基于 Nonce 重用的 ACORN v3 状态恢复攻击

张国双[1,2]，陈晓[1,2]，林东岱[1,2]，刘凤梅[3]

（1. 中国科学院信息工程研究所，北京 100093；2. 中国科学院大学网络空间安全学院，北京 100049；
3. 信息保障技术重点实验室，北京 100072）

摘 要：基于差分代数方法，利用猜测确定技术给出了 Nonce 重用两次情况下 ACORN v3 的状态恢复攻击，攻击所需的计算复杂度为 $2^{122.5}c$，数据复杂度和存储复杂度可忽略不计，其中 c 是求解线性方程组的复杂度。针对 Nonce 多次重用时的情形进行了分析，发现 ACORN v3 较复杂的滤波函数，使由密钥流直接提取关于内部状态线性方程的方法变得不可行，从而有效规避了通过增加 Nonce 重用次数来显著降低攻击复杂度的安全风险。

关键词：认证加密；密码分析；ACORN；状态恢复攻击

中图分类号：TN918.1

文献标识码：A

doi: 10.11959/j.issn.1000−436x.2020164

State recovery attack on ACORN v3 in nonce-reuse setting

ZHANG Guoshuang[1,2], CHEN Xiao[1,2], LIN Dongdai[1,2], LIU Fengmei[3]

1. Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
2. School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100049, China
3. Science and Technology on Information Assurance Laboratory, Beijing 100072, China

Abstract: Based on differential-algebraic method and guess-and-determine technique, the state recovery attack of ACORN v3 was presented when one pair of key and Nonce was used to encrypt two messages. The time complexity of the attack was $2^{122.5}c$, where c was the time complexity of solving linear equations. The data complexity and the storage complexity were negligible. Furthermore, according to the analysis on the sense of multiple nonce reuse, it is found that relatively complicated filter function of ACORN v3 makes it infeasible to extract the linear equations about the internal state directly from key streams. Thus, the risk of significantly reducing the attack complexity by increasing the times of nonce reuse can be effectively avoided.

Key words: authenticated cipher, cryptanalysis, ACORN, state recovery attack

1 引言

加密和认证是密码学的 2 个基本属性。在现代网络保密通信中，基于密码构建信息系统对消息进行加密和认证处理，是实现数据机密性和认证性保护的有效途径，然而由于使用不当或恶意敌手后门植入等，可能会造成 Nonce 重复使用、未按规则返回明文等情况发生，从而为潜在敌手攻击和分析密码算法提供便利。Nonce 重用的状态与密钥恢复攻击一般模型如图 1 所示，攻击者通过植入后门，造成发送方使用相同的 Nonce 对不同消息进行加密处理，从而获得不同结构的明文所对应的不同密文。利用这些密文，可对密钥或状态进行恢复分析，一旦成功地恢复出密钥或状态，就可以对保密通信进行实时解密监听或篡改伪造。

收稿日期：2020−04−22；**修回日期**：2020−07−05

基金项目：国家自然科学基金资助项目（No.61872040）；“十三五”国家密码发展基金资助项目（No.MMJJ20170201）；北京市自然科学基金资助项目（No.4202070）

Foundation Items: The National Natural Science Foundation of China (No.61872040), “The 13th Five-Years” National Cryptogram Development Fund (No.MMJJ20170201), Beijing Municipal Natural Science Foundation (No.4202070)

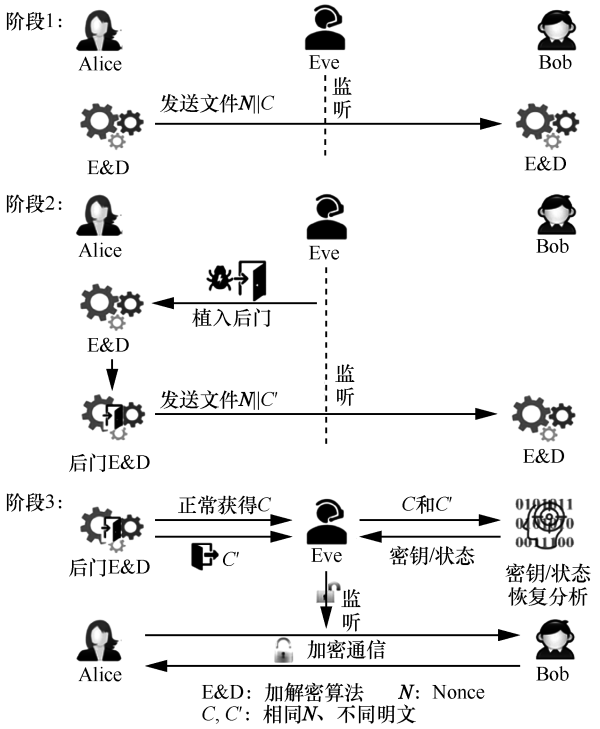


图 1　Nonce 重用的状态与密钥恢复攻击一般模型

为进一步提升认证加密算法的安全强度，增强人们对认证加密算法的认识和信心，2013 年，国际密码协会（IACR, International Association for Cryptologic Research）面向全球发起了征集认证加密算法的 CAESAR（competition for authenticated encryption: security, applicability, and robustness），旨在遴选安全高效的认证加密算法。CAESAR 自 2013 年开始至 2019 年结束持续 6 年时间，最终有 6 个算法胜出并作为认证加密算法的代表，ACORN v3 算法便是其中之一。ACORN 算法由 Wu[1]提出，是一个面向比特的轻量认证加密算法，并以其新颖的设计和轻量化高效实现引起了国内外密码学界的广泛关注和研究兴趣。

ACORN 算法自发布以来历经了 3 个版本[1-3]，目前的最新版本是 ACORN v3。针对 ACORN 算法安全性的研究很多[4-19]。其中，关于 ACORN 算法的状态或密钥恢复攻击，Liu 等[4]根据 ACORN v1 的滑动特性，利用差分代数技术给出了 Nonce 重用两次时 ACORN v1 的状态恢复攻击；Chaigneau 等[5]研究并给出了 Nonce 多次重用时 ACORN v1 的密钥恢复攻击；Wang 等[6]研究和评估了 Nonce 重用情况下 ACORN v2 的状态恢复攻击和复杂度；针对 ACORN v2 和 v3，Zhang 等[7]进一步研究并给出了 Nonce 重用三次情况下的状态恢复攻击。

然而，在实际应用中，Nonce 重用本身是小概率事件，出现 Nonce 多次重用的概率就更小，同时，密码系统可能会采用一定的手段来减少 Nonce 重用情况的发生。因此，攻击所需要的 Nonce 重用次数越多，则实际实施的难度越高，因此 Nonce 重用次数的多少是此类攻击是否容易实施的关键指标。

针对以上问题，本文给出了一种仅需 Nonce 重用两次的 ACORN v3 状态恢复攻击，该攻击通过对中间变量的猜测以期构造尽可能多关于内部状态的线性方程。结果显示，即使 Nonce 仅重用两次，对于 ACORN v3 同样存在低于穷搜索复杂度的状态恢复攻击，攻击所需的计算复杂度为 $2^{122.5}c$，其中，c 是求解 293 bit 变元线性方程组的复杂度，数据复杂度和存储复杂度可忽略不计。此外，基于本文方法对 Nonce 多次重用时的安全性进行分析发现，由于 ACORN v3 采用了较之前版本复杂的滤波函数，从而有效避免了通过增加 Nonce 重用次数来显著降低状态恢复攻击复杂度的潜在问题。研究结果也进一步印证了 ACORN 算法安全性声明中强调的 Nonce 不能被重用的要求。表 1 给出了 ACORN 算法状态恢复攻击的结果对比。

表 1　ACORN 算法状态恢复攻击的结果对比

算法版本	Nonce 重用次数	数据复杂度/bit	计算复杂度	文献
ACORN v1	2	164	$2^{114.6}c$	文献[4]
ACORN v2	2	2×100	$2^{193}c$	文献[5]
ACORN v2	3	3×142	$2^{151}c$	文献[5]
ACORN v2	4	4×184	$2^{109}c$	文献[5]
ACORN v2	2	2×192	$2^{126.5}c$	文献[6]
ACORN v2	3	3×192	$2^{98.2}c$	文献[6]
ACORN v2	4	4×192	$2^{75.8}c$	文献[6]
ACORN v2	3	3×234	$2^{78}c$	文献[7]
ACORN v3	3	3×234	$2^{120.6}c$	文献[7]
ACORN v3	2	2×148+293	$2^{122.5}c$	本文
ACORN v3	3	3×98+293	$2^{118.3}c$	本文

2　符号说明与 ACORN 算法简介

2.1　符号说明

本文使用的符号解释如表 2 所示。本文约定所有计数从 0 开始，并且数据的左侧为最低位，右侧为最高位。

表 2　　符号解释

符号	解释
$\boldsymbol{S}^t$	t 时刻内部状态
s_i^t	t 时刻内部状态 $\boldsymbol{S}^t$ 的第 i 个比特
k_t	t 时刻生成的密钥比特
f_t	t 时刻生成的更新比特
m_t	t 时刻输入比特
$\boldsymbol{K}$	128 bit 算法密钥
$\boldsymbol{N}$	128 bit Nonce
AD	关联数据
P	明文
p_t	明文 P 的第 t 个比特
$\Delta\boldsymbol{S}^t$	t 时刻的内部状态差分
Δs_i^t	t 时刻内部状态第 i 个比特的差分
Δk_t	第 t 个密钥流比特的差分

2.2　ACORN 算法简介

ACORN 算法利用 128 bit 的密钥和 128 bit Nonce 可以完成对长度不超过 2^{64} 的明文和关联数据的保护，并产生长度不超过 128 bit 的认证标签。ACORN 采用特定设计的序列密码结构，由 6 个不同的线性移位寄存器和一个 4 bit 的缓存器构成 293 bit 状态移位寄存器，整体采用二次的反馈函数，根据额外的输入比特对内部状态进行更新，并使用二次的密钥流生成函数，每一拍生成 1 bit 的密钥，其认证加密过程包含以下 4 个环节。

1) 初始化环节。加载密钥和 Nonce 并生成初始状态。

2) 关联数据处理环节。处理关联数据并进行内部状态更新。

3) 加密环节。对明文加密并进行内部状态更新。

4) 认证码生成环节。用于生成认证标签。

与 ACORN v1 相比，ACORN v2 对初始化过程进行了修改，并调整了 4 个环节迭代的拍数；相对于 ACORN v2，ACORN v3 对密钥流生成函数和反馈函数进行了调整。以下简要介绍与本文分析有关的 ACORN v3 的主要变换环节。ACORN v3 算法的原理如图 2 所示。

图 2 中，k_t、f_t 和 m_t 分别是密钥比特、状态更新比特和输入比特；a_t 和 b_t 是 2 个控制比特，影响 f_t 的计算；maj(·) 和 ch(·) 是定义在 GF(2) 上的 2 个三元布尔函数；密钥流生成函数和更新比特生成函数分别用于生成密钥比特和状态更新比特。令 t 时刻 ACORN v3 的内部状态 $\boldsymbol{S}^t=(s_0^t,s_1^t,\cdots,s_{292}^t)$，则有密钥流生成函数 $k_t=G(\boldsymbol{S}^t)$。

$$k_t=s_{12}^t\oplus s_{154}^t\oplus \mathrm{maj}(s_{235}^t,s_{61}^t,s_{193}^t)\oplus \mathrm{ch}(s_{230}^t,s_{111}^t,s_{66}^t)$$

更新比特生成函数 $f_t=F(\boldsymbol{S}^t,a_t,b_t)$。

$$f_t=1\oplus s_0^t\oplus s_{107}^t\oplus \mathrm{maj}(s_{244}^t,s_{23}^t,s_{160}^t)\oplus a_t s_{196}^t\oplus b_t k_t$$

$$\mathrm{maj}(x,y,z)=xy\oplus xz\oplus yz$$

$$\mathrm{ch}(x,y,z)=xy\oplus(1\oplus x)\ z=xy\oplus xz\oplus z$$

ACORN v3 的状态更新变换包括 LFSR（linear feedback shift register）状态线性更新、计算并生成

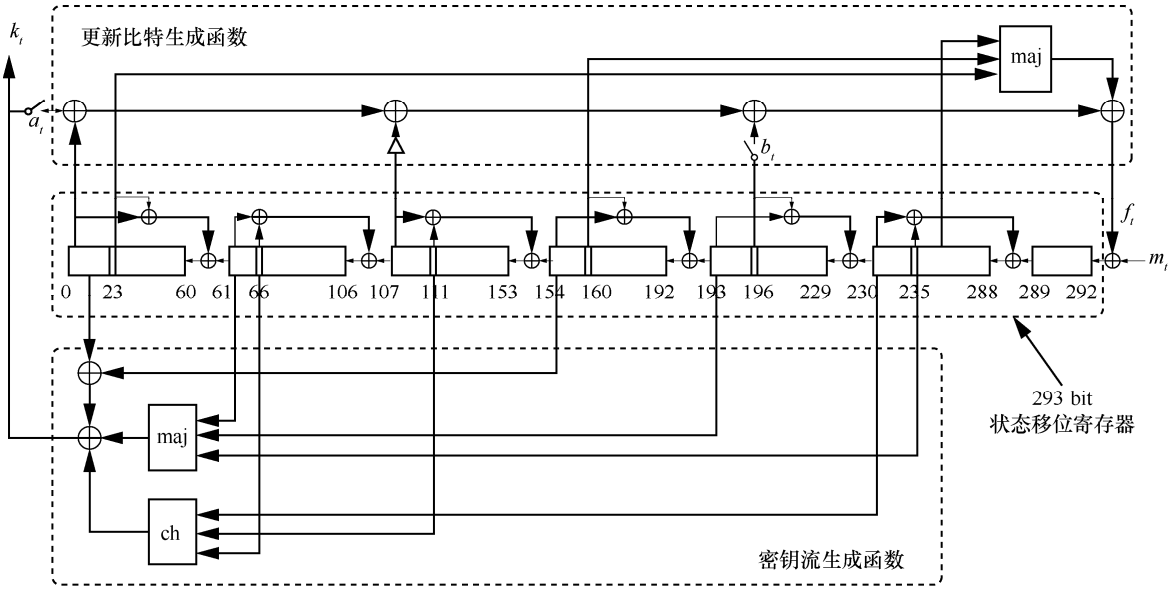


图 2　ACORN v3 算法的原理

密钥流比特、计算并生成非线性反馈比特和 293 级移位寄存器状态更新。记状态更新变换为

$$\boldsymbol{S}^{t+1}=\text{StateUpdate}(\boldsymbol{S}^t,m_t,a_t,b_t)$$

则状态更新变换的具体过程如下。

Step1 LFSR 状态线性更新。

$$s_{289}^t=s_{289}^t\oplus s_{235}^t\oplus s_{230}^t$$

$$s_{230}^t=s_{230}^t\oplus s_{196}^t\oplus s_{193}^t$$

$$s_{193}^t=s_{193}^t\oplus s_{160}^t\oplus s_{154}^t$$

$$s_{154}^t=s_{154}^t\oplus s_{111}^t\oplus s_{107}^t$$

$$s_{107}^t=s_{107}^t\oplus s_{66}^t\oplus s_{61}^t$$

$$s_{61}^t=s_{61}^t\oplus s_{23}^t\oplus s_0^t$$

Step2 计算并生成密钥流比特。

$$k_t=G(\boldsymbol{S}^t)$$

Step3 计算并生成状态更新比特。

$$f_t=F(\boldsymbol{S}^t,a_t,b_t)$$

Step4 293 级移位寄存器状态更新。

$$s_i^{t+1}=s_{i+1}^t,0\leqslant i\leqslant 291$$

$$s_{292}^{t+1}=f_t\oplus m_t$$

ACORN v3 的 4 个环节（初始化环节、关联数据处理环节、加密环节和认证码生成环节）中控制比特 a_t、b_t 取值与输入比特 m_t 的对应关系如表 3 所示。

$\boldsymbol{K}$ 和 $\boldsymbol{N}$ 分别表示 128 bit 的密钥和 Nonce，$\boldsymbol{K}'$ 表示将 $\boldsymbol{K}$ 的最高比特位取反其余比特位保持不变，AD 和 P 分别表示待处理的关联数据和明文数据。密文比特 $c_t=p_t\oplus k_t$，认证码 T 由最后 l bit 的密钥流给定。

3 ACORN v3 状态差分特性分析

本节挖掘了 maj(·) 和 ch(·) 的 4 条性质，并结合文献[5]中提出的一条性质，共同推出了 ACORN v3 加密过程的状态差分传播规律。

3.1 ACORN 算法函数性质研究

maj(·) 和 ch(·) 是 ACORN 算法中用到的 2 个最基本的非线性函数。2015 年，Chaigneau 等[5]在对 ACORN v1 的状态恢复攻击中发现，maj(·) 函数具有性质 1。

性质 1[5] 记 $\text{maj}(x,y,z)=a$，若 $\text{maj}(1\oplus x,y,z)=b$，则

$$a\oplus b=y\oplus z$$

$$(a\oplus b)(x\oplus y)=a\oplus y$$

本文研究发现，maj(·) 和 ch(·) 还具有性质 2～性质 5。

性质 2 记 $\text{maj}(x,y,z)=a$，若 $\text{maj}(1\oplus x,y,1\oplus z)=b$，则

$$a\oplus b=1\oplus x\oplus z$$

$$(1\oplus a\oplus b)(y\oplus z)=a\oplus z$$

性质 3 记 $\text{maj}(x,y,z)=a$，若 $\text{maj}(x,y,1\oplus z)=b$，则

$$a\oplus b=x\oplus y$$

$$(a\oplus b)(y\oplus z)=a\oplus y$$

性质 4 记 $\text{ch}(x,y,z)=a$，$\text{ch}(1\oplus x,y,z)=b$，则

$$a\oplus b=y\oplus z$$

$$(a\oplus b)x=a\oplus z$$

性质 5 记 $\text{maj}(x,y,z)=a$，则

$$a=(x\oplus y)(x\oplus z)\oplus x=$$

$$(y\oplus z)(x\oplus y)\oplus y$$

性质 1～性质 5 的意义有以下几个方面。

1) 给出了函数 maj(·) 和 ch(·) 特定形式输入差分所对应的输出差分的形式。例如，对于函数 $\text{maj}(x,y,z)$，若输入差分 $\Delta x=1$，$\Delta y=\Delta z=0$，由性质 1 可知，对应的输出差分为 $y\oplus z$。

2) 给出了由输出构建关于输入的线性方程的方法。例如，对于函数 $\text{ch}(x,y,z)$，若已知输出 a 和 b，并且 $\Delta x=1$，$\Delta y=\Delta z=0$，则由性质 4 可以建立关于 (x,y,z) 的 2 个线性方程，分别为 $y\oplus z=a\oplus b$

表 3 控制比特取值与输入比特对应关系

参数	初始化															关联数据处理			加密			认证码生成
m_t	$\boldsymbol{K}$	$\boldsymbol{N}$	$\boldsymbol{K}'$	$\boldsymbol{K}$	$\boldsymbol{K}$	$\boldsymbol{K}$	$\boldsymbol{K}$	$\boldsymbol{K}$	$\boldsymbol{K}$	$\boldsymbol{K}$	$\boldsymbol{K}$	$\boldsymbol{K}$	$\boldsymbol{K}$	$\boldsymbol{K}$	$\boldsymbol{K}$	AD	10…0	00…0	P	10…0	00…0	00…0
a_t	1															1		0	1		0	1
b_t	1															1			0			1

和 $(a \oplus b)x = a \oplus z$。

3) 给出了函数 maj(·) 的另外 2 种表示形式及其线性化的方法，主要用于第 4 部分由猜测确定的方式来构建和提取线性方程。

3.2 ACORN v3 状态差分传播规律

由前面算法介绍可知，在 ACORN v3 的加密过程中，每拍生成的密钥比特不进行反馈，状态更新变换根据输入的明文比特对内部状态进行更新。此时在选择明文攻击条件下，攻击者可以通过控制明文输入来影响内部状态，从而得到对其攻击有利的内部状态和密钥流，进而对内部状态或密钥进行攻击。下面，假设攻击者可以控制明文输入比特的差分，对 ACORN v3 加密过程的内部状态差分传播情况进行分析。

设初始内部状态差分 $\Delta \boldsymbol{S}^0 = (0,0,\cdots,0)$，明文输入比特差分 Δm_t 满足

$$\Delta m_t = \begin{cases} 1, & 0 \leqslant t \leqslant 48 \\ 0, & 49 \leqslant t \leqslant 147 \end{cases} \tag{1}$$

则内部状态差分 $\Delta \boldsymbol{S}^t$ 随时刻 t 传播变化情况如下。

1) 当 $1 \leqslant t \leqslant 49$ 时，由于内部状态差分 Δs_i^{t-1} 恒为 0，$0 \leqslant i \leqslant 244$，因此更新比特生成函数 F 的输入差分为 0，相应的更新比特的差分也为 0，此时内部状态差分 Δs_{292}^t 的取值由输入比特差分 Δm_{t-1} 决定，即 $\Delta s_{292}^t = \Delta m_{t-1}$；由状态更新变换易得 $\Delta s_i^t = \Delta s_{i+1}^{t-1}, 0 \leqslant i \leqslant 291$，因此 $\Delta \boldsymbol{S}^t$ 具有以下形式。

$$\Delta s_i^t = \begin{cases} 0, & 0 \leqslant i \leqslant 292 - t \\ 1, & 293 - t \leqslant i \leqslant 292 \end{cases} \tag{2}$$

2) 当 $50 \leqslant t \leqslant 58$ 时，由于内部状态差分 $\Delta s_i^{t-1} = 0$，$i \in \{0,23,61,66,107,160,196\}$，并且 $\Delta s_{244}^{t-1} = 1$，而输入比特差分 $\Delta m_{t-1} = 0$，此时，由状态更新变换和性质 4 有

$$\Delta s_{292}^t = s_{23}^{t-1} \oplus s_{160}^{t-1},\ \Delta s_i^t = \Delta s_{i+1}^{t-1}, 0 \leqslant i \leqslant 291$$

为记号方便，令 $\Delta s_{292}^t = \alpha_{t-50}$，此时，$\Delta \boldsymbol{S}^t$ 的形式如下。

$$\Delta s_i^t = \begin{cases} 0, & 0 \leqslant i \leqslant 292 - t \\ 1, & 293 - t \leqslant i \leqslant 341 - t \\ \alpha_{t+i-342}, & 342 - t \leqslant i \leqslant 292 \end{cases} \tag{3}$$

3) 当 $59 \leqslant t \leqslant 63$ 时，由于内部状态差分 $\Delta s_i^{t-1} = 0$，$i \in \{0,23,61,66,107,160,196\}$，并且 $\Delta s_{244}^{t-1} = 1$，与 2)同理可得 $\Delta s_{292}^t = s_{23}^{t-1} \oplus s_{160}^{t-1} = \alpha_{t-50}$；另外，$\Delta s_{235}^{t-1} = 1$，$\Delta s_{230}^{t-1} = 0$，$\Delta s_{289}^{t-1} = \alpha_{t-54}$，状态更新变换为

$$\Delta s_{288}^t = \Delta s_{289}^{t-1} \oplus \Delta s_{235}^{t-1} \oplus \Delta s_{230}^{t-1} = 1 \oplus \alpha_{t-54}$$
$$\Delta s_i^t = \Delta s_{i+1}^{t-1}, i \notin \{288, 292\}$$

此时，$\Delta \boldsymbol{S}^t$ 的形式如下。

$$\Delta s_i^t = \begin{cases} 0, & 0 \leqslant i \leqslant 292 - t \\ 1, & 293 - t \leqslant i \leqslant 341 - t \\ \alpha_{t+i-342}, & 342 - t \leqslant i \leqslant 346 - t \\ \alpha_{t+i-342} \oplus 1, & 347 - t \leqslant i \leqslant 288 \\ \alpha_{t+i-342}, & 289 \leqslant i \leqslant 292 \end{cases} \tag{4}$$

图 3 为 $1 \leqslant t \leqslant 63$ 时部分内部状态差分传播示意。

4) 当 $64 \leqslant t \leqslant 97$ 时，由于内部状态差分 $\Delta s_i^{t-1} = 0$，$i \in \{0,23,61,66,107,160,196\}$，并且 $\Delta s_{244}^{t-1} = \Delta s_{235}^{t-1} = \Delta s_{230}^{t-1} = 1$，$\Delta s_{289}^{t-1} = \alpha_{t-54}$，由状态更新变换有

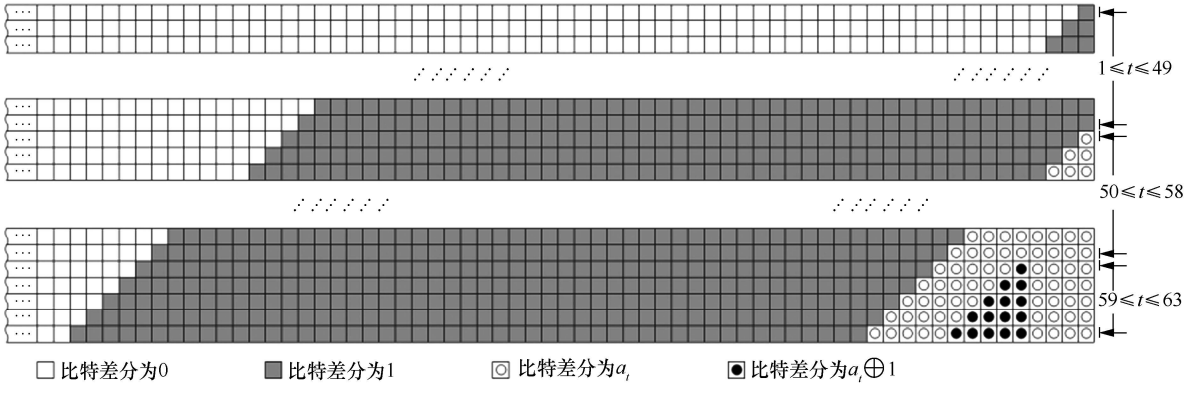


图 3　部分内部状态差分传播示意

$$\Delta s_{292}^{t}=s_{23}^{t-1}\oplus s_{160}^{t-1}=\alpha_{t-50}$$

$$\Delta s_{288}^{t}=\Delta s_{289}^{t-1}\oplus\Delta s_{235}^{t-1}\oplus\Delta s_{230}^{t-1}=\alpha_{t-54}$$

$$\Delta s_{i}^{t}=\Delta s_{i+1}^{t-1},i\notin\{288,292\}$$

此时，$\Delta\boldsymbol{S}^{t}$ 的形式如下。

$$\Delta s_{i}^{t}=\begin{cases}0, & 0\leqslant i\leqslant 292-t\\ 1, & 293-t\leqslant i\leqslant 341-t\\ \alpha_{t+i-342}, & 342-t\leqslant i\leqslant 346-t\\ \alpha_{t+i-342}\oplus 1, & 347-t\leqslant i\leqslant 351-t\\ \alpha_{t+i-342}, & 352-t\leqslant i\leqslant 292\end{cases}\tag{5}$$

当 $t\geqslant 98$ 时，Δs_{292}^{t} 的形式和取值对后面的分析不产生影响，因此，为简单起见，以下 Δs_{292}^{t} 的具体形式不再给出，仅给出对后续分析可能产生影响的部分状态差分。

5) 当 $98\leqslant t\leqslant 100$ 时，由于内部状态差分 $\Delta s_{235}^{t-1}=\Delta s_{230}^{t-1}=\Delta s_{196}^{t-1}=1$，$\Delta s_{289}^{t-1}=\alpha_{t-54}$，$\Delta s_{193}^{t-1}=0$，由状态更新变换有

$$\Delta s_{288}^{t}=\Delta s_{289}^{t-1}\oplus\Delta s_{235}^{t-1}\oplus\Delta s_{230}^{t-1}=\alpha_{t-54}$$

$$\Delta s_{229}^{t}=\Delta s_{230}^{t-1}\oplus\Delta s_{196}^{t-1}\oplus\Delta s_{193}^{t-1}=0$$

$$\Delta s_{i}^{t}=\Delta s_{i+1}^{t-1},0\leqslant i\leqslant 389-t,i\notin\{288,229\}$$

此时，$\Delta\boldsymbol{S}^{t}$ 的部分状态差分具有以下形式。

$$\Delta s_{i}^{t}=\begin{cases}0, & 0\leqslant i\leqslant 292-t\\ 1, & 293-t\leqslant i\leqslant 326-t\\ 0, & 327-t\leqslant i\leqslant 229\\ 1, & 230\leqslant i\leqslant 341-t\\ \alpha_{t+i-342}, & 342-t\leqslant i\leqslant 346-t\\ \alpha_{t+i-342}\oplus 1, & 347-t\leqslant i\leqslant 351-t\\ \alpha_{t+i-342}, & 352-t\leqslant i\leqslant 389-t\end{cases}\tag{6}$$

同理可以得出 $101\leqslant t\leqslant 148$ 时内部状态差分 $\Delta\boldsymbol{S}^{t}$ 的形式，详细推理过程这里不再给出，具体形式如附录（式(7)～式(14)）所示。以上分析说明，ACORN v3 算法加密过程的内部状态差分具有特定的传播规律。对于 Nonce 重用的情况，假设关联数据相同，由于使用了相同的密钥和 Nonce，因此加密起始时刻的内部状态相同，若加密的明文差分满足式(1)的形式，则上述状态差分传播规律同样成立，对此有以下结论。

命题 1 对于 ACORN v3 算法，假设用相同的密钥和 Nonce 分别对消息 $M_0=\mathrm{AD}\|P_0$ 和消息 $M_1=\mathrm{AD}\|P_1$ 进行处理，记加密过程起始时刻的内部状态差分为 $\Delta\boldsymbol{S}^{0}$，若明文 P_0 和 P_1 的前 49 bit 差分为 1，后 99 bit 差分为 0，其余位置差分不做要求，则当 $1\leqslant t\leqslant 148$ 时，内部状态差分 $\Delta\boldsymbol{S}^{t}$ 具有式(2)～式(14)的形式和传播规律。

4 ACORN v3 算法的状态恢复攻击

本节基于上文所述内部状态差分传播规律，利用差分代数技术，给出由密钥流差分通过猜测确定的方式建立关于内部状态线性方程的方法和具体过程，进而对 ACORN v3 算法的状态恢复攻击和复杂度进行评估。

4.1 密钥流生成函数特性分析

为了减少硬件开销，ACORN v3 算法采用了基于比特的二次布尔函数作为密钥流生成函数，且形式上比较简单，关于该函数本文给出以下性质。

性质 6 对于 ACORN v3 的密钥流生成函数，若已知 $k_t=G(\boldsymbol{S}^{t})$ 和 $k_t'=G(\boldsymbol{S}^{t}\oplus\Delta\boldsymbol{S}^{t})$，状态差分 $\Delta\boldsymbol{S}^{t}$ 满足 $(\Delta s_{235}^{t},\Delta s_{193}^{t},\Delta s_{230}^{t})\neq(0,0,0)$，$\Delta s_{i}^{t}=0$，$i\in\{12,61,111,66\}$，并且 Δs_{154}^{t} 的取值已知，那么通过猜测 1 bit，总可以得到关于 $\boldsymbol{S}^{t}$ 的 3 个线性方程。

证明 由 ACORN v3 密钥流生成函数和状态差分 $\Delta\boldsymbol{S}^{t}$ 的形式有

$$\Delta k_t=k_t\oplus k_t'=\Delta s_{154}^{t}\oplus$$
$$\Delta\mathrm{maj}(\Delta s_{235}^{t},0,\Delta s_{193}^{t})\oplus\Delta\mathrm{ch}(\Delta s_{230}^{t},0,0)$$

其中，

$$\Delta\mathrm{maj}(\Delta s_{235}^{t},0,\Delta s_{193}^{t})=\mathrm{maj}(s_{235}^{t},s_{61}^{t},s_{193}^{t})\oplus$$
$$\mathrm{maj}(s_{235}^{t}\oplus\Delta s_{235}^{t},s_{61}^{t},s_{193}^{t}\oplus\Delta s_{193}^{t})$$

$$\Delta\mathrm{ch}(\Delta s_{230}^{t},0,0)=\mathrm{ch}(s_{230}^{t},s_{111}^{t},s_{66}^{t})\oplus$$
$$\mathrm{ch}(s_{230}^{t}\oplus\Delta s_{230}^{t},s_{111}^{t},s_{66}^{t})$$

设 $\Delta s_{154}^{t}=0$，由于 $(\Delta s_{235}^{t},\Delta s_{193}^{t},\Delta s_{230}^{t})\neq(0,0,0)$，分情况讨论如下。

1) 若 $(\Delta s_{235}^{t},\Delta s_{193}^{t},\Delta s_{230}^{t})=(1,0,0)$，根据性质 1，通过猜测 $s_{111}^{t}\oplus s_{66}^{t}=\beta^{t}$，可以得到关于 $\boldsymbol{S}^{t}$ 的 3 个线性方程，分别为

$$\beta^{t}=s_{111}^{t}\oplus s_{66}^{t}$$

$$\Delta k_t=s_{61}^{t}\oplus s_{193}^{t}$$

$$k_t=s_{12}^{t}\oplus s_{154}^{t}\oplus\Delta k_t(s_{235}^{t}\oplus s_{61}^{t})\oplus s_{61}^{t}\oplus\beta^{t}s_{230}^{t}\oplus s_{66}^{t}$$

2) 若 $(\Delta s_{235}^t, \Delta s_{193}^t, \Delta s_{230}^t) = (1,0,1)$，由性质 1 和性质 4，通过猜测 $s_{61}^t \oplus s_{193}^t = \beta^t$，可以得到关于 $\boldsymbol{S}^t$ 的 3 个线性方程，分别为

$$\beta^t = s_{61}^t \oplus s_{193}^t$$

$$\Delta k_t = s_{61}^t \oplus s_{193}^t \oplus s_{111}^t \oplus s_{66}^t$$

$$k_t = s_{12}^t \oplus s_{154}^t \oplus \beta^t (s_{235}^t \oplus s_{61}^t) \oplus s_{61}^t \oplus (\Delta k_t \oplus \beta^t) s_{230}^t \oplus s_{66}^t$$

3) 若 $(\Delta s_{235}^t, \Delta s_{193}^t, \Delta s_{230}^t) = (1,1,1)$，由性质 2 和性质 4，通过猜测 $s_{235}^t \oplus s_{193}^t = \beta^t$，可以得到关于 $\boldsymbol{S}^t$ 的 3 个线性方程，分别为

$$\beta^t = s_{235}^t \oplus s_{193}^t$$

$$\Delta k_t = 1 \oplus s_{235}^t \oplus s_{193}^t \oplus s_{111}^t \oplus s_{66}^t$$

$$k_t = s_{12}^t \oplus s_{154}^t \oplus \beta^t (s_{235}^t \oplus s_{61}^t) \oplus s_{235}^t \oplus (1 \oplus \Delta k_t \oplus \beta^t) s_{230}^t \oplus s_{66}^t$$

4) 若 $(\Delta s_{235}^t, \Delta s_{193}^t, \Delta s_{230}^t) = (0,1,1)$，由性质 3 和性质 4，通过猜测 $s_{235}^t \oplus s_{61}^t = \beta^t$，可以得到关于 $\boldsymbol{S}^t$ 的 3 个线性方程，分别为

$$\beta^t = s_{235}^t \oplus s_{61}^t$$

$$\Delta k_t = s_{235}^t \oplus s_{61}^t \oplus s_{111}^t \oplus s_{66}^t$$

$$k_t = s_{12}^t \oplus s_{154}^t \oplus \beta^t (s_{235}^t \oplus s_{193}^t) \oplus s_{235}^t \oplus (\Delta k_t \oplus \beta^t) s_{230}^t \oplus s_{66}^t$$

5) 若 $(\Delta s_{235}^t, \Delta s_{193}^t, \Delta s_{230}^t) = (0,1,0)$，由性质 3，通过猜测 $s_{111}^t \oplus s_{66}^t = \beta^t$，可以得到关于 $\boldsymbol{S}^t$ 的 3 个线性方程，分别为

$$\beta^t = s_{111}^t \oplus s_{66}^t$$

$$\Delta k_t = s_{235}^t \oplus s_{61}^t$$

$$k_t = s_{12}^t \oplus s_{154}^t \oplus \Delta k_t (s_{235}^t \oplus s_{193}^t) \oplus s_{235}^t \oplus \beta^t s_{230}^t \oplus s_{66}^t$$

6) 若 $(\Delta s_{235}^t, \Delta s_{193}^t, \Delta s_{230}^t) = (1,1,0)$，由性质 2，通过猜测 $s_{111}^t \oplus s_{66}^t = \beta^t$，可以得到关于 $\boldsymbol{S}^t$ 的 3 个线性方程，分别为

$$\beta^t = s_{111}^t \oplus s_{66}^t$$

$$\Delta k_t = 1 \oplus s_{235}^t \oplus s_{193}^t$$

$$k_t = s_{12}^t \oplus s_{154}^t \oplus (1 \oplus \Delta k_t)(s_{235}^t \oplus s_{61}^t) \oplus s_{235}^t \oplus \beta^t s_{230}^t \oplus s_{66}^t$$

7) 若 $(\Delta s_{235}^t, \Delta s_{193}^t, \Delta s_{230}^t) = (0,0,1)$，由性质 4 和性质 5，则通过猜测 $s_{61}^t \oplus s_{193}^t = \beta^t$，可以得到关于 $\boldsymbol{S}^t$ 的 3 个线性方程，分别为

$$\beta^t = s_{61}^t \oplus s_{193}^t$$

$$\Delta k_t = s_{111}^t \oplus s_{66}^t$$

$$k_t = s_{12}^t \oplus s_{154}^t \oplus \beta^t (s_{235}^t \oplus s_{61}^t) \oplus s_{61}^t \oplus \Delta k_t s_{230}^t \oplus s_{66}^t$$

对于 $\Delta s_{154}^t = 1$ 可类似的证明，综上可得性质 6 成立。证毕。

上述证明中，对于每一种情况的猜测方式可能是不唯一的。例如，4) 中也可以通过猜测 $s_{111}^t \oplus s_{66}^t = \beta^t$ 来构建线性方程，这里仅给出其中的一种，更多的不再列出。性质 6 同时给出了通过猜测确定方式，由密钥流及其差分建立关于内部状态线性方程的基本思想和方法，具体如下。若已知输出的密钥流和经线性更新后的内部状态差分，假设内部状态差分的第 235、193、230 分位的不全为 0，并且第 12、61、111、66 分位的差分均为 0，由性质 6 可知，此时进行 1 bit 猜测，可以得到关于内部状态的 3 个线性方程。基于此，下面给出 Nonce 重用两次的 ACORN v3 状态恢复攻击。

4.2　Nonce 重用两次的 ACORN v3 状态恢复攻击

对于 ACORN v3 算法，假设用相同的 Nonce 分别对消息 $M_0 = \mathrm{AD} \| P_0$ 和 $M_1 = \mathrm{AD} \| P_1$ 进行处理，加密过程起始时刻的内部状态分别为 $\boldsymbol{S}_0^0$ 和 $\boldsymbol{S}_1^0$，生成的密钥流分别为 $k_{0,t}$ 和 $k_{1,t}$，$t \geqslant 0$，并且 P_0 和 P_1 满足

$$P_0 \oplus P_1 = \overbrace{1,1,\cdots,1}^{49}, \overbrace{0,0,\cdots,0}^{99}, \overbrace{*,*,\cdots,*}^{l}$$

由结论 1 可知，内部状态差分 $\Delta \boldsymbol{S}^t$ 具有并满足第 3 节给出的形式和传播规律，下面给出当 $0 \leqslant t \leqslant 147$ 时，由密钥流和状态差分构建内部状态线性方程进行状态恢复攻击的算法，并假设 $\boldsymbol{S}_0^0 = (s_{0,0}^0, s_{0,1}^0, \cdots, s_{0,292}^0)$ 是待恢复的内部状态。下文在不引起混淆的情况下，将 $s_{0,i}^t$ 记作 s_i^t，$0 \leqslant i \leqslant 292$。由 ACORN v3 状态更新变换的定义，密钥流生成函数和更新比特生成函数根据线性更新后的状态 $\boldsymbol{S}^t$ 进行密钥比特和更新比特的计算，并且除第 292 bit 外，状态 $\boldsymbol{S}^{t+1}$ 的其余比特位由 $\boldsymbol{S}^t$ 平移得到，即 $\boldsymbol{S}_{i-1}^{t+1} = \boldsymbol{S}_i^t, 1 \leqslant i \leqslant 292$，为描述方便，将加密过程的密钥流生成函数和更新比特生成函数等价表示为以下形式。

$$k_t = s_{11}^{t+1} \oplus s_{153}^{t+1} \oplus \mathrm{maj}(s_{234}^{t+1}, s_{60}^{t+1}, s_{192}^{t+1}) \oplus \mathrm{ch}(s_{229}^{t+1}, s_{110}^{t+1}, s_{65}^{t+1})$$

$$f_t = 1 \oplus s_0^t \oplus s_{106}^{t+1} \oplus \mathrm{maj}(s_{243}^{t+1}, s_{22}^{t+1}, s_{159}^{t+1}) \oplus s_{195}^{t+1}$$

其中，

$$s_{229}^{t+1} = s_{230}^t \oplus s_{196}^t \oplus s_{193}^t$$

$$s_{192}^{t+1} = s_{193}^t \oplus s_{160}^t \oplus s_{154}^t$$

$$s_{153}^{t+1} = s_{154}^t \oplus s_{111}^t \oplus s_{107}^t$$

$$s_{106}^{t+1} = s_{107}^t \oplus s_{66}^t \oplus s_{61}^t$$

$$s_{60}^{t+1} = s_{61}^t \oplus s_{23}^t \oplus s_0^t$$

结合内部状态差分 $\Delta \boldsymbol{S}^t$ 的形式和传播规律，基于输出的密钥流，针对 ACORN v3 的状态恢复攻击算法框架如算法 1 所示。

算法 1 ACORN v3 的状态恢复攻击

输入 $k_{0,t}$ 和 $k_{1,t}$，$0 \leqslant t \leqslant 147 + l$

输出 加密过程初始状态 $\boldsymbol{S}_0^0$

Step1 由 $k_{0,t}$ 和 $k_{1,t}$ 计算密钥流差分 Δk_t；

Step2 当 $0 \leqslant t \leqslant 147$ 时，根据内部状态差分的形式，通过猜测确定的方式由密钥流及其差分建立关于内部状态的线性方程，基于得到的线性方程构建关于内部状态 $\boldsymbol{S}_0^{49}$ 的线性方程系统；

Step3 对以上线性方程系统进行求解，并将每个解作为 $\boldsymbol{S}_0^{49}$ 的候选值；

Step4 由 $k_{0,t}$ 对这些候选值进行验证和筛选并得到 $\boldsymbol{S}_0^{49}$ 的正确取值；

Step5 由密钥流 $k_{0,t}$ 和明文对 $\boldsymbol{S}_0^{49}$ 进行逆状态更新变换，计算并得到加密初始状态 $\boldsymbol{S}_0^0$。

4.3 复杂度分析

算法 1 需要 $2(148+l)$ bit 的密钥流，其中 $l \geqslant 0$，其计算复杂度主要由 Step2 和 Step3 决定，Step1 和 Step5 的计算复杂度可忽略不计，假设 Step2 需要进行 n bit 猜测，Step3 中求解 293 bit 变元线性方程组的复杂度记为 c，并假设 Step4 中候选值的个数相对于 n bit 的猜测是可忽略的，则攻击算法 1 的计算复杂度约为 $2^n c$。下面对 Step2 需要进行猜测的比特个数进行估计。

当 $0 \leqslant t \leqslant 57$ 时，由于 $\boldsymbol{S}_0^0 = \boldsymbol{S}_1^0$，由密钥流生成函数易知，所生成的前 58 bit 密钥流 $k_{0,t}$ 和 $k_{1,t}$ 是相同的，此时由密钥流差分无法得到关于内部状态的线性方程，故猜测比特数为 0。

当 $58 \leqslant t \leqslant 106$ 时，由内部状态差分 $\Delta \boldsymbol{S}^{t+1}$ 的形式，根据性质 6，每个时刻进行 1 bit 猜测，可以得到关于内部状态的 3 个线性方程，总共进行 49 bit 猜测，可以得到 147 个线性方程。

当 $107 \leqslant t \leqslant 132$ 和 $134 \leqslant t \leqslant 136$ 时，由内部状态差分 $\Delta \boldsymbol{S}^{t+1}$ 的形式，根据性质 6 进行 1 bit 猜测，同时对 $\Delta s_{234}^{t+1} = s_{23}^{t-58} \oplus s_{160}^{t-58} = \alpha_{t-107}$ 进行猜测，每个时刻进行 2 bit 猜测可以得到关于内部状态的 4 个线性方程，总共进行 58 bit 猜测，可以得到 116 个线性方程。

当 $t = 133$ 以及 $137 \leqslant t \leqslant 138$ 时，内部状态差分 $\Delta \boldsymbol{S}^{t+1}$ 的第 11、153、60、110、65、192 比特取值均为 0，若 $\Delta s_{234}^{t+1} = \Delta s_{229}^{t+1} = 0$，则密钥流差分 $\Delta k_t = 0$，此时通过猜测 α_{t-107} 仅能得到一个线性方程；否则，通过猜测 α_{t-107} 及根据性质 6 进行额外的 1 bit 猜测，可以得到关于内部状态的 4 个线性方程，因此，每个时刻平均进行 $\frac{1}{4} + 2 \times \frac{3}{4} = 1.75$ bit 猜测，平均可以得到 $\frac{1}{4} + 4 \times \frac{3}{4} = 3.25$ 个线性方程，总共进行 5.25 bit 猜测，可以得到 9.75 个线性方程。

当 $139 \leqslant t \leqslant 140$ 时，由内部状态差分 $\Delta \boldsymbol{S}^{t+1}$ 的形式，根据性质 6 进行 1 bit 猜测，同时对 $\Delta s_{234}^{t+1} = s_{23}^{t-58} \oplus s_{160}^{t-58} = \alpha_{t-107}$ 进行猜测，每个时刻进行 2 bit 猜测可以得到关于内部状态的 4 个线性方程，总共进行 4 bit 猜测，可以得到的 8 个线性方程。

当 $141 \leqslant t \leqslant 147$ 时，由内部状态差分 $\Delta \boldsymbol{S}^{t+1}$ 的形式，对 $\Delta s_{234}^{t+1} = s_{23}^{t-58} \oplus s_{160}^{t-58} = \alpha_{t-107}$ 进行猜测，结合密钥流差分 Δk_t 可以得到关于内部状态的 2 个线性方程，总共进行 7 bit 猜测，可以得到 14 个线性方程。

基于猜测变量 α_{t-107}，以上所得关于内部状态的线性方程均可以表示为关于状态 $\boldsymbol{S}_0^{49}$ 的线性方程。

综上所述，如果 Step2 进行 123.25 bit 的猜测，平均可以得到 294.75 个关于内部状态的线性方程；如果 Step2 进行 122.25 bit 的猜测，则平均可以得到 292.75 个线性方程。

定理 1 当 Nonce 重用两次时，算法 1 给出的 ACORN v3 状态恢复攻击的数据复杂度为 $(2 \times 148 + 293)$ bit 的选择明文；计算复杂度为 $2^{122.5} c$，其中 c 是求解线性方程组的复杂度；存储复杂度可忽略不计。

证明 由以上分析可知，算法 1 中 $k_{0,t}$ 和 $k_{1,t}$ 的

前 148 bit 用于 Step2 线性方程的构建，并且 $k_{0,l}$ 的最后 l bit 用于 Step4 中对候选值进行验证和筛选，本文选择 l=293，因此算法 1 所需的数据复杂度为 $2\times148+293$ bit 的选择明文；Step1 和 Step5 的计算复杂度可忽略不计，记 Step3 中求解 293 bit 变元线性方程组的复杂度为 c，Step2 平均进行 123.25 bit 的猜测，可以得到 294.75 个关于内部状态的线性方程，假设线性无关的方程的个数是 293，则 Step3 有唯一解，此时 Step2 和 Step3 的计算复杂度约为 $2^{123.25}c$；如果 Step2 中进行 122.25 bit 的猜测，则平均可以得到 292.75 个线性方程，假设这些方程是线性无关的，则 Step2 和 Step3 的计算复杂度约为 $2^{122.5}c$；另外，假设对于 Step2 中错误的猜测在 Step3 中以很大概率无解，即 Step4 中候选值的个数远小于 Step2 的猜测量，则 Step4 的计算复杂度相对于 $2^{122.5}c$ 是可忽略的，综合可得，算法 1 的计算复杂度约为 $2^{122.5}c$，算法所需的存储复杂度可忽略不计。

证毕。

4.4　Nonce 多次重用的安全性分析

对 Nonce 重用三次、四次的情况，有以下分析结果。

对于 Nonce 重用三次的情况，假设 P_0、P_1 和 P_2 用相同的 Nonce 进行加密，为了尽可能使得到的方程是线性无关的，令 P_0、P_1、P_2 满足以下形式。

$$P_0 \oplus P_1 = \overbrace{1,1,\cdots,1}^{49},\overbrace{0,0,\cdots,0}^{l+49+n}$$

$$P_0 \oplus P_2 = \overbrace{0,0,\cdots,0}^{49+n},\overbrace{1,1,\cdots,1}^{49},\overbrace{0,0,\cdots,0}^{l}$$

其中，n 是需要猜测的变量 α_t 的个数，此时通过引入 49+n 个新增变量 $(s_{292}^{50},s_{292}^{51},\cdots,s_{292}^{98+n})$，可以得到 294+5$n$ 个线性方程和 49 个二次方程，由文献[7]可知，这 49 个二次方程能够以 $2^{20.3}$ 的复杂度进行线性化，对应得到 49 个线性方程，此时若要使方程有唯一解，需 294+5n+49≥293+49+n，即 $n\geqslant 0$，因此需要猜测的变量个数为 98，对应的状态恢复攻击的复杂度为 $2^{20.3}\times 2^{98}c=2^{118.3}c$。

对于 Nonce 重用四次的情况，根据上述分析，令需要猜测的变量 α_t 的个数为 0，需要猜测的变量 β_t 的个数为 n，此时通过引入 98 个新变量 $(s_{292}^{50},s_{292}^{51},\cdots,s_{292}^{147})$，可以得 147+2$n$ 个线性方程和 98 个二次方程，98 个二次方程能够以 $2^{40.6}$ 的复杂度进行线性化，对应得到 98 个线性方程，当 $n\geqslant 73$ 时，$147+2n+98\geqslant 293+98$ 恒成立，此时选择 P_0、P_1、P_2、P_3 形式如下。

$$P_0 \oplus P_1 = \overbrace{1,1,\cdots,1}^{49},\overbrace{0,0,\cdots,0}^{l+98}$$

$$P_0 \oplus P_2 = \overbrace{0,0,\cdots,0}^{49},\overbrace{1,1,\cdots,1}^{49},\overbrace{0,0,\cdots,0}^{l+49}$$

$$P_0 \oplus P_3 = \overbrace{0,0,\cdots,0}^{49},\overbrace{0,0,\cdots,0}^{49},\overbrace{1,1,\cdots,1}^{49},\overbrace{0,0,\cdots,0}^{l}$$

通过对 73 个变量 β_t 进行猜测和对 98 个二次方程的线性化，恰好可以得到 391 个线性方程，对应的状态恢复攻击的复杂度为 $2^{40.6}\times 2^{73}c=2^{113.6}c$。

类似地，可以对 Nonce 重用更多次的情况进行分析，状态恢复攻击复杂度估计如表 4 所示。

表 4　Nonce 多次重用的状态恢复攻击复杂度估计

Nonce 重用次数	数据复杂度/bit	计算复杂度
3	3×98+l	$2^{118.3}c$
4	4×147+l	$2^{113.6}c$
5	5×196+l	$2^{109.4}c$
6	6×245+l	$2^{105.2}c$
7	7×294+l	$2^{101.5}c$
8	8×343+l	$2^{101.5}c$

表 4 中可以选择 l=293。由表 4 的数据和结果可以看出，相对于之前的版本（ACORN v2 和 ACORN v1），由于 ACORN v3 采用了相对复杂的滤波函数，增加了冗余，使通过增加 Nonce 重用次数来大幅降低攻击复杂度的方法很难奏效，从而有效降低了 Nonce 多次重用时的安全风险。

4.5　基于已知初始状态的伪造攻击

由于 ACORN 算法采用了特定设计的序列密码结构，因此初始状态对算法整体安全性的影响至关重要，对于早期的 ACORN v1，由于其初始化过程是可逆的，一旦恢复初始状态，就可以利用初始化过程的逆变换逐步求解和恢复密钥，从而实现对密码算法的完全破解；对于 ACORN v2 和 ACORN v3，由于采用了不可逆的初始化过程，很难由初始状态来恢复和求解密钥，尽管如此，攻击者仍然可以利用所恢复的初始状态实现对任意消息的加密和认证标签的生成，从而进行伪造攻击。

5　结束语

本文分析了 ACORN v3 在 Nonce 重用两次情况下的安全性，结果表明，基于差分代数的方法，通

过对中间变量进行猜测来对 ACORN v3 的密钥流生成函数进行线性化，可以由密钥流及其差分构造足够多的关于内部状态的线性方程，进而通过求解线性方程组实现对 ACORN v3 的状态恢复攻击。尽管本文的分析结果远没到实际可行的程度，但进一步完善了 ACORN v3 算法 Nonce 重用条件下的安全性分析结果。另外，基于本文方法对 Nonce 多次重用情况的安全性进行的分析和评估发现，由于 ACORN v3 采用了较之前版本复杂的滤波函数，避免了通过增加 Nonce 重用次数来显著降低状态恢复攻击复杂度的潜在问题。最后，关于 ACORN 算法的可证明安全性研究是很必要的，可以从差分和线性指标的可控性等方面对算法的可证明安全性进行研究，这将是下一步研究工作的重点。

附录 $101 \leqslant t \leqslant 148$ 时内部状态差分 $\Delta \boldsymbol{S}^t$ 形式

$101 \leqslant t \leqslant 148$ 时内部状态差分 $\Delta \boldsymbol{S}^t$ 的形式如下。

1) 当 $101 \leqslant t \leqslant 131$ 时，内部状态差分 $\Delta \boldsymbol{S}^t$ 满足

$$\Delta s_i^t = \begin{cases} 0, & 0 \leqslant i \leqslant 292-t \\ 1, & 293-t \leqslant i \leqslant 326-t \\ 0, & 327-t \leqslant i \leqslant 329-t \\ 1, & 330-t \leqslant i \leqslant 341-t \\ \alpha_{t+i-342}, & 342-t \leqslant i \leqslant 346-t \\ \alpha_{t+i-342} \oplus 1, & 347-t \leqslant i \leqslant 351-t \\ \alpha_{t+i-342}, & 352-t \leqslant i \leqslant 375-t \end{cases} \tag{7}$$

2) 当 $132 \leqslant t \leqslant 133$ 时，内部状态差分 $\Delta \boldsymbol{S}^t$ 满足

$$\Delta s_i^t = \begin{cases} 0, & 0 \leqslant i \leqslant 292-t \\ 1, & 293-t \leqslant i \leqslant 326-t \\ 0, & 327-t \leqslant i \leqslant 329-t \\ 1, & 330-t \leqslant i \leqslant 341-t \\ \alpha_{t+i-342}, & 342-t \leqslant i \leqslant 346-t \\ \alpha_{t+i-342} \oplus 1, & 347-t \leqslant i \leqslant 351-t \\ \alpha_{t+i-342}, & 352-t \leqslant i \leqslant 360-t \\ \alpha_{t+i-342} \oplus 1, & 361-t \leqslant i \leqslant 229 \\ \alpha_{t+i-342}, & 230 \leqslant i \leqslant 377-t \end{cases} \tag{8}$$

3) 当 $t = 134$ 时，内部状态差分 $\Delta \boldsymbol{S}^t$ 满足

$$\Delta s_i^t = \begin{cases} 0, & 0 \leqslant i \leqslant 292-t \\ 1, & 293-t \leqslant i \leqslant 325-t \\ 0, & 326-t \leqslant i \leqslant 329-t \\ 1, & 330-t \leqslant i \leqslant 341-t \\ \alpha_{t+i-342}, & 342-t \leqslant i \leqslant 346-t \\ \alpha_{t+i-342} \oplus 1, & 347-t \leqslant i \leqslant 351-t \\ \alpha_{t+i-342}, & 352-t \leqslant i \leqslant 360-t \\ \alpha_{t+i-342} \oplus 1, & 361-t \leqslant i \leqslant 229 \\ \alpha_{t+i-342}, & 230 \leqslant i \leqslant 378-t \end{cases} \tag{9}$$

4) 当 $135 \leqslant t \leqslant 136$ 时，内部状态差分 $\Delta \boldsymbol{S}^t$ 满足

$$\Delta s_i^t = \begin{cases} 0, & 0 \leqslant i \leqslant 292-t \\ 1, & 293-t \leqslant i \leqslant 325-t \\ 0, & i = 326-t \\ 1, & 327-t \leqslant i \leqslant 192 \\ 0, & 193 \leqslant i \leqslant 329-t \\ 1, & 330-t \leqslant i \leqslant 341-t \\ \alpha_{t+i-342}, & 342-t \leqslant i \leqslant 346-t \\ \alpha_{t+i-342} \oplus 1, & 347-t \leqslant i \leqslant 351-t \\ \alpha_{t+i-342}, & 352-t \leqslant i \leqslant 360-t \\ \alpha_{t+i-342} \oplus 1, & 361-t \leqslant i \leqslant 229 \\ \alpha_{t+i-342}, & 230 \leqslant i \leqslant 380-t \end{cases} \tag{10}$$

5) 当 $t = 137$ 时，内部状态差分 $\Delta \boldsymbol{S}^t$ 满足

$$\Delta s_i^t = \begin{cases} 0, & 0 \leqslant i \leqslant 292-t \\ 1, & 293-t \leqslant i \leqslant 325-t \\ 0, & i = 326-t \\ 1, & 327-t \leqslant i \leqslant 341-t \\ \alpha_{t+i-342}, & 342-t \leqslant i \leqslant 346-t \\ \alpha_{t+i-342} \oplus 1, & 347-t \leqslant i \leqslant 351-t \\ \alpha_{t+i-342}, & 352-t \leqslant i \leqslant 360-t \\ \alpha_{t+i-342} \oplus 1, & 361-t \leqslant i \leqslant 229 \\ \alpha_{t+i-342}, & 230 \leqslant i \leqslant 381-t \end{cases} \tag{11}$$

6) 当 $138 \leqslant t \leqslant 139$ 时，内部状态差分 $\Delta \boldsymbol{S}^t$ 满足

$$\Delta s_i^t = \begin{cases} 0, & 0 \leqslant i \leqslant 292-t \\ 1, & 293-t \leqslant i \leqslant 325-t \\ 0, & i = 326-t \\ 1, & 327-t \leqslant i \leqslant 329-t \\ 0, & 330-t \leqslant i \leqslant 192 \\ 1, & 193 \leqslant i \leqslant 341-t \\ \alpha_{t+i-342}, & 342-t \leqslant i \leqslant 346-t \\ \alpha_{t+i-342} \oplus 1, & 347-t \leqslant i \leqslant 351-t \\ \alpha_{t+i-342}, & 352-t \leqslant i \leqslant 360-t \\ \alpha_{t+i-342} \oplus 1, & 361-t \leqslant i \leqslant 366-t \\ \alpha_{t+i-342}, & 367-t \leqslant i \leqslant 383-t \end{cases} \tag{12}$$

7) 当 $140 \leqslant t \leqslant 146$ 时，内部状态差分 $\Delta \boldsymbol{S}^t$ 满足

$$\Delta s_i^t = \begin{cases} 0, & 0 \leqslant i \leqslant 292-t \\ 1, & 293-t \leqslant i \leqslant 325-t \\ 0, & i = 326-t \\ 1, & 327-t \leqslant i \leqslant 329-t \\ 0, & 330-t \leqslant i \leqslant 331-t \\ 1, & 332-t \leqslant i \leqslant 341-t \\ \alpha_{t+i-342}, & 342-t \leqslant i \leqslant 346-t \\ \alpha_{t+i-342} \oplus 1, & 347-t \leqslant i \leqslant 351-t \\ \alpha_{t+i-342}, & 352-t \leqslant i \leqslant 360-t \\ \alpha_{t+i-342} \oplus 1, & 361-t \leqslant i \leqslant 366-t \\ \alpha_{t+i-342}, & 367-t \leqslant i \leqslant 390-t \end{cases} \tag{13}$$

8) 当$147 \leqslant t \leqslant 148$时，内部状态差分$\Delta \boldsymbol{S}^t$满足

$$\Delta s_i^t = \begin{cases} 0, & 0 \leqslant i \leqslant 292-t \\ 1, & 293-t \leqslant i \leqslant 325-t \\ 0, & i = 326-t \\ 1, & 327-t \leqslant i \leqslant 329-t \\ 0, & 330-t \leqslant i \leqslant 331-t \\ 1, & 332-t \leqslant i \leqslant 341-t \\ \alpha_{t+i-342}, & 342-t \leqslant i \leqslant 346-t \\ \alpha_{t+i-342} \oplus 1, & 347-t \leqslant i \leqslant 351-t \\ \alpha_{t+i-342}, & 352-t \leqslant i \leqslant 360-t \\ \alpha_{t+i-342} \oplus 1, & 361-t \leqslant i \leqslant 366-t \\ \alpha_{t+i-342}, & 367-t \leqslant i \leqslant 375-t \\ \alpha_{t+i-342} \oplus \alpha_{t+i-376} \oplus 1, & 376-t \leqslant i \leqslant 229 \\ \alpha_{t+i-342}, & 230 \leqslant i \leqslant 392-t \end{cases} \quad (14)$$

参考文献：

[1] WU H J. ACORN: a lightweight authenticated cipher (v1)[R]. CAESAR First Round Submission, 2014.

[2] WU H J. ACORN: a lightweight authenticated cipher (v2)[R]. CAESAR Second Round Submission, 2015.

[3] WU H J. ACORN: a lightweight authenticated cipher (v3)[R]. Candidate for the CAESAR Competition, 2016.

[4] LIU M C, LIN D D. Cryptanalysis of lightweight authenticated cipher acorn[R]. Crypto-Competition Mailing List, 2014.

[5] CHAIGNEAU C, FUHR T, GIBERT H. Full key-recovery on acorn in nonce-reuse and decryption-misuse settings[R].Crypto-Competition Mailing List, 2015.

[6] WANG S P, HU B, LIU Y, et al. Nonce-reuse attack on authenticated cipher ACORN[C]//2016 International Conference on Artificial Intelligence and Computer Science. Lancaster: DEStech Publication, 2016: 379-385.

[7] ZHANG X J, LIN D D. Cryptanalysis of ACORN in nonce-reuse setting[C]//13th International Conference on Information Security and Cryptology. Berlin: Springer, 2017: 342-361.

[8] SALAM M, BARTLETT H, DAWSON E, et al. Investigating cube attacks on the authenticated encryption stream cipher acorn[C]//2016 International Conference on Applications and Techniques in Information Security. Berlin: Springer, 2016: 15-26.

[9] SALAM M, WONG K, BARTLETT H, et al. Finding state collisions in the authenticated encryption stream cipher ACORN[C]//2016 Proceedings of the Australasian Computer Science Week Multiconference. New York: ACM Press, 2016, 36: 1-10.

[10] LAFITTE F, LERMAN L, MARKOWITCH O, et al. SAT-based cryptanalysis of ACORN[R]. IACR Cryptology ePrint Archive, Report 2016/521, 2016.

[11] DWIVEDI A D, KLOUČEK M, MORAWIECKI P. SAT-based cryptanalysis of authenticated ciphers from the CAESAR competetion[C]//2017 The 14th International Conference on Security and Cryptography. Berlin: Springer, 2017: 275-284.

[12] DIBYENDU R, SOURAV M. Some results on ACORN[R]. IACR Cryptology ePrint Archive, Report 2016/1132, 2016.

[13] TODO Y, ISOBE T, HAO Y L, et al. Cube attacks on non-blackbox polynomials based on division property[C]//2017 37th Annual International Cryptology Conference. Berlin: Springer, 2017: 250-279.

[14] SIDDHANTI A A, MAITRA S, SINHA N. Certain observations on acorn v3 and the implications to TMDTO attacks[C]//2017 7th International Conference on Security, Privacy, and Applied Cryptography Engineering. Berlin: Springer, 2017: 264-280.

[15] GHAFARI V A, HU H G. A new chosen IV statistical distinguishing framework to attack symmetric ciphers, and its application to ACORN-v3 and Grain-128a[J]. Journal of Ambient Intelligence and Humanized Computing, 2019, 10: 2393-2400.

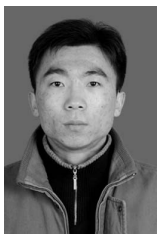
[16] WANG Q J, HAO Y L, TODO Y, et al. Improved division property based cube attacks exploiting algebraic properties of superpoly[C]//2018 38th Annual International Cryptology Conference. Berlin: Springer, 2018: 275-305.

[17] ZHANG F, LIANG Z Y, YANG B L, et al. Survey of design and security evaluation of authenticated encryption algorithms in the CAESAR competition[J]. Frontiers of Information Technology & Electronic Engineering, 2018, 19(12): 1475-1499.

[18] YANG J C, LIU M C, LIN D D. Cube cryptanalysis of round-reduced ACORN[C]//2019 22nd International Conference on Information Security. Berlin: Springer, 2019: 44-64.

[19] KESARWANI A, ROY D, SARKAR S, et al. New cube distinguishers on NFSR-based stream ciphers[J]. Design, Codes and Cryptography, 2020, 88: 173-199.

[作者简介]



张国双（1982− ），男，河北临城人，中国科学院信息工程研究所博士生，主要研究方向为密码理论、认证加密算法设计与分析等。

陈晓（1968− ），女，浙江杭州人，博士，中国科学院信息工程研究所研究员、博士生导师，主要研究方向为信息安全。

林东岱（1964− ），男，山东聊城人，博士，中国科学院信息工程研究所研究员、博士生导师，主要研究方向为密码理论、安全协议、网络空间安全等。

刘凤梅（1973− ），女，河南郸城人，博士，信息保障技术重点实验室研究员，主要研究方向为密码理论与应用。