAMANN M 等人[10]指出 Fruit-v1 和 Fruit-v2 没有达到 80 bit 的安全性，他们发现了 $2^{64}$ 个弱密钥，利用移位寄存器的特定状态和初始化过程的可逆性，进行了 TMD 攻击，可以恢复密钥。TODO Y 等人[14]也指出 Fruit-80 的设计策略，即每一对密钥和初始向量(K，IV)至多产生 $2^{43}$ bit 密钥流，对安全而言是不够的。他们利用扩展了的相关攻击，发现轮密钥的引入具有周期性，只要找到高相关的多维线性掩码逼近密钥流输出函数，利用不同初始向量产生的密钥流，在时间复杂度为 $2^{77.870\,2}$、存储复杂度为 $2^{43+21}$ 的条件下，可以恢复全轮的 Fruit-80 密钥。

### 2.3 Plantlet 算法分析结果

TODO Y 等人[14]指出，如果一对密钥初始向量(K，IV)产生的密钥流的长度为 $2^{53}$ bit，利用扩展的相关攻击可以恢复全轮的 Plantlet 算法的密钥。而 Plantlet 算法的设计者限制一对密钥、初始向量(K，IV)可以使用密钥流的长度为 $2^{30}$ bit，显然这有利于增强算法的安全性。MAITRA S 等人[15]对 Plantlet 算法进行了差分错误攻击(Differential Fault Attack，DFA)，所谓差分错误攻击，是指在 FPGA 电路板卡上，使用物理手段使得算法的某些特殊状态的比特值发生翻转，通过分析翻转前后的不同密钥流之间的关系，来恢复内部状态。MAITRA S 等人指出只需四个错误就可以实际地恢复内部状态，进而恢复密钥。但是需要指出的是，错误信号的注入条件苛刻，不能说明算法在正常运行的情况下存在安全弱点。

### 2.4 Lizard 算法分析结果

BANIK S 等人[16]发现，进行 $2^{58}$ 次随机试验，可以找到 $2^{64}$ 个三元组$(K, IV_0, IV_1)$，使得两个密钥、初始向量对$(K, IV_0)$、$(K, IV_1)$产生相同的密钥流。基于这种不同的初始向量产生移位关系的密钥流序列，对 Lizard 算法进行了区分攻击，计算复杂度是 $2^{51.5}$ 次加密运算，存储复杂度是 $2^{76.6}$。进而，基于初始向量的碰撞性，恢复 223 轮的密钥的时间复杂度为 $2^{69}$。MAITRA S 等人[17]对Lizard 进行了时间存储折中攻击，恢复内部状态的预计算复杂度为 $2^{67}$，在线时间复杂度是 $2^{54}$，但是由于初始化过程不可逆，因此不能恢复密钥。SIDDHANTI A 等人[18]对 Lizard 进行了差分错误攻击，至少需要五个错误才能恢复内部

状态，但是也没有恢复密钥。

总结目前的分析结果，对轻量级序列密码的设计建议如下：

(1)轮密钥应线性地影响更新函数，降低轮密钥的某些比特值被非线性函数零化掉的风险。

(2)两个移位寄存器的长度(分别为 40 bit)都太小，应至少长 50 bit，这样猜测确定攻击才没有优势。

(3)建议使用两个 NFSR 串联，而不是 LFSR 和 NFSR 的串联。LFSR 的线性关系在轻量级算法中产生了安全隐患，显然，Lizard 算法的安全性明显高一些。

(4)初始向量与密钥一样都在初始化过程和密钥流产生过程参与运算，这样初始向量和密钥会混淆和扩散得更加充分，更能抵抗 TMD 攻击。

(5)改变传统的每次加密运算输出 1 bit 密钥流的方式，例如，移位寄存器每连续移位 16 步，输出它们的异或运算和作为1 bit 密钥流，能有效增加猜测确定攻击的难度。

(6)根据算法实际的使用场景，限制一个密钥、初始向量对(K，IV)产生的密钥流序列的长度，这样的密钥流越短，算法的安全强度越高。

## 3 讨论

本文介绍的轻量级序列密码，严格意义上讲都是不安全的。因为与 Grain 算法相比，在内部状态减小的情况下，对算法的反馈函数、密钥流输出函数、密钥和初始向量的引入方式以及对密钥流的使用长度的设定都提出了很高的设计要求，设计者很难多方面兼顾。

序列密码的小状态的设计理念值得肯定，因为这样的设计利于算法的轻量化，便于满足实际场景的需求；但是这种基于已有驱动模型不断增设新的部件、复杂化的设计方法，例如增加轮密钥函数、改变初始化过程、增加计数器等值得商榷，因为这违背了简单、易分析的传统的设计理念，而且从分析的结果来看，这样做反而容易带来安全隐患。

密钥非易失性存储的设计方法可以增强算法的安全性，但是增加了功耗。如何平衡安全性和功耗以及硬件资源代价是轻量级序列密码设计的难点所在，这需要设计者根据应用需求进行灵活的选择。

考虑到密码算法的实际应用场景，轻量级序列密码算法将更加注重应用需求，进一步降低硬件实现代价，提高算法效率。通过限制使用的密钥流的长度，使得安全性满足应用要求即可，实现算法的安全性和硬件实现代价的有效折中。

## 4 结论

轻量级序列算法 Sprout、Fruit、Lizard 和 Plantlet 的提出以及相应的分析结果标志着序列密码轻量级化取得了新的研究进展，减小内部状态的设计方法得到了具体实现，这一设计理念得到密码学界的逐步认可。下一步，算法要应用目前从分析结果中得到的一些设计技巧，而且要进一步研究提高算法安全性的设计方法；同时，寻找新的驱动部件来设计轻量级序列密码将是一个有意义的研究方向。

## 参考文献


[1] HELL M，JOHANSSON T，MEIER W. Grain：a stream cipher for constrained environments[J]. International Journal of Wireless and Mobile Computing，2007，2(1)：86-93.

[2] ÅGREN M，HELL M，JOHANSSON T，et al. Grain-128a：a new version of Grain128 with optional authentication[J]. International Journal of Wireless and Mobile Computing，2011，5(1)：48-59.

[3] CANNIERE C D，PRENEEL B. TRIVIUM specifications[J]. Estream Ecrypt Stream Cipher Project，2008，2006(3)：233-236.

[4] ARMKNECHT F，MIKHALEV V. On lightweight stream ciphers with shorter internal states[C]. International Workshop on Fast Software Encryption(FSE2015). Springer，Berlin，Heidelberg，2015(9054)：451-470.

[5] GHAFARI V A，HU H. Fruit-80：a secure ultra-lightweight stream cipher for constrained environments[J]. Entropy，2018，20(3)：180-193.

[6] MIKHALEV V，ARMKNECHT F，MULLER C. On ciphers that continuously access the non-volatile key[J]. IACR Transaction of Symmetric Cryptology，2016(2)：52-79.

[7] HAMANN M，KRAUSE M，MEIER W. Lizard-a lightweight stream cipher for power-constrained devices[J]. IACR Transaction of Symmetric Cryptology，2017(1)：45-79.

[8] GHAFARI V A，HU H，CHEN Y. Fruit：Ultra-light-weight stream cipher with shorter internal state[EB/OL]. (2016-04-08). https://eprint.iacr.org/2016/355.pdf.

[9] GHAFARI V A，HU H. Fruit-v2：Ultra-lightweight stream cipher with shorter internal state[Z]. IACR

Cryptology ePrint Archive 2016/355.
[10] HAMANN M, KRAUSE M, MEIER W, et al. Design and analysis of small-state Grain-like stream ciphers[J]. Cryptography and Communications, 2017, 10(5): 803-834.
[11] LALLEMAND V, NAYA-PLASENCIA M. Cryptanalysis of Full Sprout[C]. Advances in Cryptology-CRYPTO 2015. Springer, Berlin, 2015(9215): 663-682.
[12] BANIK S. Some results on Sprout[C]. Progress in Cryptology-INDOCRYPT 2015. Springer, Cham, 2015(9462): 124-139.
[13] ESGIN M F, KARA O. Practical cryptanalysis of Full Sprout with TMD tradeoff attacks[C]. International Conference on Selected Areas in Cryptography. Springer, Cham, 2015(9566): 67-85.
[14] TODO Y, MEIER W, AOKI K. On the data limitation of small-state stream ciphers: correlation attacks on Fruit-80 and Plantlet[C]. Selected Areas in Cryptography-SAC 2019. Springer, Chan, 2020(11959): 365-392.
[15] MAITRA S, SIDDHANTI A, SARKAR S. Differential fault attack on Plantlet[J]. IEEE Transactions on Computers, 2017, 66(10): 1804-1808.
[16] BANIK S, ISOBE T, CUI T T. Some cryptanalytic results on Lizard[Z]. IACR Cryptology eprint Archive 2017/346.
[17] MAITRA S, SINHA N, SIDDHANTI A, et al. A TMDTO attack against Lizard[J]. IEEE Transactions on Computers, 2017, 67(5): 733-739.
[18] SIDDHANTI A, SARKAR S, MAITRA S, et al. Differential fault attack on Grain v1, ACORN v3 and Lizard[C]. Security, Privacy, and Applied Cryptography Engineering(SPACE 2017). Springer, Cham, 2017(10662): 247-263.

（收稿日期：2020-07-09）

**作者简介：**

**王明兴**(1983-)，男，博士研究生，工程师，主要研究方向：信息安全。

**苗三立**(1991-)，男，硕士研究生，助理工程师，主要研究方向：信息安全。

**朱明佳**(1993-)，男，本科，助理工程师，主要研究方向：网络安全。



# 《信息技术与网络安全》2021年征文主题

| 月份 | 主题 | 分主题方向 |
|---|---|---|
| 1、2月 | 网络安全防护技术 | 1. 安全防护技术；2. 安全审计技术；3. 安全检测与监控技术；<br>4. 解密、加密技术；5. 身份认证技术等 |
| 3、4月 | 工业互联网安全 | 1.工业互联网安全检测评估；2.网安全协议标准研究；3.安全态势感知技术研究；<br>4.身份安全认证体系；5.安全架构技术研究;6.智能化的安全防护体系；<br>7.工业互联网软件监测安全研究；8.工业互联网通信系统安全研究 |
| 5、6月 | 物联网安全 | 1. 物联网终端设备的安全认证技术；2. 物联网射频识别技术研究；<br>3. 物联网入侵检测和态势感知技术；4. 物联网安全路由技术；<br>5. 物联网容错、容侵技术 |
| 7、8月 | 数据安全 | 1.数据库安全防护技术；2.数据访问、数据爬取等行为检测和审计技术研究；<br>3.大数据环境下数据脱敏技术研究；4.云安全数据防护技术；<br>5.敏感数据分布、漏洞扫描、威胁预警等数据安全管理技术研究 |
| 9、10月 | 网络测绘技术 | 1.网络靶场技术；2.网络流量技术研究；3.网络测量技术研究；<br>4.漏洞分析研究；5.恶意代码分析与检测 |
| 11、12月 | 人工智能与安全 | 1. 人工智能发展战略研究；2. 人工智能标准研究；3. 人工智能入侵检测技术；<br>4. 人工智能体态识别与行为监测；5. 人工智能用户实体行为分析（UEBA）；<br>6. 人工智能数据检索和分析技术；7. 人工智能算法安全、模型安全研究；8. 人工智能可控技术研究 |

# 版权声明