约减轮数轻量级密码 PFP 的密钥恢复分析*

刘　健[1,2]，张　岩[1]，黄丁韫[3]，王伊婷[1]，章　涛[1]

（1. 中国电子科技集团公司第十五研究所　信息产业信息安全测评中心，北京　100083；
2. 清华大学 网络科学与网络空间研究院，北京　100084；3. 北京电子科技学院 密码科学与技术系，北京　100070）

摘　要： PFP 算法是 2017 年提出的一种借鉴国际标准 PRESENT 算法设计的轻量级分组密码。它基于 Feistel-SP 结构设计，采用比特置换技术，在软硬件实现效率方面较 PRESENT 算法更高。为评估其抗差分分析能力，基于已提出的 25 轮区分器，在区分器之前增加 1 轮，之后增加 2 轮，形成 28 轮的结构。通过分析新增 3 轮的结构特点，构造符合区分器的明文结构，优化密钥猜测顺序；并利用提前抛弃技术，首次实现了对 PFP 算法 28 轮的密钥恢复，比现有的最高攻击轮数 27 轮多 1 轮。整个攻击的过程需要 2^{63} 个明文的数据量，时间复杂度约为 $2^{57.2}$ 次 28 轮加密，与整体 34 轮相比，还剩 17.4% 的轮数（6 轮）作为安全冗余，这表明目前 PFP 算法仍然有足够的安全性。

关键词： 轻量级分组密码；差分攻击；明文结构；密钥恢复；提前抛弃技术

中图分类号： TP309.7　　**文献标识码：** A　　**DOI：** 10.19358/j.issn.2097-1788.2025.10.006

引用格式： 刘健，张岩，黄丁韫，等. 约减轮数轻量级密码 PFP 的密钥恢复分析［J］. 网络安全与数据治理，2025，44(10)：35-39.

Key-recovery analysis of reduced-round lightweight ciphers PFP

Liu Jian[1,2], Zhang Yan[1], Huang Dingyun[3], Wang Yiting[1], Zhang Tao[1]

(1. Information Industry Information Security Evaluation Center, The 15th Research Institute of China Electronics Technology Group Corporation, Beijing 100083, China; 2. Institute for Network Sciences and Cyberspace, Tsinghua University, Beijing 100084, China; 3. Department of Cryptographic Science and Technology, Beijing Electronic Science and Technology Institute, Beijing 100070, China)

Abstract: The PFP algorithm is a lightweight block cipher proposed in 2017 that draws design inspiration from the international standard PRESENT algorithm, featuring a Feistel-SP structure with bit permutation techniques to achieve higher software/hardware implementation efficiency than PRESENT. To evaluate its differential cryptanalysis resistance, this study extends a known 25-round distinguisher by adding 1 round before and 2 rounds after to construct a 28-round attack framework. Through analyzing the structural characteristics of these additional 3 rounds, we developed conforming plaintext structures, optimized key-guessing sequences, and employed early-abort techniques to achieve 28-round key-recovery attack on PFP for the first time, surpassing the previous 27-round record. The attack requires 2^{63} plaintexts with computational complexity of approximately $2^{57.2}$ 28-round encryptions. Compared to the overall 34 rounds, there is still 17.4% (6 rounds) of safety redundancy, demonstrating PFP's current sufficient security level.

Key words: lightweight block cipher; differential attack; plaintext structure; key recovery; early-abort technique

0　引言

在当今数字化时代，随着物联网和嵌入式设备的广泛应用[1]，如何在体积小、能耗低、软硬件计算资源受限[2]等环境中确保数据安全性，已经成为密码研究的一个重要课题。轻量级分组密码算法以其高效的加密速度和较低的资源占用，逐渐成为当今密码领域的研究热点。这些算法不仅在物联网中得到了广泛应用，还延伸至 5G/6G 通信、智能医疗、车联网等高安全性和高实时性场景，进一步凸显了其重要性和应用价值。

*基金项目：装备数据安全与保障技术教育部重点实验室资助项目（2024010102）

近年来，众多轻量级密码算法相继被提出，如 PRESENT、LBlock、ZORRO、PFP 等[3-6]。与传统的分组密码算法相比，这些算法在设计时通常会降低复杂度以适应资源受限的环境，这可能导致其安全性下降，从而增加被攻击的风险。因此，对轻量级分组密码算法进行系统的安全性分析显得尤为重要。分析成果既可以为密码算法的应用提供参考，也可以为后续设计者提供相应的参考。对于分组密码，目前有效的分析方法包括差分分析[7]、线性分析[8]、不可能差分分析[9]、中间相遇攻击、积分分析等。其中，差分分析最早是由 Biham 等人[10]于 1991 年针对 DES 提出的，这种攻击方法本质上是寻找密码算法中的高概率差分特征以此构成相应差分路径来恢复密钥，并且其对具有迭代差分属性的分组密码是十分高效的。近年来，随着计算技术的发展，许多基于数学工具的自动化搜索最优解技术逐渐成熟，如基于混合整数线性规划（MILP）、基于布尔可满足性问题（SAT/SMT）求解等。

在轻量级分组密码中，PFP 算法是 2017 年黄玉划等人[6]提出的一种基于 Feistel-SP 结构的轻量级分组密码，其设计借鉴了国际标准 PRESENT 算法，但在软硬件实现效率上超越了 PRESENT 算法。在设计者提出 PFP 算法之初，便通过差分分析、不可能差分分析、线性分析等方法对该算法进行了安全性评估。其中，对于差分分析设计者指出加密 15 轮时，PFP 至少有 53 个活跃的 S 盒，并以此计算 PFP 算法的 15 轮差分概率为 2^{-106}，从而推断该算法没有明显的 15 轮差分特征；并且设计者也用他们找到的 5 轮不可能差分区分器，进行了 6 轮的不可能差分攻击。然而，2020 年沈璇等人[9]找到了 7 轮不可能差分区分器，并进行了 9 轮的攻击；2023 年李艳俊等人[11]找到 PFP 算法的 4 轮迭代差分路径，从而构造了 22 轮的区分器，并实现了 26 轮的密钥恢复攻击。2024 年陆金玉等人[12]建立了 PFP 算法的 SMT 模型，找到了 20 条概率为 2^{-10}的 4 轮迭代差分，并以此构造了 25 轮的差分路径，基于该差分路径实现了 27 轮的密钥恢复攻击。

本文在文献［12］构造的 25 轮区分器基础上，前面增加 1 轮，后面增加 2 轮，形成 28 轮简化的加密算法。通过分析新增 3 轮的结构特点，进行了明文结构构造，并利用提前抛弃技术优化密钥猜测过程，首次实现了对 PFP 算法 28 轮的密钥恢复，整个攻击的过程需要 2^{63}个明文的数据量，时间复杂度约为 $2^{57.2}$次 28 轮加密。表 1 给出了 PFP 算法现有攻击结果比较。

表 1　PFP 算法攻击结果比较

来源	攻击方法	攻击轮数	恢复密钥量/比特	时间复杂度	数据复杂度
文献［9］	不可能差分分析	9	36	2^{58}	2^{36}
文献［13］	积分分析	12	40	$2^{63.12}$	$2^{62.39}$
文献［14］	相关密钥差分分析	34	80	2^{48}	2^{63}
文献［11］	差分分析	26	48	$2^{54.3}$	2^{60}
文献［12］	差分分析	27	40	$2^{75.25}$	2^{43}
本文	差分分析	28	48	$2^{57.2}$	2^{63}

1　PFP 算法介绍

PFP 算法是一种典型 Feistel-SP 结构的轻量级分组算法，该算法一共迭代 34 轮，其明密文长度均为 64 比特，密钥长度为 80 比特。

1.1　常用符号与术语

本文主要用到以下符号：

M：明文，$M \in F_2^{64}$；

C：密文，$C \in F_2^{64}$；

K_i：第 $i+1$ 轮子密钥，$K \in F_2^{32}$；

L_i：第 $i+1$ 轮输入的左支，$L \in F_2^{32}$；

R_i：第 $i+1$ 轮输入的右支，$R \in F_2^{32}$；

$\oplus$：异或运算；

$\|$：连接符，将左右两边的数据连接起来；

$K_{i,j}$：第 i 轮子密钥的第 j 个半字节。

1.2　加密变换

将输入的 64 比特的分组明文 M 分为长度为 32 比特的左右两分支 L_0 和 R_0，即 $M = L_0 \| R_0$。对 $i = 0, 1, 2, \cdots, 33$，进行如下第 i 轮变换计算：

$$L_{i+1} = R_i$$

$$R_{i+1} = L_i \oplus F(R_i, K_i)$$

迭代 34 轮后输出密文 $C = L_{34} \| R_{34}$。加密流程如图 1 所示，最后一轮不交换。

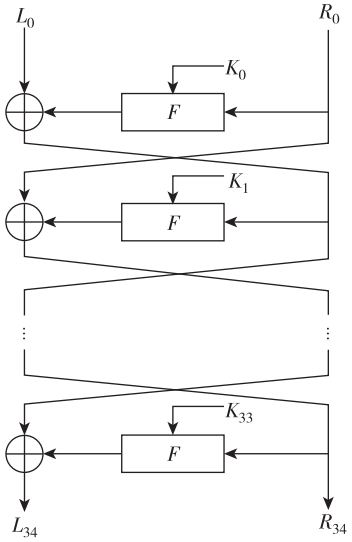


图 1　PFP 算法整体结构示意图

1.3 轮函数设计

PFP 算法的轮函数 F 为 SP 结构，包括轮密钥加、S 盒以及 P 置换，如图 2 所示。

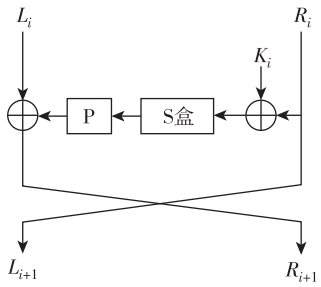


图 2 PFP 算法轮函数图

（1）轮密钥加：将右分支 R_i 与第 i 轮的子密钥 K_i 进行按位异或，即对 32 比特 $R_i = b_{31}\cdots b_0$，32 比特 K_i，进行运算：$R' = R_i \oplus K_i$。

（2）S 盒置换：PFP 算法使用了 PRESENT 算法的 S 盒，S 盒置换层一共由 8 个相同的 S 盒构成，该 S 盒是 4×4 位的，即：$F_2^4 \rightarrow F_2^4$，如表 2 所示。

表 2 PFP 算法的 S 盒

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S(x)$	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

（3）P 置换：PFP 算法的比特扩散层，采用位排列置换，数据经过置换后将第 i 比特移动到第 $P(i)$ 比特，具体比特位置变化如表 3 所示。

表 3 PFP 算法 P 置换

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$P(i)$	0	8	16	24	1	9	17	25	2	10	18	26	3	11	19	27
i	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
$P(i)$	4	12	20	28	5	13	21	29	6	14	22	30	7	15	23	31

1.4 密钥编排

密钥寄存器中存储用户主密钥 K，用 $k_{79}k_{78}\cdots k_1k_0$ 表示，每轮只取 K 中最左边 32 比特作为本轮子密钥。因此在第 i 轮中，有：$K_i = k_{31}^i\cdots k_0^i = k_{79}\cdots k_{48}$，表示取密钥寄存器中的前 32 比特作为 K_i。在提取了轮密钥 K_i 之后，密钥寄存器 $K_i = k_{79}k_{78}\cdots k_0$ 按如下方式进行更新，其中 $S[K]$ 表示对 4 比特密钥进行 S 盒置换，rc 以当前轮数作为轮常数：

（1）$k_{79}k_{78}\cdots k_1k_0 = k_{18}k_{17}\cdots k_{20}k_{19}$；

（2）$k_{79}k_{78}k_{77}k_{76} = S[k_{79}k_{78}k_{77}k_{76}]$；

（3）$k_{19}k_{18}k_{17}k_{16}k_{15} = [k_{19}k_{18}k_{17}k_{16}k_{15}] \oplus \mathrm{rc}$。

2 PFP 算法的 28 轮密钥恢复

2.1 PFP 的差分路径

本文将基于陆金玉等人找到的由 4 轮迭代差分构成的概率为 2^{-61} 的 25 轮差分区分器进行 28 轮的密钥恢复过程。所选取的差分特征如表 4 所示，表中差分均为十六进制表示。

表 4 PFP 算法 25 轮差分特征

轮数	ΔL	ΔR	$\log_2 p$
0	00040d00	00000900	−2
1	00000900	00000d00	−2
2	00000d00	00000d00	−2
3	00000d00	00000900	−3
4	00000900	00000900	−3
5	00000900	00000d00	−2
6	00000d00	00000d00	−2
7	00000d00	00000900	−3
8	00000900	00000900	−3
9	00000900	00000d00	−2
10	00000d00	00000d00	−2
11	00000d00	00000900	−3
12	00000900	00000900	−3
13	00000900	00000d00	−2
14	00000d00	00000d00	−2
15	00000d00	00000900	−3
16	00000900	00000900	−3
17	00000900	00000d00	−2
18	00000d00	00000d00	−2
19	00000d00	00000900	−3
20	00000900	00000900	−3
21	00000900	00000d00	−2
22	00000d00	00000d00	−2
23	00000d00	00000900	−3
24	00000900	00000900	−3
25	00000900	00040d00	
$\sum_{r=0}^{24}\log_2 p$			−61

2.2 密钥恢复步骤

上节 25 轮差分特征可以构成一个区分器，在区分器的前后分别添加 1 轮、2 轮进行 28 轮的密钥恢复分析，如图 3所示，左/右输入差分为：

$\Delta L_0 = 00000900 \oplus P$（000 * 0 * 00）

$\Delta R_0 = 00040\mathrm{d}00$

其中“*”表示任意非零半字节，“?”表示未知字节。

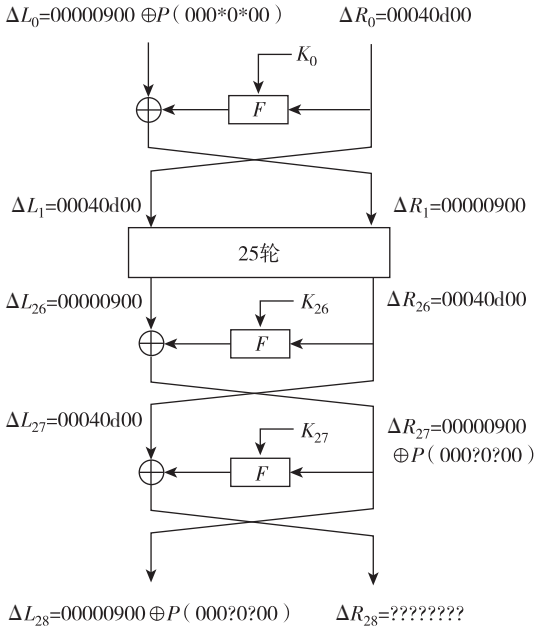


图 3　28 轮密钥恢复攻击

步骤 1　构造 2^n 个明文结构，每个结构的明文含两组数据，第一组为：

$L_0=(x_7,\ x_6,\ x_5,\ x_4,\ x_3,\ x_2,\ x_1,\ x_0)\oplus P\ (t_7t_6t_5*t_3*t_1t_0)$

$R_0=(y_7,\ y_6,\ y_5,\ y_4,\ y_3,\ y_2,\ y_1,\ y_0)$

第二组为：

$L_0'=(x_7,\ x_6,\ x_5,\ x_4,\ x_3,\ x_2\oplus 9,\ x_1,\ x_0)\oplus P\ (t_7t_6t_5*t_3*t_1t_0)$

$R_0'=(y_7,\ y_6,\ y_5,\ y_4\oplus 4,\ y_3,\ y_2\oplus \mathrm{d},\ y_1,\ y_0)$

其中 * 为遍历 4 比特取值，每个结构有 2×2^8 个明文，加密所有明文 28 轮得到对应密文 C。

对每个结构中两组明 - 密文进行两两组对并过滤。这些明 - 密文共组成 2^{n+15} 个对，若 $\triangle L_{28}$ 满足

$$00000900\oplus P(000*0*00)$$

则留下，不满足则抛弃。最后剩余对的数量为：$2^{n+15}\times 2^{-24}=2^{n-9}$。

步骤 2　猜测第 28 轮的 32 比特子密钥 K_{27}，进行第 28 轮解密，共需要 $2^{n-9}\times 2^{32}=2^{n+21}$ 次一轮运算。

对密文对进行过滤，剩余 $2^{n-9}\times 2^{-32}=2^{n-41}$ 对。

步骤 3　猜测第 27 轮的 8 比特子密钥 $K_{26,\{2,4\}}$，进行第 27 轮解密，共需要 $2^{n-41}\times 2^8\times 2^{32}=2^{n-1}$ 次一轮运算。

对密文对进行过滤，剩余 $2^{n-41}\times 2^{-8}=2^{n-49}$ 对。

步骤 4　猜测第 1 轮的 8 比特子密钥 $K_{0,\{2,4\}}$，进行第 1 轮解密，共需要 $2^{n-49}\times 2^8\times 2^{40}=2^{n-1}$ 次一轮运算。

对密文对进行过滤，剩余 $2^{n-49}\times 2^{-8}=2^{n-57}$ 对。

2.3　复杂度分析

在密钥随机的情况下，明 - 密文经过一轮加密后满足差分特征输入的剩余对数为 2^{n+15-8}，取 $n=54$ 时，通过差分路径的明文对剩余 1 对。根据以上猜测密钥过程，对于正确的密钥，理论上剩余 1 个明 - 密文对；对于错误的密钥，平均剩余 2^{-3} 个明 - 密文对。

复杂度分析：步骤 1 需要加密 $2^{54}\times 2^8\times 2=2^{63}$ 个明文。步骤 2 需要进行 2^{54+21} 次一轮运算，相当于 $2^{54+21}/28\approx 2^{70.2}$ 次 28 轮加密。步骤 3 和步骤 4 的运算量相比步骤 2 可以忽略不计。

接下来采用提前抛弃技术优化步骤 2 的计算复杂度。

步骤 2.1　猜测第 28 轮的低 16 比特子密钥 $K_{27,\{0,1,2,3\}}$，进行一轮解密运算，共需要 $2^{n-9}\times 2^{16}=2^{n+7}$ 次。对密文对进行过滤，剩余 $2^{n-9}\times 2^{-16}=2^{n-25}$ 对。

步骤 2.2　猜测第 28 轮的高 16 比特子密钥 $K_{27,\{4,5,6,7\}}$，进行一轮解密运算，共需要 $2^{n-25}\times 2^{16}\times 2^{16}=2^{n+7}$ 次。对密文对进行过滤，剩余 $2^{n-25}\times 2^{-16}=2^{n-41}$ 对。

当 $n=54$ 时，根据以上步骤可以将主要的计算复杂度降至 $2^{54+7}\times 2/28\approx 2^{57.2}$ 次 28 轮加密。

综上所述，28 轮 PFP 算法的密钥恢复分析需要数据复杂度为 2^{63} 个明文，时间复杂度为 $2^{57.2}$ 次 28 轮加密，共恢复了 48 比特密钥信息。

3　结论

本文针对 PFP 分组密码算法开展了系统的密钥恢复研究。基于前人提出的 25 轮 PFP 差分区分器，本研究通过构造明文结构并使用提前抛弃技术，将密钥恢复攻击扩展至 28 轮，取得了显著进展。

在技术实现层面，本研究主要包含三个创新点：首先，在明文构造方面采用了特殊的结构设计，通过精心选择明文对的形式，在仅需 2^{63} 个明文的条件下实现了高密度的有效明文对组合，显著提升了数据利用效率。其次，在密钥猜测阶段引入了“提前抛弃”技术，通过建立多级筛选机制，在部分密钥猜测不满足条件时立即终止后续计算，将时间复杂度成功降低至 $2^{57.2}$ 次 28 轮加密。最终成功恢复了 48 比特的关键密钥信息，验证了 PFP 算法在 28 轮条件下的安全性缺陷。

从攻击效果评估来看，本次 28 轮密钥恢复攻击的数据复杂度 2^{63} 已接近理论上的穷举攻击边界 2^{64}，这表明现有攻击方法在数据效率方面仍有较大提升空间。针对未来研究，本文提出三个关键改进方向：（1）开发更高效的差分或线性区分器，可能考虑结合混合区分器策略；（2）优化密钥猜测技术，引入机器学习辅助的密钥筛选机制；（3）探索部分密钥恢复与中间相遇攻击的结合方

案。这些研究不仅对评估 PFP 算法的实际安全性具有重要意义，也将为同类 Feistel 结构密码的分析提供新的参考。

参考文献

[1] 梁广俊，辛建芳，王群，等. 物联网取证综述 [J]. 计算机工程与应用，2022，58（8）：12－32.

[2] DOBRAUNIG C，EICHLSEDER M，MENDEL F，et al. Ascon v1.2：lightweight authenticated encryption and hashing [J]. Journal of Cryptology，2021，34：1－42.

[3] BOGDANOV A，KNUDSEN L R，LEANDER G，et al. PRESENT：an ultra-lightweight block cipher [C]//9th International Workshop on Cryptographic Hardware and Embedded Systems，2007：450－466.

[4] WU W，ZHANG L. LBlock：a lightweight block cipher [C]//9th International Conference on Applied Cryptography and Network Security，2011：327－344.

[5] GERARD B，GROSSO V，NAYA-PLASENCIA M，et al. Block ciphers that are easier to mask：how far can we go? [C]//15th International Workshop on Cryptographic Hardware and Embedded Systems，2013：383－399.

[6] 黄玉划，代学俊，时阳阳，等. 基于 Feistel 结构的超轻量级分组密码算法（PFP） [J]. 计算机科学，2017，44（3）：163－167.

[7] BEYNE T，RIJMEN V. Differential cryptanalysis in the fixed-key model [C]//42nd Annual International Cryptology Conference，2022：687－716.

[8] 王建华，怀进鹏. 多重线性密码分析中线性逼近方程的构造 [J]. 计算机工程与应用，2007，43（8）：118－120.

[9] 沈璇，王欣玫，何俊，等. PFP 算法改进的不可能差分分析 [J]. 计算机科学，2020，47（7）：263－267.

[10] BIHAM E，SHAMIR A. Differential cryptanalysis of DES like cryptosystems [J]. Journal of Cryptology，1991，4（1）：3－72.

[11] 李艳俊，李寅霜，杨明华，等. 轻量级 PFP 算法的差分攻击 [J]. 计算机工程与应用，2023，59（21）：296－302.

[12] 陆金玉，刘国强，熊黎依，等. 轻量级分组密码算法 PFP 的差分分析 [J]. 密码学报，2024，11（6）：1293－1307.

[13] 刘道瞳，袁征，魏锦鹏，等. 轻量级分组密码算法 PFP 和 SLIM 的积分分析 [J]. 密码学报，2023，10（3）：1－13.

[14] 严智广，韦永壮，叶涛. 全轮超轻量级分组密码 PFP 的相关密钥差分分析 [J]. 电子与信息学报，2025，47（3）：729－738.

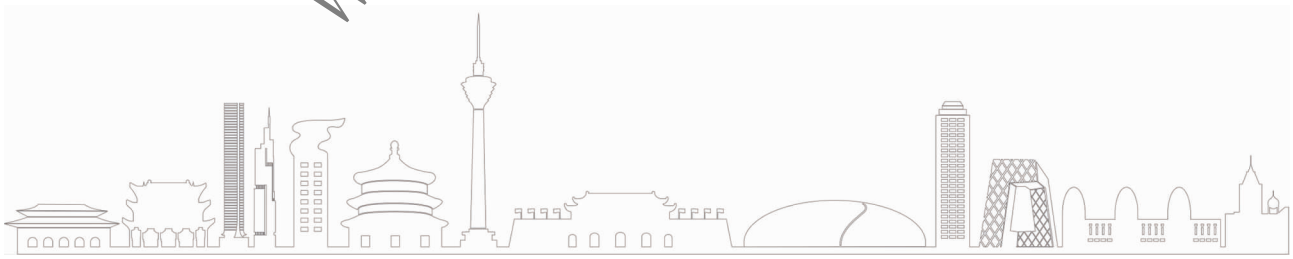
（收稿日期：2025－06－23）

作者简介：

刘健（1983－），男，博士研究生，正高级工程师，主要研究方向：信息系统与信息安全、密码算法与协议安全。

张岩（1987－），女，本科，工程师，主要研究方向：网络信息安全。

黄丁韫（2001－），通信作者，男，硕士研究生，主要研究方向：密码算法设计与分析。E-mail：1295465454@qq.com。



版权声明

凡《网络安全与数据治理》录用的文章，如作者没有关于汇编权、翻译权、印刷权及电子版的复制权、信息网络传播权与发行权等版权的特殊声明，即视作该文章署名作者同意将该文章的汇编权、翻译权、印刷权及电子版的复制权、信息网络传播权与发行权授予本刊，本刊有权授权本刊合作数据库、合作媒体等合作伙伴使用。同时，本刊支付的稿酬已包含上述使用的费用，特此声明。

《网络安全与数据治理》编辑部