基于 NTRU 的加密及签名算法研究

贺婧楠[1,2], 张振飞[3]

[1] 中国科学院信息工程研究所 信息安全国家重点实验室, 北京 中国, 100093
[2] 密码科学技术国家重点实验室, 北京 5159 信箱, 北京 中国, 100878
[3]Algorand, Boston, MA, 02119, US

摘要 NTRU 密码系统作为格密码重要分支, 由于其具有结构简洁、计算速度较快、尺寸较小等优点, 在后量子密码算法研究中受到广泛关注。美国国家标准与技术研究院(NIST)于 2017 年 11 月开始征集后量子密码算法, 三个 NTRU 加密算法(NTRUEncrypt, NTRU Prime, NTRU HRSS)和两个 NTRU 签名算法(pqNTRUSign, Falcon)进入了第一轮评估。这五个算法在基于 NTRU 的加密及签名算法中具有代表性, 因此本文将从设计思路、参数选择、性能对比、安全性评估方面对其进行介绍。

关键词 NTRU; 公钥加密; 数字签名
中图法分类号 TP309.7 DOI 号 10.19363/J.cnki.cn10-1380/tn.2019.03.04

Encryption and Signature Algorithms from NTRU

HE Jingnan[1,2], ZHANG Zhenfei[3]

[1] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
[2] State Key Laboratory of Cryptology, Mail Box 5159, Beijing 100878, China
[3] Algorand, Boston, MA, 02119, US

Abstract In lattice-based cryptography, the cryptosystem based on NTRU is an important research field which owning benefits such as compact structure and better performance in computation and space. In November, 2017, National Institute of Standards and Technology (NIST) called for post-quantum secure algorithms. There are three NTRU encryption algorithms (NTRUEncrypt, NTRU Prime, NTRU HRSS) and two NTRU signature algorithms (pqNTRUSign, Falcon) in the first round submission. Those five NTRU algorithms are representative algorithms of the NTRU cryptosystem. Consequently, in this paper, we will focus on the design rationale, parameter selection, performance and security analysis of those five algorithms.

Key words NTRU; public key encryption; digital signature

1 NTRU 研究背景

近年来, 随着量子计算技术的研究深入, 对实用化可替代现有密码系统的抗量子密码算法的需求日渐迫切。2017 年 11 月美国国家标准与技术研究院(NIST)征集后量子密码算法, 更加推动了可实用化的抗量子密码系统的研究。在五类抗量子攻击的密码体系中, 格密码由于其具有最坏情况安全性归约保证、并行处理以及方案构造全面的优点, 受到大量关注与研究。NTRU 属于格密码领域的典型分支, 具有结构简洁、尺寸较小、计算速度快的优势。

1996 年, Hoffstein, Silverman 和 Pipher[1]提出了 NTRU 公钥加密(Public Key Encryption, PKE)方案, NTRU 假设自此第一次被提出。2003 年, Hoffstein 等人[2]基于 NTRU 提出了签名方案, 是 GGH 签名方案[3]的相对高效的实例化。但是在 2006 年, Nguyen 和 Regev[4]指出 GGH 签名方案及 NTRU 签名方案不安全, 这类签名方案的私钥是较短的格基, 当签名次数过多时, 签名会暴露格基形成的基本区域形状, 从而泄露私钥。此后主要有两个方式构造基于格的签名方案, 从而避免泄露私钥形状信息的问题。一个方法是 Gentry 等人[5]在 2008 年提出的 GPV 签名, 常被称为 hash and sign 设计思路, 该方案需要借助陷门函数进行原像采样。另一个方法是 Lyubashevsky[6,7]

通讯作者: 贺婧楠, 博士, 助理研究员, Email: hejingnan@iie.ac.cn。
本课题得到密码科学技术国家重点实验室开放课题(No. MMKFKT201810), “十三五”国家密码发展基金(No. MMJJ20170123)资助。
收稿日期: 2018-11-28; 修改日期: 2019-03-01; 定稿日期: 2019-03-04

在 2009 年提出的基于 Fiat-Shamir 变换的方法, 该方法不使用陷门函数, 利用拒绝采样的技术避免签名泄露格基形状。Hoffstein 等人[8]在 2014 年基于 NTRU 提出了 hash and sign 类型的签名方案。

1996 年, Hoffstein, Silverman, Pipher 以及 Lieman 成立了 NTRU Cryptosystems 公司, 之后于 2009 年被 Security Innovation 公司收购。NTRU 团队一直致力于商业化和标准化工作, 1997 年, NTRU 加密系统获得了专利, 随后, NTRU 签名等相关技术也获得了专利。2017 年 NTRU 相关核心专利已到期。

1.1 NTRU 格

NTRU 格是多项式环 $\mathcal{R}=\mathbb{Z}[x]/\phi(x)$上的 q 元格。令 $\mathbf{f},\mathbf{g}$ 是多项式环上的短向量, $\mathbf{h}=\mathbf{g}\mathbf{f}^{-1} \bmod q$ (或 $\mathbf{h}=\mathbf{g}(p\mathbf{f})^{-1} \bmod q$), 则 NTRU 的 q 元垂直格为

$$\Lambda^{\perp}\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)=\left\{\boldsymbol{x}\in\mathcal{R}^2\colon \boldsymbol{x}\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}=\mathbf{0} \bmod q\right\}.$$

这和$\mathbb{Z}$上定义的q元垂直格是类似的, 同样地, NTRU q 元垂直格的陪集为

$$\Lambda_{\mathbf{u}}^{\perp}\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)=\{\mathbf{z}\in\mathcal{R}^2:\mathbf{z}\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}=\mathbf{u} \bmod q\}.$$

NTRU 的另一个 q 元格为

$$\Lambda\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)=\{\mathbf{z}\in\mathcal{R}^2:\exists\mathbf{s}\in\mathcal{R}_q,\ s.t.\ \begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\mathbf{s}=\mathbf{z} \bmod q\}.$$

与$\mathbb{Z}$上定义的 q 元格类似, $\Lambda^{\perp}\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)$和$\Lambda\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)$存在 q 倍对偶关系。$\begin{bmatrix}q & 0\\ -\mathbf{h} & 1\end{bmatrix}$为$\Lambda^{\perp}\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)$的 Hermite 形式格基, 即为一组不好的格基, 作为公钥公开。由于$\mathbf{f}$模 q 可逆, 则 $\Lambda^{\perp}\left(\begin{bmatrix}\mathbf{f}\\ \mathbf{g}\end{bmatrix}\right)=\Lambda^{\perp}\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)$, 并且 $\begin{bmatrix}\mathbf{g} & -\mathbf{f}\\ \bar{\mathbf{g}} & -\bar{\mathbf{f}}\end{bmatrix}$是 $\Lambda^{\perp}\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)$的一组短格基, 其中$\mathbf{f}\bar{\mathbf{g}}-\mathbf{g}\bar{\mathbf{f}}=q,\ \bar{\mathbf{g}},\bar{\mathbf{f}}\in\mathcal{R}$。因此将$\mathbf{f},\mathbf{g}$作为私钥。

基于以上信息, NTRU 的加密方案使用 q 元格 $\Lambda\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)$构造, 解密相当于求解格 $\Lambda\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)$上的最近向量问题(closest vector problem, CVP)。与带错误的学习问题 (learning with errors, LWE) 求逆相似。NTRU 的签名方案使用 q 元垂直格的陪集$\Lambda_{\mathbf{u}}^{\perp}\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)$构造, 签名与非齐次短整数解问题 (inhomogeneous short integer solution, ISIS) 求逆相似, 即求垂直格$\Lambda^{\perp}\left(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}\right)$的短向量。

NTRU 假设是指已知$\mathbf{h}$, 求$\mathbf{f},\mathbf{g}$是困难的。短向量$\mathbf{f},\mathbf{g}$目前有两种抽样的方法, 一种是文献[1]及后续相关工作采用的分布, 每个系数从$\{0,1\}$或$\{-1,0,1\}$中选取, 有固定个数的 1 及–1。采用这类分布时, 目前 NTRU 假设还未能归约到格上的困难问题, 但从 1996 年提出至今, 也仍未发现可以攻击该假设的有效算法。另一种分布是系数取自高斯分布, Stehlé和 Steinfeld[9]将采用高斯分布情况下的 NTRU 假设归约到了理想格上最坏情况下的困难问题, 在一定程度上得到了理论归约的保证, 但此时参数要求过大, 不适用于实用的方案。

基于 NTRU 格实现的方案具有结构简洁、密钥密文尺寸小的优点, 是利于实用化的方案。虽然目前在理论归约方面还未能获得保证, 但提出二十多年来还没有有效的攻击算法, 从攻击角度在方案安全性方面得到了保证。

1.2 基于 NTRU 的 NIST 候选方案

NIST 于 2017 年 11 月征集后量子密码算法后, 共有 69 个算法进入了第一轮评估, 其中格密码算法有 30 个。在格密码算法中, 基于 NTRU 的算法共有 5 个。3 个密钥封装(Key Encapsulation Mechanisms, KEM) 算法: NTRUEncrypt、NTRU Prime 以及 NTRU HRSS, 这三个算法都是基于 PKE 封装实现, 因此本文在第 3 节将以公钥加密方式进行讨论。2 个签名算法: pqNTRUSign 和 Falcon, 本文将在第 4 节进行讨论。

1.3 章节安排

本文第 2 部分对基本符号进行说明; 第 3 部分介绍 NTRU 公钥加密方案算法、参数选择及性能对比; 第 4 部分介绍 NTRU 签名算法、参数选择及性能对比; 第 5 部分介绍目前针对 NTRU 密码系统的主要攻击方法及攻击复杂度对比。

2 符号说明

2.1 符号

令加粗小写字母表示向量, 如$\mathbf{a}$, 加粗大写字母表示矩阵, 如$\mathbf{A}$。使用向量$\mathbf{f}=(f_0,f_1,\cdots,f_{n-1})$表示多项式 $f(x)=f_0+f_1x+\cdots+f_{n-1}x^{n-1}$。

$\mathcal{B}_{\mathrm{N}}$ 表示二元多项式集合, 即多项式系数从{0,1}中选取, $\mathcal{T}_{\mathrm{N}}$ 表示三元多项式集合, 即多项式系数从{–1,0,1}中选取, $\mathcal{T}(d,e)$ 表示有 d 个 1 和 e 个–1 的三元多项式集合。当 N 确定时使用$\mathcal{B},\mathcal{T},\mathcal{T}(d,e)$标记。

2.2 NIST 安全级别分类

NIST 征集后量子密码算法对安全级别分类[10]如下:

(1) 级别 1: 与 128 位分组密码安全性相同;
(2) 级别 2: 与 256 位哈希函数安全性相同;
(3) 级别 3: 与 192 位分组密码安全性相同;
(4) 级别 4: 与 384 位哈希函数安全性相同;
(5) 级别 5: 与 256 位分组密码安全性相同。

3 NTRU 公钥加密方案

在提交 NIST 后量子候选算法中, 基于 NTRU 的加密方案共有三个, 分别是 Zhang 等人[11]提出的 NTRUEncrypt 算法, Bernstein 等人[12]提出的 NTRU Prime 算法, 以及 Schanck 等人[13]提出的 NTRU HRSS 算法。其中, NTRUEncrypt 算法设计思路与 NTRU 最初方案类似, 其他方案设计思路在结构、参数方面略有不同, 因此, 在 3.1 节详细说明 NTRUEncrypt 算法, 3.2 及 3.3 节说明其余算法不同之处, 不再列明算法步骤。

3.1 NTRUEncrypt 方案构造

NTRUEncrypt 方案构造方式如表 1-3, 其中多项式生成算法 Sampler, 在集合 $\mathcal{B},\mathcal{T},\mathcal{T}(d,e)$ 中均匀抽取多项式, 当输入种子 *seed* 时该生成算法是确定性算法。原提交算法[11]中 NTRUEncrypt-443/743 采用早期 NTRU 加密类似方法, 参数及实现效率均有利于实用化。而 NTRUEncrypt-1024 采用文献[9]的设计思路, 由于参数过大不利于实用化, 但又未达到文献[9]的理论归约参数要求, 因此本文不讨论 NTRUEncrypt-1024 算法。

表 1 NTRUEncrypt 密钥生成算法

Table 1 Key generation algorithm of NTRUEncrypt

算法 1 NTRUEncrypt.KEYGEN
输入: 参数集合 PARAM = $\{N, p, q, d\}$ 及随机数种子 seed
1: 使用 $\mathcal{T}(d,d)$, seed 实例化 Sampler 算法
2: $\mathbf{f} \leftarrow$ Sampler
3: 如果 $(p\mathbf{f}+1) \bmod q$ 不可逆, 则跳转到第 2 步
4: $\mathbf{g} \leftarrow$ Sampler
5: $\mathbf{h} = \mathbf{g}/(p\mathbf{f}+1) \mod q$
输出: 公钥 $\mathbf{h}$ 和私钥 $(p\mathbf{f}, \mathbf{g})$

表 2 NTRUEncrypt 加密算法

Table 2 Encryption algorithm of NTRUEncrypt

算法 1 NTRUEncrypt.ENCRYPT
输入: 公钥 $\mathbf{h}$, 长度为 *mlen* 的消息 *msg*, 参数集 PARAM 及随机数种子 seed
1: $\mathbf{m} = \text{Pad}(msg, seed)$
2: $rseed = \text{HASH}(\mathbf{m}\|\mathbf{h})$
3: 使用 $\mathcal{T}$ 和 *rseed* 实例化 Sampler 算法
4: $\mathbf{r} \leftarrow$ Sampler
5: $\mathbf{t} = \mathbf{r} * \mathbf{h}$
6: $tseed = \text{HASH}(\mathbf{t})$
7: 使用 $\mathcal{T}$ 和 *tseed* 实例化 Sampler 算法
8: $\mathbf{m}_{mask} \leftarrow$ Sampler
9: $\mathbf{m}' = \mathbf{m} - \mathbf{m}_{mask} \pmod p$
10: $\mathbf{c} = \mathbf{t} + \mathbf{m}'$
输出: 密文 $\mathbf{c}$

表 3 NTRUEncrypt 解密算法

Table 3 Decryption algorithm of NTRUEncrypt

算法 1 NTRUEncrypt.DECRYPT
输入: 私钥 $\mathbf{f}$, 公钥 $\mathbf{h}$, 密文 $\mathbf{c}$, 参数集 PARAM
1: $\mathbf{m}' = \mathbf{f} * \mathbf{c} \pmod p$
2: $\mathbf{t} = \mathbf{c} - \mathbf{m}'$
3: $tseed = \text{HASH}(\mathbf{t})$
4: 使用 $\mathcal{T}$ 和 *tseed* 实例化 Sampler 算法
5: $\mathbf{m}_{mask} \leftarrow$ Sampler
6: $\mathbf{m} = \mathbf{m}' + \mathbf{m}_{mask} \pmod p$
7: $rseed = \text{HASH}(\mathbf{m}\|\mathbf{h})$
8: 使用 $\mathcal{T}$ 和 *rseed* 实例化 Sampler 算法
9: $\mathbf{r} \leftarrow$ Sampler
10: $msg, mlen = \text{Extract}(\mathbf{m})$
11: 如果 $p \cdot \mathbf{r} * \mathbf{h} = \mathbf{t}$ 则 $result = msg, mlen$
12: 否则 $result = \perp$
输出: *result*

3.2 NTRU Prime 方案

NTRU Prime 是 KEM 方案, 但仍是从 PKE 搭建 KEM 的设计思路, 可以将其看作 PKE 方案。包含两种方案, 分别称为 Streamlined NTRU Prime 和 NTRU LPRime。

(1) Streamlined NTRU Prime 方案结构和 NTRUEncrypt 类似, 公钥采用商的形式, 密文只有一项。具体如下, 密钥生成算法中, 公钥为 $\mathbf{h} = \mathbf{g}/(p\mathbf{f})$, 私钥仍是 $(\mathbf{g}, \mathbf{f})$。加密算法中, 由于实际是 KEM 算法, 在具体算法中, 密文是对 $\mathbf{hr}$ 在 $[-\frac{q-1}{2}, \frac{q-1}{2}]$ 上取整, 相当于 NTRUEncrypt 中的密文 $\mathbf{hr}+\mathbf{m}$, $\mathbf{m}$ 的选取使得 $\mathbf{hr}+\mathbf{m}$ 的系数能够落在 $\mathbb{Z}_q$ 内。解密算法类似 NTRUEncrypt, 使用私钥恢复 $\mathbf{r}$。

(2) NTRU LPRime 采用环 LWE[14]方案结构, 公钥是类似环 LWE 实例, 密文由两项组成。具体如下, 密钥生成算法中, 公钥 $\mathbf{h}$ 为 $\mathbf{as}$ 在 $[-\frac{q-1}{2}, \frac{q-1}{2}]$ 上取整, 相当于环 LWE 实例 $\mathbf{as}+\mathbf{e}$, 其中 $\mathbf{a}$ 是公开生成的随机多项式, 私钥为 $\mathbf{s}$。加密算法中, 密文第一项是 $\mathbf{ar}$ 在 $[-\frac{q-1}{2}, \frac{q-1}{2}]$ 上取整, 相当于环 LWE 实例 $\mathbf{ar}+\mathbf{e}'$; 密文第二项是 $\mathbf{hr}+\mathbf{m}\lfloor\frac{q}{2}\rfloor$。

3.3 NTRU HRSS 方案

NTRU HRSS 是 KEM 方案, 也是从 PKE 搭建 KEM 的设计思路, 仍将其看作基础 OW-CPA 安全的 PKE 方案。方案结构和 NTRUEncrypt 相同, 多项式环、模数、私钥及消息采样空间等参数选择不同, 具体见 3.4 节。

3.4 参数选择及性能对比

上述三个加密方案关于维度 N, 模数 q,p, 私钥

$\mathbf{f},\mathbf{g}$系数分布参数 d, 以及多项式环 $\mathcal{R}$ 的参数选择见表 4, 其中 NIST 栏表示方案对应 NIST 要求的安全级别。

表 4　ntru-pke 参数选择

Table 4　Parameters of NTRUEncrypt

方案名称	N,q,p,d	$\mathcal{R}$	NIST
NTRUEncrypt –443	443,2048,3,143	$\frac{Z_q[x]}{x^N-1}$	1
NTRUEncrypt –743	743,2048,3,247	$\frac{Z_q[x]}{x^N-1}$	1,2, 3,4,5
Streamlined NTRU Prime	1024,4591,3,286	$\frac{Z_q[x]}{x^N-x-1}$	5
NTRU LPRime	1024,4591,3,250	$\frac{Z_q[x]}{x^N-x-1}$	5
NTRU HRSS	701,8192,3,N/A	$\frac{Z_q[x]}{\sum_{i=0}^{N-1} xi}$	1

从公钥、私钥、密文长度、消息长度以及密钥生成、加密、解密算法使用 CPU 轮数对上述加密方案进行对比, 具体见表 5。其中公钥、私钥、密文、消息长度均为字节, 来源为 SAFE Crypto 测试数据[15]。对应 KEM 方案, 公钥、密文长度代表通信量, 消息长度代表协商密钥的长度。

表 5　ntru-pke 性能对比[15]

Table 5　Performance comparison of ntru-based PKE schemes[15]

方案名称	公钥	私钥	密文	消息	生成密钥 CPU 轮数	加密 CPU 轮数	解密 CPU 轮数
NTRUEncrypt -443	611	701	611	32	1257307	394406	363281
NTRUEncrypt -743	1023	1173	1023	48	3031086	579527	767267
Streamlined NTRU Prime	1218	1600	1047	32	43865807	27101314	62508579
NTRU LPRime	1047	1238	1175	32	14060919	44116905	71245370
NTRU HRSS	1138	1418	1278	32	191376309	3965430	11383908

4　NTRU 签名方案

NIST 候选方案中有两个基于 NTRU 的签名方案, 均是基于 Hash and Sign 的设计思路, 分别是 Zhang 等人[16]提出的 pqNTRUSign 方案, 以及 Prest 等人[17]提出的 Falcon 方案。本文将在 4.1 节与 4.2 节分别讨论这两个方案。

4.1　pqNTRUSign 方案

方案设计思路: 将待签名的消息 μ 通过 $hash(\mathbf{h}|\mu)$ 映射到 $\mathbb{Z}_p^{2N}$ 上均匀分布的元素 $(\mathbf{u}_p,\mathbf{v}_p)$, 借助短向量 $\mathbf{f},\mathbf{g}$, 在集合 $\mathcal{L}_h\cap\{p\mathbb{Z}^{2N}+(\mathbf{u}_p,\mathbf{v}_p)\}$ 中找到较短的向量 $(\mathbf{u},\mathbf{v})$ 作为签名, 其中 $\mathcal{L}_h=\{(\mathbf{u},\mathbf{v})\in\mathcal{R}^2:\mathbf{uh}=\mathbf{v}\}$。即 $(\mathbf{u},\mathbf{v})$ 是格 $\mathcal{L}_h$ 中满足 $(\mathbf{u},\mathbf{v})\equiv(\mathbf{u}_p,\mathbf{v}_p)\bmod p$ 的较短的格点。验证签名时, 判断签名 $(\mathbf{u},\mathbf{v})$ 是否是格点, 且长度是否符合要求。各算法具体步骤见表 6-10, 其中, 签名算法分别采用高斯拒绝采样和均匀拒绝采样避免过多的签名泄露格基形状。

表 6　pqNTRUSign 密钥生成算法

Table 6　Key generation algorithm of pqNTRUSign

算法 1 pqNTRUSign.KEYGEN

输入: 参数集合 PARAM = $\{N,p,q,d,B_k\}$

1: 使用 $\mathcal{T}(d+1,d)$ 实例化 Sampler 算法

2: $\mathbf{f}\leftarrow$ Sampler

3: 如果 $\mathbf{f}\bmod q$ 不可逆, 则跳转到第 2 步

4: 如果 $\|\mathbf{f}\|\geqslant B_k$, 则跳转到第 2 步

5: $\mathbf{g}\leftarrow$ Sampler

6: 如果 $\mathbf{g}\bmod p$ 不可逆, 则跳转到第 5 步

7: 如果 $\|\mathbf{g}\|\geqslant B_k$, 则跳转到第 5 步

8: $\mathbf{h}=\mathbf{g}/(p\mathbf{f})\bmod q$

输出: 公钥 $\mathbf{h}$ 和私钥 $(p\mathbf{f},\mathbf{g})$

表 7　pqNTRUSign 针对高斯分布参数的签名算法

Table 7　Signing algorithm of pqNTRUSign for Gaussian parameters

算法 1 pqNTRUSign.SIGN

输入: 参数集合 PARAM = $\{N,p,q,M_s,B_s,B_t\}$ 及公钥 $\mathbf{h}$, 私钥 $\mathbf{f},\mathbf{g}$, 消息 μ, 分布 χ_σ。

1: $(\mathbf{u}_p,\mathbf{v}_p)=\mathrm{HASH}(\mu|\mathbf{h})$

2: $\mathbf{r}\leftarrow\chi_\sigma^N$, $b\leftarrow\{0,1\}$

3: $\mathbf{u}_1=p\mathbf{r}+\mathbf{u}_p$; $\mathbf{v}_1=\mathbf{u}_1\mathbf{h}\bmod q$

4: $\mathbf{a}=(\mathbf{v}_p-\mathbf{v}_1)/\mathbf{g}\bmod p$

5: 如果 $\|\mathbf{af}\|_2>B_s$ 或 $\|\mathbf{ag}\|_\infty>B_t$, 则跳转到第 2 步

6: $\mathbf{v}=\mathbf{v}_1+(-1)^b\mathbf{ag}$

7: 如果 $\|\mathbf{v}\|_\infty>q/2-B_t$, 则跳转到第 2 步

8: 以概率 $\frac{1}{M_s exp(-\frac{\|\mathbf{af}\|}{2\sigma^2})cosh(\frac{<\mathbf{b},\mathbf{af}>}{\sigma^2})}$ 输出 $\mathbf{b}=(\mathbf{r}+(-1)^b\mathbf{af})$

输出: 签名 $\mathbf{b}$

表 8　pqNTRUSign 针对高斯分布参数的验证算法

Table 8　Verification algorithm of pqNTRUSign for Gaussian parameters

算法 1 pqNTRUSign .Verification

输入: 参数集合 PARAM = $\{N,p,q,B_t,\sigma\}$ 及公钥 $\mathbf{h}$, 签名 $\mathbf{b}$, 消息 μ。

1: $(\mathbf{u}_p,\mathbf{v}_p)=\mathrm{HASH}(\mu|\mathbf{h})$

2: $\mathbf{u}=p\mathbf{b}+\mathbf{u}_p$

3: 如果 $\|\mathbf{u}\|^2>p^2\sigma^2N$, 则输出拒绝

4: $\mathbf{v}=\mathbf{uh}\bmod q$

5: 如果 $\mathbf{v}\neq\mathbf{v}_p\bmod p$ 或 $\|\mathbf{v}\|_\infty>q/2-B_t$, 则输出拒绝

6: 输出接受

输出: 接受或拒绝

表 9 pqNTRUSign 针对均匀分布参数的签名算法
Table 9 Signing algorithm of pqNTRUSign for uniform parameters

算法 1 pqNTRUSign.SIGN for uniform distributions
输入: 参数集合 PARAM = $\{N, p, q, M_s, B_s, B_t\}$ 及公钥 $\mathbf{h}$, 私钥 $\mathbf{f}, \mathbf{g}$, 消息 $\mu \in \{0,1\}^*$。
1: $(\mathbf{u}_p, \mathbf{v}_p) = \text{HASH}(\mathbf{h}, \mu)$
2: $\mathbf{r} \xleftarrow{\$} \mathcal{U}^N_{\lfloor q/(2p)+0.5 \rfloor}$
3: $\mathbf{u}_0 = p\mathbf{r} + \mathbf{u}_p$; $\mathbf{v}_0 = \mathbf{u}_0\mathbf{h} \bmod q$
4: $\mathbf{a} = (\mathbf{v}_p - \mathbf{v}_0)/\mathbf{g} \bmod p$
5: $(\mathbf{u}, \mathbf{v}) = (\mathbf{u}_0, \mathbf{v}_0) + (\mathbf{af}, \mathbf{ag})$
6: 如果 $\|\mathbf{af}\| > B_s$ 或 $\|\mathbf{ag}\| > B_t$ 或 $\|\mathbf{u}\| > \frac{q}{2} - B_s$ 或 $\|\mathbf{v}\| > \frac{q}{2} - B_t$, 则跳转到第 2 步
输出: 签名 $(\mathbf{u}, \mathbf{v}, \mu)$

表 10 pqNTRUSign 针对均匀分布参数的验证算法
Table 10 Verification algorithm of pqNTRUSign for uniform parameters

算法 1 pqNTRUSign.Verification
输入: 参数集合 PARAM = $\{N, p, q, B_t, B_s\}$ 及公钥 $\mathbf{h}$, 签名 $\mathbf{u}, \mathbf{v}$, 消息 μ。
1: $(\mathbf{u}_p, \mathbf{v}_p) = \text{HASH}(\mathbf{h}, \mu)$
2: 如果 $\mathbf{v} \neq \mathbf{hu} \bmod q$ 则输出拒绝
3: 如果 $\|\mathbf{u}\| > \frac{q}{2} - B_s$ 或 $\|\mathbf{v}\| > \frac{q}{2} - B_t$, 则输出拒绝
4: $\mathbf{v} = \mathbf{uh} \bmod q$
5: 如果 $(\mathbf{u}, \mathbf{v}) \neq (\mathbf{u}_p, \mathbf{v}_p) \bmod p$, 则输出拒绝
6: 输出接受
输出: 接受或拒绝

表 11 pqNTRUSign 参数选择
Table 11 Parameters of pqNTRUSign

方案名称	N, q, p	$\mathcal{R}, d, \sigma$	B_k, B_s, B_t	NIST
Gaussian –1024	1024,65537,2	$\frac{q[x]}{x^N+1}$,205,250	40,500,49	1,2, 3,4,5
Uniform –1024	1024,65537,2	$\frac{q[x]}{x^N+1}$,205,N/A	40,98,49	1,2, 3,4,5

4.2 Falcon 方案

该方案是基于 NTRU 对 Gentry 等人[5]提出的签名算法的实现。设计思路: 将待签名的消息 μ 通过 hash 映射到 $\mathcal{R}_q$ 上均匀分布的元素 $\mathbf{u}$, 通过原像采样找到服从高斯分布且满足 $\mathbf{s}\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix} = \mathbf{u} \bmod q$ 的较短 $\mathbf{s}$, 即为签名。

NTRU q 元垂直格为

$$\mathcal{L}_h = \Lambda^{\perp}(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}) = \{\mathbf{x} \in \mathcal{R}^2 : \mathbf{x}\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix} = \mathbf{0} \bmod q\},$$

其中, $\mathbf{h} = \mathbf{gf}^{-1} \bmod q$, $\mathbf{g}, \mathbf{f} \in \mathbb{Z}(x)/\phi$ 为小系数的多项式, ϕ 是 x 的某个多项式。令 $\bar{\mathbf{g}}, \bar{\mathbf{f}} \in \mathbb{Z}(x)/\phi$ 为小系数的多项式并且满足 $\mathbf{f}\bar{\mathbf{g}} - \mathbf{g}\bar{\mathbf{f}} = q \bmod \phi$, 由于

$$\mathbf{B} = \begin{bmatrix}\mathbf{g} & -\mathbf{f}\\ \bar{\mathbf{g}} & -\bar{\mathbf{f}}\end{bmatrix}$$

的行向量为 $\mathcal{L}_h$ 的格点, 且 $\det(\mathbf{B}) = q = \det(\mathcal{L}_h)$ 表明 $\mathbf{B}$ 的行向量组成的基本区域内没有其他格点, 所以 $\mathbf{B}$ 是 $\mathcal{L}_h$ 的一组短格基。将 $\mathbf{B}$ 作为私钥, $\mathbf{h}$ 作为公钥。选择合适的多项式 ϕ 将有利于计算符合要求的 $\mathbf{g}, \mathbf{f}, \bar{\mathbf{g}}, \bar{\mathbf{f}}$。

签名 $\mathbf{s}$ 是集合

$$\Lambda^{\perp}_{\mathbf{u}}(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix}) = \{\mathbf{s} \in \mathcal{R}^2_q : \mathbf{s}\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix} = \mathbf{u} \bmod q\}$$

中较短的向量。$\Lambda^{\perp}_{\mathbf{u}}(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix})$ 是 $\mathcal{L}_h$ 所有格点平移 $(\mathbf{u}, \mathbf{0})$ 后的所有点的集合, 即为 $\mathcal{L}_h$ 的陪集。因此, 在格 $\mathcal{L}_h$ 中使用格基 $\mathbf{B}$ 找到离 $(\mathbf{u}, \mathbf{0})$ 最近的格点 $(\mathbf{s}'_1, \mathbf{s}'_2)$, 则

$$(\mathbf{u}, \mathbf{0}) - (\mathbf{s}'_1, \mathbf{s}'_2)$$

是集合 $\Lambda^{\perp}_{\mathbf{u}}(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix})$ 上的短向量, 即为满足条件的签名。

验证签名时, 检验签名 $\mathbf{s}$ 是否足够短, 并且是否属于集合 $\Lambda^{\perp}_{\mathbf{u}}(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix})$。

Falcon 完整方案中还考虑了有利于提高实现效率的加速方法, 具体算法可见文献[17]。

表 12 Falcon 参数选择
Table 12 Parameters of Falcon

方案名称	N	q	$\mathcal{R}$	σ	NIST
Falcon-512	512	12289	$\frac{q[x]}{x^N+1}$	4.05	1
Falcon-768	768	18433	$\frac{q[x]}{x^N - x^{N/2}+1}$	4.05	2,3
Falcon-1024	1024	12289	$\frac{q[x]}{x^N+1}$	2.87	4,5

4.3 NTRU 签名方案性能对比

本节从私钥、公钥、签名长度以及密钥生成、签名、验证算法使用 CPU 轮数对 pqNTRUSign 和 Falcon 方案进行对比, 具体见表 13, 其中私钥、公钥、签名长度均为字节。来源为 SAFE Crypto 测试数据[15]。

5 安全性分析

基于 NTRU 假设, 上述 ntru-pke 方案是 CPA 安全的。基于 NAEP 转换[18]的性质, NTRUEncrypt 的 KEM 方案是 CCA2 安全的。NTRU Prime 和 NTRU HRSS 通过 KEM+DEM 模式[19]可以获得 CCA2 安全性。

下面从常用攻击方面分析方案安全性。

表 13 NTRU 签名方案性能对比[15]

Table 13 Performance comparison of ntru-based signature schemes

方案名称	私钥	公钥	签名	生成密钥(轮数)	签名(轮数)	验证(轮数)
Falcon-512	4097	897	690	300030872	19884364	666108
Falcon-768	6145	1441	1077	91009209	8359971	1117624
Falcon-1024	8193	1793	1330	157623028	13058641	1384574
Gaussian-1024	2604	2065	2065	259672814	349028118	2955494
Uniform-1024	2604	2065	2065	268329761	202185303	2726230

5.1 uSVP 攻击

上述加密方案及签名方案的公钥所在 NTRU 格基的矩阵形式为

$$\mathbf{B}=\begin{pmatrix} q\mathbf{I}_N & 0 \\ \mathbf{H} & \mathbf{I}_N \end{pmatrix}$$

其中, $\mathbf{I}_N$ 是 N 维单位阵, $\mathbf{H}$ 的行向量是公钥 $\mathbf{h}$ 循环移位形成的(在 NTRU Prime 方案中, $\mathbf{H}$ 的第 i 行向量对应 $x^i \cdot 3\mathbf{h} \bmod x^N - x - 1$)。根据高斯启发式, 私钥 $(\mathbf{f},\mathbf{g})$ 及其循环移位的向量是该 NTRU 格的最短向量。求解该格的唯一最短向量问题 (unique shortest vectors problem, uSVP), 通过找到该格的最短向量, 作为私钥正确解密, 从而进行攻击[20,21]。

求解 uSVP 问题目前主要通过 BKZ 格基约化算法[22,23]得到相对垂直的优质格基, 从而求得短向量。因此, BKZ 算法的复杂度作为该攻击方法复杂度估计。BKZ 算法将格分为多个低维度格, 在低维度格使用求解最短向量问题 (shortest vectors problem, SVP) 算法, 从而获得格的短向量。划分的低维度格越少, 维度越接近原格维度, 则得到的向量越接近最短向量, 但求解短向量时间复杂度也越高。反之, 划分的低维度格越多, 时间复杂度越低, 但得到的向量越长。因此 BKZ 算法需要根据具体求解 SVP 算法的时间复杂度情况合理选择划分方式。目前求解 SVP 主要有两种实现方法: 筛选[24]和枚举方法[25]。

具体攻击复杂度见表 14, 其中"自评估"一栏表示候选算法提交文档中评估的最好的攻击复杂度, 其余经典筛法、经典枚举、量子筛法、量子枚举攻击复杂度采用 Albrecht 等人在文献[26]中提供的评估数据。

5.2 Hybrid 攻击

由于上述多个方案私钥 $(\mathbf{f},\mathbf{g})$ 的系数从{–1,0,1}中选取, 且分布稀疏, 因此对格进行分解处理, 缩小搜索范围, 利用中间相遇攻击[27]进行搜索碰撞将有一定优势。Hybrid 攻击即是结合求最近向量问题(CVP)和中间相遇攻击找到私钥 $(\mathbf{f},\mathbf{g})$ [28]。当目标点和格点的距离小到一定程度后, CVP 问题是容易求解的[29,30]。因此, 将格转化为三个相互垂直的格 L1, L2 以及 L3, 在 L3 的子集中使用中间相遇攻击搜索短向量, 将 L2 格基进行格基约化获得相对较好的格基, 再将 L3 中搜索的短向量对应到 L2 中求解 CVP 问题, 即可求出 L 中的最短向量, 即 NTRU 加密方案中的私钥。中间相遇攻击将短向量所在搜索空间分为更稀疏的两个子空间分别搜索, 使用空间换取时间, 考虑量子攻击的情况下, 对其使用量子 Grover 算法分析复杂度。具体攻击复杂度见表 15, 数据来源自 NIST 算法文档[11,13,16]。由于 Falcon 方案的 $(\mathbf{f},\mathbf{g})$ 不是稀疏分布, 不利于中间相遇攻击, 因此该攻击方法不如 uSVP 攻击有效。

表 14 uSVP 攻击复杂度

Table 14 Costs of uSVP attack

方案名称	自评估	经典筛法	经典枚举	量子筛法	量子枚举	NIST
NTRUEncrypt-443	84	93	186	85	93	1
NTRUEncrypt-743	159	175	441	159	221	1,2,3,4,5
Streamlined NTRU Prime-1024	225	154	356	139	183	5
NTRU LPRime-1024	248	155	370	140	187	5
NTRU HRSS-701	123	136	313	123	157	1
pqNTRUSign-1024	149	168	416	152	208	1,2,3,4,5
Falcon-512	103	141	330	128	165	1
Falcon-768	172	213	571	193	286	2,3
Falcon-1024	230	285	836	259	418	4,5

表 15 Hybrid 攻击复杂度

Table 15 Costs of hybrid attack

方案名称	Hybrid 攻击		
	经典筛法	经典枚举	量子筛法/量子枚举
NTRUEncrypt-443	89	128	84/-
NTRUEncrypt-743	173	267	163/-
NTRU HRSS	82	200	-/162
pqNTRUSign-1024	165	269	154/-

5.3 伪造签名攻击

针对 pqNTRUSign 方案，伪造签名攻击相当于找到垂直格 $\Lambda^{\perp}(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix})$ 与 $p\mathbb{Z}^{2N}$ 交集中的短向量；针对 Falcon 方案，伪造签名攻击相当于在垂直格 $\Lambda^{\perp}(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix})$ 上求解 CVP 问题。按照方案的参数选择，这两个方式均不会比恢复私钥 $\mathbf{f},\mathbf{g}$ 更容易，私钥 $\mathbf{f},\mathbf{g}$ 为垂直格 $\Lambda^{\perp}(\begin{bmatrix}1\\ \mathbf{h}\end{bmatrix})$ 上的最短向量。因此可直接考虑 uSVP 攻击复杂度。

5.4 子域攻击

Albrecht, Cheon 以及 Kiltz 等人[31-33] 提出当 $(\mathbf{f},\mathbf{g})$ 对比模数 q 过小时存在子域攻击，此时 NTRU 格是一个过于拉伸的格，公钥 $\mathbf{h}$ 可以被分解到某个子域上，该子域格上的 SVP 及 CVP 问题将容易解决，从而恢复私钥 $(\mathbf{f},\mathbf{g})$。上述算法选择的参数均不在目前子域攻击范围。

6 展望

基于 NTRU 的密码系统具有尺寸较小、结构简洁、可构造方案全面的优势，有利于设计实用化抗量子密码方案。但同时由于结构、分布特殊，在安全性方面需要进行更有针对性的分析评估，在理论归约方面还有较大的探索空间。在实现效率方面，参数选择结合快速实现采样、多项式乘加运算是值得研究的问题。

参考文献

[1] J. Hoffstein, J. Pipher, and J. H. Silverman, “NTRU: A new high speed public key cryptosystem”. Technical report, presented at the rump session of *Annual International Cryptology Conference (CRYPTO)*, 1996.

[2] J. Hoffstein, N.H. Graham, J. Pipher, J. H. Silverman, and W. Whyte, “NTRUSign: Digital Signatures Using the NTRU Lattice,” in *Cryptographers' Track at the RSA Conference (CT-RSA)*, pp. 122-140, 2003.

[3] O. Goldreich, S. Goldwasser, and S. Halevi, “Public-key cryptosystems from lattice reduction problems,” in *Annual International Cryptology Conference (CRYPTO)*, pp. 112-131, 1997.

[4] P. Nguyen and O. Regev, “Learning a Parallelepiped: Cryptanalysis of GGH and NTRU Signatures,” in *Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pp. 271-288, 2006.

[5] C. Gentry, C. Peikert, and V. Vaikuntanathan, “Trapdoors for hard lattices and new cryptographic constructions,” in *Annual ACM Symposium on Theory of Computing (ACM STOC)*, pp. 197-206, 2008.

[6] V. Lyubashevsky, “Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures,” in *International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT)*, pp. 598-616, 2009.

[7] V. Lyubashevsky, “Lattice signatures without trapdoors,” in *Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pp. 738-755, 2012.

[8] J. Hoffstein, J. Pipher, J.M. Schanck, J.H. Silverman, and W. Whyte, “Transcript Secure Signatures Based on Modular Lattices,” in *International Workshop on Post-Quantum Cryptography*, pp. 142–159, 2014.

[9] D. Stehlé, R. Steinfeld, “Making NTRU as Secure as Worst-Case Problems over Ideal Lattices,” in *Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)* , pp. 27-47, 2011.

[10] NIST Post-Quantum Cryptography Project. Available at https://csrc.nist.gov/Projects/Post-Quantum-Cryptography.

[11] Z. Zhang, C. Chen, J. Hoffstein, and W. Whyte, “NTRUEncrypt”. Technical report, National Institute of Standards and Technology, 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

[12] D.J. Bernstein, C. Chuengsatiansup, T. Lange, and C.V. Vredendaal, “NTRU Prime”. Technical report, National Institute of Standards and Technology, 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

[13] J.M. Schanck, A. Hulsing, J. Rijneveld, and P. Schwabe, “NTRU HRSS KEM”. Technical report, National Institute of Standards and Technology, 2017. Available at https: //csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

[14] V. Lyubashevsky, C. Peikert, and O. Regev, “On ideal lattices and learning with errors over rings,” in *Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pp. 1-23, 2013.

[15] SAFE Crypto via https://www.safecrypto.eu/.

[16] Z. Zhang, C. Chen, J. Hoffstein, and W. Whyte, “pqNTRUSign”. Technical report, National Institute of Standards and Technology, 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

[17] T. Prest, P.A. Fouque, J. Hoffstein, P. Kirchner, V. Lyubashevsky, T. Pornin, T. Ricosset, G. Seiler, W. Whyte, and Z. Zhang, “Falcon”. Technical report, National Institute of Standards and Technology, 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

[18] N. Howgrave-Graham, J.H. Silverman, and W. Whyte, “Choosing Parameter Sets for NTRUEncrypt with NAEP and SVES-3,” in *Cryptographers' Track at the RSA Conference* (*CT-RSA*) , pp. 118–135, 2005.

[19] V. Shoup, “A proposal for an ISO standard for public key encryption (version 2.1),” in *IACR e-Print Archive*, 112, 2001.

[20] J. Hoffstein, J. Pipher, and J.H. Silverman, “NTRU: A ring-based public key cryptosystem,” in *Algorithmic Number Theory, Third International Symposium (ANTS-III'98)*, pp. 267-288, 1998.

[21] D. Coppersmith and A. Shamir, “Lattice attacks on NTRU,” in *Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pp. 52-61, 1997.

[22] C.P Schnorr and M. Euchner, “Lattice basis reduction: Improved practical algorithms and solving subset sum problems,” in

Mathematical programming, 66(1-3), pp.181-199, 1994.

[23] Y. Chen and P.Q. Nguyen, "BKZ 2.0: Better lattice security estimates," in *International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT)*. pp. 1-20, 2011.

[24] A. Becker, L. Ducas, N. Gama, and T. Laarhoven, "New directions in nearest neighbor searching with applications to lattice sieving," in *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*, pp. 10-24, 2016.

[25] D. Micciancio and M. Walter, "Fast lattice point enumeration with minimal overhead," in Proceedings of *the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*, pp. 276-294, 2014.

[26] M.R. Albrecht, B.R. Curtis, A. Deo, et al, "Estimate all the {LWE, NTRU} schemes!" in *Security and Cryptography for Networks - 11th International Conference (SCN)*, pp.351-367, 2018.

[27] N. Howgrave-Graham, J.H. Silverman, and W. Whyte, "A meet-in-the-middle attack on an NTRU private key". Technical report, NTRU Cryptosystems, 2003.

[28] N. Howgrave-Graham, "A Hybrid Lattice-Reduction and Meet-in-the-Middle Attack Against NTRU", in *Annual International Cryptology Conference (CRYPTO)*, pp. 150-169, 2007.

[29] M. L. Furst and R. Kannan, "Succinct certificates for almost all subset sum problems", in *SIAM Journal on Computing*, pp. 550-558, 1989.

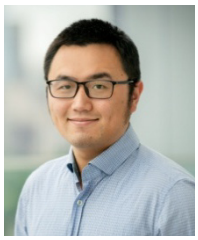
[30] P.N. Klein, "Finding the closest lattice vector when it's unusually close," in Proceedings *ACM-SIAM Symposium on Discrete Algorithms*, pp. 937-941, 2000.

[31] M.R. Albrecht, S. Bai, and L. Ducas, "A subfield lattice attack on overstretched NTRU assumptions," in *Annual Cryptology Conference (CRYPTO)* , pp. 153-178, 2016.

[32] J.H. Cheon, J. Jeong, and C. Lee, "An algorithm for NTRU problems and cryptanalysis of the GGH multilinear map without a low level encoding of zero," in *LMS Journal of Computation and Mathematics*, 19(A), pp. 255-266, 2016.

[33] E. Kiltz, V. Lyubashevsky, and C. Schaffner, "A concrete treatment of fiatshamir signatures in the quantum random-oracle model," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pp. 552-586, 2018.

贺婧楠 于 2017 年在中国科学院大学信息安全专业获得博士学位。现任中国科学院信息工程研究所助理研究员。研究领域为格密码与可证明安全。Email: hejingnan@iie.ac.cn



张振飞 于 2014 年在 University of Wollongong 计算机专业获得博士学位。现任 Algorand 密码工程师。研究领域为实用格密码。Email: zhenfei@algorand.com