计算机科学与探索
Journal of Frontiers of Computer Science and Technology
1673-9418/2022/16(05)-1087-09
doi: 10.3778/j.issn.1673-9418.2011052

模指数外包方案ExpSOS的格基密码分析

郑云海[1],田呈亮[1,2+]
1. 青岛大学 计算机科学技术学院,山东 青岛 266071
2. 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 100093
+ 通信作者 E-mail: tianchengliang@qdu.edu.cn

摘　要:随着云计算的普及,外包计算作为一种重要的云服务形式,日益引起学术界与工业界的广泛关注。模指数操作作为一种耗时的基本密码运算广泛地应用于RSA、数字签名算法(DSA)等,其外包方案的设计得到了广泛关注和研究。当前基于单个云服务器的外包方案,大多需要在本地端执行一个小指数的模指数操作,一般地,该指数的大小决定了方案的效率,其机密性决定着方案的安全性。对Zhou等提出的一个单服务器模指数外包方案ExpSOS进行了唯密文安全性分析。通过将算法中底数与指数的机密性转换为求解模多项式的小整数解的问题,使用Coppersmith的格构造技术对ExpSOS方案潜在的弱密钥进行了全面分析,并分别估计了安全应用场景下方案适用的底数大小和方案中安全参数选取的规模,为该方案在实际应用中的安全部署提出了具体建议。最后,给出了数字签名标准推荐参数下的ExpSOS方案弱密钥攻击实例,证明了理论攻击的有效性。

关键词:外包计算;模指数;唯密文攻击;Coppersmith算法;格基约化
文献标志码:A　　**中图分类号:**TP309.7

Lattice-Based Cryptanalysis on Outsourcing Scheme of Modular Exponentiations ExpSOS

ZHENG Yunhai[1], TIAN Chengliang[1,2+]
1. College of Computer Science & Technology, Qingdao University, Qingdao, Shandong 266071, China
2. State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China

Abstract: With the popularity of cloud computing, outsourcing computing, as an important form of cloud service, has attracted more and more attention from academia and industry. As a time-consuming basic cryptographic operation, modular exponential operation is widely used in RSA, digital signature algorithm (DSA), etc. The design of its outsourcing scheme has received extensive attention and research. At present, most of the outsourcing schemes based on a single cloud server need to perform a small exponential operation on the local. Generally, the size of the exponential determines the efficiency of the scheme, and its confidentiality determines the security of the scheme. This paper gives ciphertext-only security analysis on Zhou et al's modular exponentiation outsourcing scheme ExpSOS. By converting the problem of recovering the base and exponent in their algorithm to the problem of finding small root of polynomial modular unknown divisors, this paper analyzes the potential weak keys of ExpSOS by

基金项目:国家自然科学基金(61702294);国家密码发展基金(MMJJ20170126);信息安全国家重点实验室开放课题(2016-MS-23)。
This work was supported by the National Natural Science Foundation of China (61702294), the National Development Foundation of Cryptography (MMJJ20170126) and the Open Research Project of State Key Laboratory of Information Security (2016-MS-23).
收稿日期:2020-11-18　　**修回日期:**2021-01-14

invoking Coppersmith's lattice-based construction technique, and estimates the size of the secure base and the size of the security parameters in the scheme. Further, the specific suggestions for the security deployment of the scheme in practical application are put forward. Finally, some practical attack examples of weak key in ExpSOS scheme are given, which confirms the effectiveness of the theoretical attack.

Key words: computation outsourcing; modular exponentiations; ciphertext-only attack; Coppersmith's method; lattice basis reduction

模指数运算又称模幂运算,其作为一种基本运算广泛应用于RSA密码体制[1]和DSA(digital signature algorithm)的签名算法[2]中。通常地,指数长 n 比特时大约需要执行 $1.5n$ 次模乘操作,这对于计算资源有限的本地设备来说代价十分昂贵。因此,研究模指数的安全外包具有重要的理论与现实意义。根据方案所基于的安全模型,现有的外包方案可分为两类:基于双服务器的外包方案和基于单服务器的外包方案。

基于双服务器的方案将输入的数据在本地端分成两部分并分别发送给两个互不串通的服务器,便于底数指数的逻辑拆分与加解密设计。Hohenberger等[3]首次给出了外包计算的安全模型的形式化定义并提出了基于双服务器模型的模指数外包协议。然而,在他们的协议中,服务器可以 $\frac{1}{2}$ 概率欺骗用户。随后,Chen等[4]基于一种新的底数与指数逻辑拆分方法,设计了一种新的双服务器模指数外包算法,改进了Hohenberger等的方案的效率,并将可验证概率提高到 $\frac{2}{3}$。Ye等[5]进一步优化了之前的底数与指数拆分方法,分别针对单模指数与双模指数运算提出了两种新的外包方案,在保持可验证性概率不变的情况下进一步提高了外包算法的效率。最近,Fu等[6]提出了一种新的高效、可验证的模指数外包算法,该算法利用EBPV生成的随机整数提高了效率,并且可验证性概率达到最优的1。总的来说,双服务器方案一般可以节省比单服务器方案更多的计算开销,但其安全性假设高,无法抵御服务器共谋攻击。

单服务器的外包方案只使用一个服务器,安全性假设低,不需要考虑共谋的情况,但单服务器的方案节省计算开销不如双服务器有优势,当前单服务器外包方案,本地端大多需要执行一个小秘密指数的模指数运算。一般地,该指数的大小决定了方案的效率,其机密性决定着方案的安全性。Dijk等[7]提出了第一个基于单个不可信服务器来加速模幂的算法,但是他们的算法不能同时保证底数和指数的安全。在ESORICS 2014中,Wang等[8]提出了一个同时保护底数与指数的安全外包模幂运算的通用算法。但是,它们算法的可验证概率仅为 $\frac{1}{2}$,且容易遭受格算法攻击[9]。随后,Ding等[10]设计了一种新的安全外包算法,用户能以 $\frac{119}{120}$ 的概率检测出云服务器的欺诈行为。Li等[11]与Fu等[12]相继考虑并设计了模数为合数时的模幂运算的外包算法,并声称他们算法可验证概率可以达到最优的1。然而Rangasamy等[13]分析了他们的方案,并给出了相应的伪造攻击。以上方案均未考虑模数的隐私性,Zhou等[14]设计了第一个同时保护底数、指数与模数的单服务器外包方案。最近,Chevalier等[9]全面比较分析了当前单服务器外包方案的优缺点,并给出了各种场景下的最优构建。

在文献[14]中,Zhou等提出了一种基于单个服务器的模指数外包方案ExpSOS,旨在安全高效地实现本地端模指数的计算。文中证明了其方案能保护本地端底数 u、指数 a 和模数 N 的机密性,并能以高概率检测出服务器端的恶意行为。最近,在文献[13]中,Rangasamy等模仿Chevalier等[9]的攻击,对ExpSOS进行了简略的安全分析,他们的攻击假设用户调用了两次外包算法且要求两次外包的模指数运算的指数相同,这样的攻击条件是十分严格的。本文对ExpSOS进行了更全面的评估。具体来说,本文的贡献可以概括如下:

(1)对方案进行了唯密文分析,指出了方案潜在的弱密钥。通过将方案弱密钥的求解转化为模线性多项式的小整数解问题,调用Coppersmith的格基构造求解算法,仅利用密文就可以在多项式时间内恢复方案中的多个私有参数。

(2)为避免弱密钥攻击,详细估计了安全应用场景下底数和方案中安全参数选取的规模。

(3)实验给出了数字签名标准推荐参数下ExpSOS方案的弱密钥攻击实例,验证了理论分析的正确性。

1 预备知识

1.1 常用符号及其含义

一般地，本文用小写加粗字母表示列向量。表1列出了文中常用的专用符号及数学函数。

表1 符号说明

Table 1 Symbol description

符号	说明
$\mathbb{R}^m$	m 维实数向量集
$\mathbb{Z}$	整数集
$\mathbb{Z}^n$	n 维整数向量集
det(·)	行列式函数
$\|\cdot\|$	欧氏范数
$\varphi(\cdot)$	欧拉函数
e	欧拉常数
lb	以2为底的对数

1.2 格与相关不变量

作为一种经典的数学对象，格在解决计算机科学中的许多计算问题，特别是在密码学领域发挥着重要的作用[15-18]。

定义1（格）有 n 个线性无关的 m 维向量 $\boldsymbol{b}_1\boldsymbol{b}_2\cdots\boldsymbol{b}_n\in\mathbb{R}^m$，由 $\{\boldsymbol{b}_1\boldsymbol{b}_2\cdots\boldsymbol{b}_n\}$ 张成的格 $\mathcal{L}$ 定义为：

$$\mathcal{L}=\mathcal{L}(\boldsymbol{B})=\left\{\sum_{i=1}^{n}x_i\boldsymbol{b}_i|x_i\in\mathbb{Z}\right\}=\{\boldsymbol{Bx}|\boldsymbol{x}\in\mathbb{Z}^n\},$$

其中，$\boldsymbol{B}=[\boldsymbol{b}_1\boldsymbol{b}_2\cdots\boldsymbol{b}_n]$ 是格 $\mathcal{L}$ 的一个基，称 m 为格的维数，n 为格的秩。若 $m=n$，则称之为满秩格。格 $\mathcal{L}$ 的行列式定义为 $\det(\mathcal{L})=(\det(\boldsymbol{BB}^{\mathrm{T}}))^{\frac{1}{2}}$。如果 $\mathcal{L}$ 满秩，则有 $\det(\mathcal{L})=|\det(\boldsymbol{B})|$。在本文中，只使用满秩格。格中最短向量的范数与格的行列式有如下关系：

定理1（Minkowski定理[19]）设 $\mathcal{L}$ 为秩为 n 的格，存在非零向量 $\boldsymbol{v}$ 使得：

$$\|\boldsymbol{v}\|\leqslant\sqrt{n}\det(\mathcal{L})^{\frac{1}{n}} \tag{1}$$

1982年，著名的LLL算法[20]提出，它可以在多项式时间内找到格 $\mathcal{L}$ 的一个由“短”向量组成的约化基，其具有以下性质：

定理2[21] 设 $\mathcal{L}$ 为秩为 n 的格，在多项式时间内，LLL算法可以输出一系列约化基向量 $\boldsymbol{v}_i$，$1\leqslant i\leqslant n$，满足：

$$\|v_1\|\leqslant\|v_2\|\leqslant\cdots\leqslant\|v_i\|\leqslant 2^{\frac{n(n-1)}{4(2+1-i)}}\det(\mathcal{L})^{\frac{1}{n+1-i}}$$

1.3 多项式范数

定义2（多项式范数）对于任意整系数多项式函数 $f:\mathbb{Z}^n\to\mathbb{Z}$，记 $f(x_1,x_2,\cdots,x_n)=f_1x_1+f_2x_2+\cdots+f_nx_n$，其范数定义为 $\|f(x_1,x_2,\cdots,x_n)\|=\sqrt{f_1^2+f_2^2+\cdots+f_n^2}$。

1.4 Howgrave-Graham定理

定理3（Howgrave-Graham定理[22]）设 $g(x_1,x_2,\cdots,x_n)\in\mathbb{Z}[x_1,x_2,\cdots,x_n]$ 为包含 w 个单项式的整系数多项式，若：

（1）$(y_1,y_2,\cdots,y_n)$ 为 $g(x_1,x_2,\cdots,x_n)=o\bmod p^m$ 的一组上界不超过正整数 $X_1,X_2,\cdots,X_n$ 的解，即 $g(y_1,y_2,\cdots,y_n)=0\bmod p^m$，$|y_1|\leqslant X_1,|y_2|\leqslant X_2,\cdots,|y_n|\leqslant X_n$。

（2）$\|g(x_1X_1,x_2X_2,\cdots,x_nX_n)\|<\dfrac{p^m}{\sqrt{w}}$，则 $g(y_1,y_2,\cdots,y_n)=0$ 在整数范围内成立。

2 Zhou等ExpSOS方案的回顾

最近，Zhou等[14]提出了一种底数、指数和模数均可变的外包方案ExpSOS以计算 $u^a\bmod N$。在ExpSOS中，模数 N 的隐私性基于大整数分解的困难问题，底数 u 使用环扩张技术隐藏，指数 a 通过欧拉定理隐藏。即给定3个大整数 N、u、a，它们的两方算法过程如下所示：

KeyGen：客户端 T 随机选择大素数 p 和五个整数 r、k_1、k_2、t_1、t_2，其中 $t_1,t_2\leqslant b$，b 是一个小的安全参数。

Blinding：客户端 T 首先计算 $L=pN$，然后执行以下逻辑拆分：

$$A_1=a+k_1\varphi(N) \tag{2}$$

$$A_2=t_1a+t_2+k_2\varphi(N) \tag{3}$$

$$U=(u+rN)\bmod L \tag{4}$$

并将 (A_1,A_2,U) 发送到服务器 S。

ServerCom：服务器计算并向客户端返回：

$R_1=U^{A_1}\bmod L$，$R_2=U^{A_2}\bmod L$

Verify&Recovery：客户端通过验证

$(R_1)^{t_1}u^{t_2}=R_2\bmod N$

是否成立来检查结果，若成立，则恢复结果为 $R_0=R_1\bmod N$。

3 对方案ExpSOS的弱密钥分析与改进

本章将给出Zhou等外包方案ExpSOS中潜在的安全威胁。确切地说，对此方案在使用不同参数的场景下进行了唯密文攻击，并对安全参数的规模给出了具体的估计。攻击中使用的主要技术是Herrmann和May的格基的求解模线性方程的小整数解的方法。

3.1 Herrmann&May方法

给定方程 $f(x_1,x_2,\cdots,x_n)=a_0+\sum_{i=1}^{n}a_ix_i$ 及未知整数

N 的倍数 L，求解 $f(x_1,x_2,\cdots,x_n)=0 \bmod N$ 的根在计算机科学界，尤其是在密码学中有重要的应用。Howgrave-Graham[23]和May[21]先后研究了单变量方程（即 $n=1$）并提出了求解 $N>L^{\beta}$ 情况下所有的小整数解的格基算法。2008年，Herrmann和May[24]总结概括了他们的结果。具体地，他们证明了以下结论：

定理4[24] 设 $\varepsilon>0$，L 是一个足够大的合数且有因数 N 使 $N>L^{\beta}$。设 $f(x_1,x_2,\cdots,x_n)\in\mathbb{Z}[x_1,x_2,\cdots,x_n]$ 为有 n 个变量的线性多项式。若满足：

$$\prod_{i=1}^{n}X_i\leqslant L^{1-(1-\beta)^{\frac{n+1}{n}}-(n+1)(1-\sqrt[n]{1-\beta})(1-\beta)-\varepsilon} \tag{5}$$

则可求得 $f(x_1,x_2,\cdots,x_n)=0 \bmod N$ 的解 $(y_1,y_2,\cdots,y_n)$，其中 X_i 为 y_i 的上界，即 $|y_i|\leqslant X_i=L^{\gamma_i}$。该算法的时间复杂度和空间复杂度分别是 $\mathrm{lb}\,L$ 和 $\left(\frac{e}{\varepsilon}\right)^n$，其中 e 为欧拉函数。

根据Herrmann和May[24]的分析，可以通过以下步骤将 $f(x_1,x_2,\cdots,x_n)$ 的小整数解问题转化为寻找格中短向量问题：

（1）通过乘上 $a_i^{-1} \bmod L$ 将方程 $f(x_1,x_2,\cdots,x_n)$ 转化为首一多项式 $f_1(x_1,x_2,\cdots,x_n)$。

（2）构造多项式集合：

$$g_{i_2,i_3,\cdots,i_n,k}=x_2^{i_2}x_3^{i_3}\cdots x_n^{i_n}f_1^{k}(x_1,x_2,\cdots,x_n)L^{\max\{t-k,0\}}$$

其中，$i_2,i_3,\cdots,i_n$ 和 k 是非负整数，使得：

$$k+i_2+i_3+\cdots+i_n\leqslant m,\ t=\left\lfloor 1-(1-\beta)^{\frac{1}{n}}m\right\rfloor$$

$$m=\left\lceil\frac{n\left(\frac{1}{n}(1-\beta)^{-0.278\,465}-\beta\ln(1-\beta)\right)}{\varepsilon}\right\rceil$$

显然，方程 $f_1(x_1,x_2,\cdots,x_n)=0 \bmod N$ 的根均适合方程 $g_{i_2,i_3,\cdots,i_n,k}(x_1,x_2,\cdots,x_n)=0 \bmod N^t$。

（3）对于 $1\leqslant i\leqslant n$，构造多项式 $h_i(x_1,x_2,\cdots,x_n)=\sum_{i_2+i_3+\cdots+i_n+k\leqslant m}z_{i_2,i_3,\cdots,i_n,k}g_{i_2,i_3,\cdots,i_n,k}(x_1,x_2,\cdots,x_n)$，$z_{i_2,i_3,\cdots,i_n,k}\in\mathbb{Z}$，其共有 $d=\binom{m+n}{n}$ 项。显然，方程 $f(x_1,x_2,\cdots,x_n)=0 \bmod N$ 的根都适合方程 $h_i(x_1,x_2,\cdots,x_n)=0 \bmod N^t$。

根据定理3，若要消去模数 N^t，需要找到合适的整数 $z_{i_2,i_3,\cdots,i_n,k}$ 使得 $h_i(x_1X_1,x_2X_2,\cdots,x_nX_n)$ 的范数小于 $\frac{N^t}{\sqrt{w}}$，可由LLL算法约化由 $g_{i_2,i_3,\cdots,i_n,k}(x_1X_1,x_2X_2,\cdots,x_nX_n)$ 系数向量生成的矩阵得到。根据定理2中LLL算法输出结果的性质及定理3，若LLL约化基 $\mathcal{L}$ 的前 n 个向量长度满足 $2^{\frac{d(d-1)}{4(d-n+1)}}\det(\mathcal{L})^{\frac{1}{d-n+1}}<\frac{N^t}{\sqrt{d}}$，则：

$$h_i(x_1,x_2,\cdots,x_n)=0$$

由于 N 未知而 $N\geqslant L^{\beta}$，即满足：

$$2^{\frac{d(d-1)}{4(d-n+1)}}\det(\mathcal{L})^{\frac{1}{d-n+1}}<\frac{L^{\beta t}}{\sqrt{d}} \tag{6}$$

（4）通过求解由 $h_i(x_1,x_2,\cdots,x_n)$ 组成的线性系统可以找到 $f(x_1,x_2,\cdots,x_n)=0 \bmod N$ 的小整数解，其中 $h_i(x_1,x_2,\cdots,x_n)(1\leqslant i\leqslant n)$ 是代数无关的。

3.2 对ExpSOS的攻击及对策

本节对方案中底数与指数的隐私性进行分析。基于方程（2）~（4），根据定理4，它们的隐私性可以转化为模多项式方程的小整数解问题。通过分析，指出了方案潜在的弱密钥，并给出了方案适用的底数的大小以及方案逻辑拆分中参数的安全取值范围。设ExpSOS方案中各参数的上界表示如表2所示。在模指数运算最常见的两个应用场景（RSA[1]与数字签名标准（digital signature standard，DSS）[2]）中，模数分别为两个大素数的乘积与一个大素数。本文的分析基于上述两种情景，分别对 N 为大素数和两个大素数乘积两种情况进行分析。

表2 变量上界

Table 2 Upper-bounds of variables

变量	上界	变量	上界
k_1	X_K	k_2	X_K
t_1	X_T	t_2	X_T
a	X_A	u	X_U
t_1a	X_{t_1a}	$t_1k_1-k_2$	$X_{t_1k_1-k_2}$
u_0	X_{u_0}	u_1	X_{u_1}

3.2.1 N 为素数时

当 N 为素数时，$\varphi(N)=N-1$，根据Zhou等方案的构造，模数 $L=pN$，由于模数的隐私性基于大整数分解的困难性，根据整数分解的困难性假设[17]，p 应约等于 N，因此假设 $N\geqslant L^{\beta}$ 且取 $\beta=\frac{1}{2}$。

（1）底数 u 的隐私性分析

根据等式（4），有：

$$U=u+rN+sL=u+rN+spN=u \bmod N$$

即 $u-U=0 \bmod N$，可得：

$$f=a_0+a_1x_1=0 \bmod N \tag{7}$$

其有根 u，这里 $a_0=-U$，$a_1=1$。基于定理4的分析，可得 $X_U<L^{\frac{1}{4}-\varepsilon}\approx N^{0.5-2\varepsilon}$，这表明当 $\mathrm{lb}\,u<(0.5-2\varepsilon)\mathrm{lb}\,N$ 时，ExpSOS无法保证底数 u 的隐私性。

现在考虑 $L^{\frac{1}{4}} \approx N^{0.5} \leqslant u \leqslant N$ 的情况，此时，记底数 u 为 $u = u_1 L^{\frac{1}{4}} + u_0$，其中 $u_0, u_1 < L^{\frac{1}{4}}$，则有 $u_1 L^{\frac{1}{4}} + u_0 - U = 0 \bmod N$。分析可得：

$$X_{u_0} X_{u_1} \leqslant L^{(1-(1-\beta)^{\frac{3}{2}} - 3(1-\sqrt{1-\beta})(1-\beta)-\varepsilon)} \approx N^{(0.414-2\varepsilon')}$$

这表明当 $u \geqslant N^{0.5}$ 且 $u_0 u_1 \leqslant N^{0.414-2\varepsilon}$ 时，ExpSOS无法保证底数 u 的隐私性。

综上，当外包的模指数运算中底数满足 $u \leqslant N^{0.5}$ 或 $u \geqslant N^{0.5}$ 且 $u_0 u_1 \leqslant N^{0.414-2\varepsilon}$ 条件时，ExpSOS方案不能保护底数 u 的隐私性。

（2）指数 a 的隐私性分析

由等式（2），有：

$$A_1 = a + k_1(N-1) = a + k_1 N - k_1$$

即 $k_1 - a + A_1 = 0 \bmod N$，可得多项式：

$$f(x_1, x_2) = a_0 + a_1 x_1 + a_2 x_2 = 0 \bmod N$$

有根 (k_1, a)，这里 $a_0 = A_1$，$a_1 = 1$，$a_2 = -1$。由定理4，可得：

$$X_k X_A < L^{1-(1-\beta)^{\frac{3}{2}} - 3(1-\sqrt{1-\beta})(1-\beta)-\varepsilon} \approx N^{0.414-2\varepsilon}$$

即当 $\mathrm{lb}\, k_1 a < (0.414 - 2\varepsilon)\mathrm{lb}\, N$ 时，ExpSOS无法保证 k_1, a 的隐私性。

由等式（3），有：

$$A_2 = t_1 a + t_2 + k_2(N-1) = t_1 a + t_2 - k_2 + k_2 N$$

可得 $k_2 - t_1 a - t_2 + A_2 = 0 \bmod N$，即多项式：

$$f = a_0 + a_1 x_1 + a_2 x_2 + a_3 x_3 = 0 \bmod N$$

有根 $(k_2, t_1 a, t_2)$，这里 $a_0 = A_2$，$a_1 = 1$，$a_2 = -1$，$a_3 = -1$。根据定理4得：

$$X_K X_{t_1 a} X_T < L^{1-(1-\beta)^{\frac{4}{3}} - 4(1-\sqrt[3]{1-\beta})(1-\beta)-\varepsilon} \approx N^{0.38-2\varepsilon}$$

即当 $\mathrm{lb}\, t_1 t_2 k_2 a < (0.38 - 2\varepsilon)\mathrm{lb}\, N$ 时，ExpSOS无法保证 $k_2, t_1 a, t_2$ 的隐私性。同时由于 $|t_1|, |t_2| < b$，当 $\mathrm{lb}\, k_2 a < (0.38 - 2\varepsilon)\mathrm{lb}\, N - 2\,\mathrm{lb}\, b$ 时，ExpSOS同样无法保证 $k_2, t_1 a, t_2$ 的隐私性。

由等式（2）、（3）综合可得：

$$t_1 A_1 - A_2 = (t_1 k_1 - k_2)(N-1) - t_2$$

即 $t_2 + t_1 A + t_1 k_1 - k_2 - A_2 = 0 \bmod N$，将 $t_1 k_1 - k_2$ 看作一个变量，可得多项式：

$$f = a_0 + a_1 x_1 + a_2 x_2 + a_3 x_3 = 0 \bmod N$$

分析可得，若：

$$\mathrm{lb}\,(t_1 t_2 (t_1 k_1 - k_1)) < (0.38 - 2\varepsilon)\mathrm{lb}\, N$$

则ExpSOS无法保证 t_1, t_2 的隐私性。

3.2.2 N 为合数时

假设 N 是两个大素数的乘积，即 $N = qq'$，且 $q \approx q' \approx N^{\frac{1}{2}}$，有 $\varphi(N) = (q-1)(q'-1)$。本小节中 $\mathrm{lb}\,(q + q' - 1) \approx \mathrm{lb}\, q$。根据Zhou等方案的构造，模数 $L = pN$，由于模数的隐私性基于大整数分解的困难性，根据整数分解的困难性假设，有 $p \approx q \approx q' \approx L^{\frac{1}{3}}$，因此假设 $N \geqslant L^{\beta}$ 且取 $\beta = \frac{2}{3}$。

（1）底数 u 的隐私性分析

与之前的分析类似，可以得到界：

$$X_U < L^{\beta^2 - \varepsilon} \approx N^{\frac{2}{3} - \frac{3}{2}\varepsilon}$$

即若 $\mathrm{lb}\, u < \left(\frac{2}{3} - \frac{3}{2}\varepsilon\right)\mathrm{lb}\, N$，ExpSOS无法保证底数 u 的隐私性。当 $u \geqslant N^{\frac{2}{3}}$ 时，若 $u_0 u_1 \leqslant N^{0.577-1.5\varepsilon}$，则ExpSOS无法保证底数 u 的隐私性，其中有：

$$u = u_1 L^{\beta} + u_0,\ u_0, u_1 \leqslant L^{\beta}$$

（2）指数 a 的隐私性分析

由等式（2），有：

$$A_1 = a + k_1 N - k_1(q + q' - 1)$$

即 $a - k_1(q + q' - 1) - A_1 = 0 \bmod N$，可得 $f(x_1, x_2) = a_0 + a_1 x_1 + a_2 x_2 = 0 \bmod N$，根为 $(a, k_1(q + q' - 1))$，这里 $a_0 = -A_1$，$a_1 = 1$，$a_2 = -1$，$a_3 = -1$。根据定理4的分析过程可得：

$$X_A X_K (q + q' - 1) < L^{1-(1-\beta)^{\frac{3}{2}} - 3(1-\sqrt[2]{1-\beta})(1-\beta)-\varepsilon} \approx N^{0.577-1.5\varepsilon}$$

即若

$$\mathrm{lb}\, k_1 a \leqslant (0.577 - 1.5\varepsilon)\mathrm{lb}\, N - \mathrm{lb}\, q \approx (0.077 - 1.5\varepsilon)\mathrm{lb}\, N$$

ExpSOS无法保证指数 a 的隐私性。

由等式（3）有：

$$A_2 = t_1 a + t_2 + k_2 N - k_2(q + q' - 1)$$

即 $t_1 a + t_2 - k_2(q + q' - 1) - A_2 = 0 \bmod N$，可得多项式 $f = a_0 + a_1 x_1 + a_2 x_2 + a_3 x_3 = 0 \bmod N$，其有根 $(t_1 a, t_2, k_2(q-1))$，这里 $a_0 = -A_1$，$a_1 = 1$，$a_2 = -1$，$a_3 = -1$。根据定理4，由于 $|t_1|, |t_2| < b$，分析可得，当 $\mathrm{lb}\, k_2 a \leqslant (0.04 - 1.5\varepsilon)\mathrm{lb}\, N - 2\,\mathrm{lb}\, b$ 时，ExpSOS无法保证指数 $t_1 a, t_2, k_2(q-1)$ 的隐私性。

由等式（2）、（3）综合得：

$$t_1 A_1 - A_2 = (t_1 k_1 - k_2)(q-1)(q'-1) - t_2$$

即 $t_2 + t_1 A_1 + (t_1 k_1 - k_2)(q + q' - 1) - A_2 = 0 \bmod N$。将 $(t_1 k_1 - k_2)(q + q' - 1)$ 看作一个变量，则有多项式 $f = a_0 + a_1 x_1 + a_2 x_2 + a_3 x_3 = 0 \bmod N$，这里 $a_0 = -A_2$，$a_1 = 1$，$a_2 = A_1$，$a_3 = 1$。基于定理4分析可得，若：

$$\mathrm{lb}\,(t_1 t_2 (t_1 k_1 - k_2)) \leqslant (0.54 - 1.5\varepsilon)\mathrm{lb}\, N - \mathrm{lb}\, q \approx (0.04 - 1.5\varepsilon)\mathrm{lb}\, N$$

ExpSOS无法保证 t_1, t_2 的隐私性。

3.2.3 规避攻击的对策

基于3.2.1小节与3.2.2小节安全性分析结果，为避免本文提出的弱密钥攻击，表3详细总结了不同场景下方案安全参数的选取范围。

表3 安全参数的选取范围

Table 3 Selection range of security parameters

安全参数	N 为素数	N 为合数
$\mathrm{lb}\,u$	$\mathrm{lb}\,u \geqslant 0.5\,\mathrm{lb}\,N$, $\mathrm{lb}\,u_0u_1 \geqslant 0.414\,\mathrm{lb}\,N$	$\mathrm{lb}\,u \geqslant \frac{2}{3}\mathrm{lb}\,N$, $\mathrm{lb}\,u_0u_1 \geqslant 0.577\,\mathrm{lb}\,N$
$\mathrm{lb}\,u_0u_1$		
$\mathrm{lb}\,k_1a$	$\mathrm{lb}\,k_1a \geqslant 0.414\,\mathrm{lb}\,N$	$\mathrm{lb}\,k_1a \geqslant 0.077\,\mathrm{lb}\,N$
$\mathrm{lb}\,t_1t_2k_2a$	$\mathrm{lb}\,t_1t_2k_2a \geqslant 0.38\,\mathrm{lb}\,N$	$\mathrm{lb}\,t_1t_2k_2a \geqslant 0.04\,\mathrm{lb}\,N$
$\mathrm{lb}\,k_2a$	$\mathrm{lb}\,k_2a \geqslant 0.38\,\mathrm{lb}\,N - 2\,\mathrm{lb}\,b$	$\mathrm{lb}\,k_2a \geqslant 0.04\,\mathrm{lb}\,N - 2\,\mathrm{lb}\,b$
$\mathrm{lb}\,(t_1k_1 - k_2)$	$\mathrm{lb}\,(t_1k_1 - k_2) \geqslant 0.38\,\mathrm{lb}\,N - \mathrm{lb}\,t_1t_2$	$\mathrm{lb}\,(t_1k_1 - k_2) \geqslant 0.04\,\mathrm{lb}\,N - \mathrm{lb}\,t_1t_2$

4 攻击实例

本章以对方案ExpSOS的分析为例给出了攻击实例。在本文的例子中使用的所有参数均参考NIST的数字签名标准[2]以及RSA算法标准[1]。实验使用Wolfram Mathematica 11.3完成，运行环境为Windows 10，Intel Core™ i5-6500 3.2 GHz，8 GB RAM。攻击实验结果如表4和表5所示。

首先描述 N 为素数的攻击实例。取模数 N 为3 072位的大素数，基数 u 为1 024位整数，指数 a 为256位整数。在对ExpSOS进行仿真之后，表4列出了云服务器拥有的参数及攻击结果 u。由于对其他方程的攻击具有极高的时间和空间复杂度，这里只给出基于方程(4)和方程(2)的底数 u 以及指数 a 的

表4 N 为素数时的实验参数及结果

Table 4 Experimental parameters and results while N is prime

名称	数值
L	0x7810B36D 4CDDCD6D DE8AEFA5 E8B09DC4 094CA47D 117907B6 BB3256DD 78EEF3F1 BA78FE91 5BAEC309 4D69DE5A 7603037D C09D16D4 A72C8258 808AC8AD 7D63296B B65F9F45 B20745E6 DC1C894C FF221967 A9F4948A 77DA3976 9BE92EB5 2CFC07C0 B0CD5238 658E5C2D 582EAEFA 071E4C64 2FB4F5B4 3BE3C169 0BD03E75 71731B39 D816CD71 72201474 45DC8C79 1A218E06 EF794F68 0EF0D2D3 D712D41B 0D6C86AB 0ED9BD4F 4E328531 DEEE8FA8 E93B5371 F04C8EFE D6C65A16 7E07713E 694B4918 DAB2439C 6838D90A E5FEF426 4C12557F 86319DD0 817C8ACD 72292C7B 3BF55C37 184E04F3 8ABE45CE C790245E 12CFA0E3 C2AD7805 E4DC1AE3 ECC0D96F 050E089D 44AE275C BAAD5378 E64625DE 3178547 5518E765 FF6C2CBE 667B7E39 51B8587B AD49701F 4163C222 AF8D5959 CF253E32 9274A53E 000634FD 786A0A87 62E2D9BF 2177965A 8C1873AC 2E96ABAA D75785B9 4561C100 1CE4E7B9 A603C0BF A0376FB8 D1871A32 80773432 177D3ED0 9A779A6F 25EF86F3 231B07A2 51D8D75E EBB719D3 1BCE8EFD 00D36A65 52BB9BE4 0EA453D8 4876B7D4 769DA3DA D358D42B 84FFD5A0 C0AC4FB6 63FFFDFD B28D1DF2 AEC677E7 169BEF14 12F70130 5298B5C3 2916ACFB 96859E64 7679B3CD 124DF94E 68594C21 DFE7DC79 4A58F093 07151AD6 BA04871D 28B52789 85740ACA 487956B7 F83E9001 9DD62BB0 37411D9F E75E4481 D0B251DA 1100F01A 9AF24F2F 2C02A2BB F28BFC31 3395A495 63763FAD 8A25AAB0 2CE8C0D9 2B3C0B36 C4FCBEE0 B56C0952 60F264BA B55F6C84 C0716294 510D644C 22A1CFB7 ABA77416 2F5C4789 0B9B9990 25B901DC EA838AB7 6DF1A602 3985142B 66BFACD0 17964990 3A2D2E18 E624247D B3E3D9BA C319197C 553E31D8 F3D75E65 8556A972 DC588FCB 1D4F7712 087241C1 9F535365 A59563B3 7E587186 F0A8B7F8 27104D45 3CEA7A18 C9469FF4 A8C92621 FCAD85CE 1053A568 8BF0F5EC 3581208F 5F40FB99 E1997B76 3055ABDF 490937FC FDB10EBE 30DB293B 96B8F948 51D1395F 711FA945 D919499E 7146936E F17A5166 C0FF39DF 04748203 F3E0CD0A 2AEB73D0 EBE40BC5
U	0x5FC6F7AB D3978738 A06742E5 75B3CDFF 7BDA7C91 5336F771 A7C46FD2 9A87B0FE E0AA810A 48C6F11E 1D6A10B9 C5DA251A 3C49B831 517FF01B DCDE01F7 A7F8321E 1747DE98 F77DFF4B A5052CB9 51629AA0 52A6F3D8 73C1FEFF F9690E2D E94C4ACA 9D568E52 55AEAB51 E964FEB7 0D2A6875 D5CAAB7E 82A63305 97B602E8 ED33AD0E EF9A90E1 F188A8D3 C2E01CE9 0D6F65EA 9B2881C0 01994011 03B84B6C DB129ACD DAF2124F 0965CC33 42E95682 A5308C4C 58AAD1D2 A91439FB DFED8CCE EEEF2FE6 CFBF15BE F497F000 41D2F60C C8CF2E96 4CE82B30 902DF17D 6E125EFA 1C5BD841 E66147D3 8A1AB88A 61541542 5A133023 5554E980 95637068 5919DE2B 793A0D1D B8D99B84 F2253BD3 351AD8FD F715AFD1 6A43A533 CB01ECCB 41E532D1 7B4D72A4 5F00CCB1 27CE38A1 19866557 B877B29C BD921E95 78570525 170E50E2 A0A3E391 04F024AE D1192AA D0527A2 18FB9C1C B40C15D1 9B79DEE3 F1A241A6 409623B9 7CEE9008 5C8C8A50 5CD12879 C510C360 4CB4459B 181F373A 78520CC7 87FD767C 1FD4C6B6 6249CE55 6691633A 7DFB3C3F BBD56DA4 FD489DF9 CB05631E F6293B4B 30DCDED5 8BBE87A8 AB0542EB AFF05ECC F7A8000F 273E3DD7 66D57C61 77296749 EAA49BDD 00B4F8F4 B1A6B0CF 0120DA9E 6FADCB8A 8D87A849 2E5613B8 AE8B1D7C 0F168027 BD6CD0EF 74A47FBD 7C91C0DB E2FC5F25 1F517BE1 353FD0BE 1C1EDABC 1D067DAC 8EFEB536 DEEF9DD6 23CD4CCF 57A60697 EC3716DB C244D8E1 F89BF6D3 B00915A0 7B3286EA C8AB79AC 35B96F5E CC6D7172 41296197 DB61BF78 3DC6DB48 2D8AAD43 9B31B6EB C6287817 010E571F 47A8D947 CFA160DE 398EDDD5 AAA6C489 5A7B0B95 6F962A9D 6629E8AB 5A1747FD 805A7A95 05BC2661 93EDD35B 1CF82140 FCD53A05 341B9B74 A7321895 86D409D2 B248D1A8 461DD94A 2BEAEB88 48F2DBE4 D42BE0BE 601B516F DCA03DE3 C69C17BB 13B6914A 7CBAC4AF B66160B0 9AB06327 545B8C6D 9838C036 0A9E2EED 056376AC 778221A8 C83D5813 D3780DF9 F425E2D1 9A83DDE9 426CE764 3206166C B476D23A 82BD2050 94A3BBE6 860F51D2 01D4CB68

表4 (续)

名称	数值
A_1	0x86AC816C A208E159 A6C15DEC 985F2B39 32F7491B AA67C602 47F24157 9E99229 470CC3A7 6759065D E14D01F8 79CC50A6 CF8AE298 1149D2C9 39E936B1 0AA5FB4D 4CE18790 4918028D B81D1B08 16A1F57F 6B20E02B 8A8113A0 2C464291 18137CBD 58B857B7 F4CC016B 1D66C6D7 CD77BBF4 8FF25C0E 74DFE992 719D2F5 E20390FD BF5173D4 BAB558C7 308687EE 13275A36 13B06231 6878C5BE 8F2147C8 D54D60D3 E1B3D200 58896D9D 46A89FA6 A8ED8759 E2F69C21 74AE840A 73717A46 F8032E67 50691098 73EB53C0 0C979D84 C73428FF 9979852B 042433D2 AD90A0E6 4A87E3F6 BA8AF0ED F89D224C 2CEBBD42 726201AC 26A5B1D2 015A5D3E 2112CAC ACC9104 7AEEB973 ACE66A96 6FE3CD60 1D2D36C1 9AE31AA4 EEF751A 74C1855B E4C30D31 6B8DB258 3960AC01 E6F1AFBA FA400309 B8DACA2D 5F1D69B 73A79B1 6A344384 CDD69C88 AFB7B32B 949BE578 74A4C71D 75F48C35 1EC68F31 8A22FE86 D59AB0DD 2ECF14DF 67341921 61C3DD3C 3B78E830 FEA2B76 CBC5F168 3E474E8B 0C9021AB D3FC
X_A	0x87AE326D 95367438 F5601AD2 8961E264 87730841 3952EA1C DC742402 7EEDC8AD
X_U	0xD71CE05B F55FA8D9 7B26C1A2 374EB24F 7D173A5C B4031776 1827982C 20D7A85 37640DCB D1574716 163CB7C7 4DF3C271 09B1AB20 0F837E5E 913C0CBD 5BE4B4AC AF342C44 9EB60B04 7E8FD56C 5D854186 F012A17C 338CFA2D 56B3AD4 45465B04 0ACB9690 9AC59B2B FDA1448C 5EA069E0 5056E57D 62E2FE51 F36E68E8 060DFD97
X_K	0xD0A9
u	0xD71CE05B F55FA8D9 7B26C1A2 374EB24F 7D173A5C B4031776 1827982C A20D7A85 37640DCB D1574716 163CB7C7 4DF3C271 09B1AB20 0F837E5E 13C0CBD 5BE4B4AC AF342C44 9EB60B047E8FD56C 5D854186 F012A17C 338CFA2D B56B3AD4 45465B04 0ACB9690 9AC59B2B FDA1448C 5EA069E0 5056E57D 62E2FE51 F36E68E8 060DFD96

表5 N 为合数时的实验参数及结果

Table 5 Experimental parameters and results while N is composite

名称	数值
L	0xB95DE788 E63A90C9 A5E9DEEF 5B0FAECB 1961D18E 58CFBACC DAE90596 457702CB 4D221FAF CDFC4081 5C441225 ADF36468 BAAB2590 7CA49D3C 8DC1AC97 97911179 DDF67911 5304D649 1B794894 3F6F2C76 17B90055 040AE021 8C70F859 C61DDB2A 16A0996E 8EB96FB2 0E20C388 A6A9FA1A 10690DF9 6212D7A3 C7C2D11F C80E9DE8 69CC2589 23FF5408 6AD49D60 2538DB62 CEAD6375 2CDF252F C5B88B34 AF2173E3 0A618BEC 2158BA97 682A9C99 3FBF0180 23378C0C D8EA2996 6ECF23F1 28751D1F 9181FF87 9D99C199 EBF7BC95 B020460A B0328CBC 797FB429 43ACFD6C D6068C87 CF345E1E 2239FF1A 532A364F E3977637 14476173 C598766D 56945F61 3497DF01 B83652A6 EF59EB90 165DA127 82DC369C 2ED4BE53 747AA583 1371676B F18CEEC8 1F6F5AE7 EF75656F 8D5A49A7 46BDF136 474972B1 815F930C 55986E3D C4D113AE EEC08FBC 15F5F63C FDA2BB52 7DA6534F EDC4A45F 01301DAA D95BD794 BBA9BD26 EB09EF3C 6D6F8F35 9D338DFC 59AC56C2 0E95B660 AB52FC19 4E9980A3 335EF915 2F250066 CC51DE49 7F781C91 DC9A453D 3DD245C8 6BFEDACE 22FB1DD6 5C58E12C 17BB169F C50C3F73 EB0BEA83 A53CBFB8 4E8667E6 D0403CE4 35B37F6A 47CE5A33 1DA0DFF8 4C05F3C4 69541B3C 46D4D062 499C0C73 42B87611 CC56FDC5 23EB724B 39419D6E EB22F142 CF1EA13A A2D84435 573C564D 4D67725C 779B9708 408A9F70 52225D36 FB24642D 1DF297A0 C1C31952 668070C0 C97D97A3 0E1B2740 CF3943FC 7CAD1BC4 2D130DE2 57B31889 34983779 0A4865F2 A5EF3F6E 8F9273B3 4B9BBA2D
U	0x72C10C71 160F1A78 F2928B99 FAF8B277 F5FAE527 969CA911 FC642E94 CA117C15 EADDD11F E3C0F506 6589095F 6AA90772 290F6BB3 AC0BC130 415B0662 553809D0 D8DCD942 DA9E006C 61483AAC 69FCE190 22BF6D00 A9BCCDEA 6028D270 6003A6DF C975BBA1 A972CCD2 8EB9F918 1FBFAC09 957E5FD1 46D21E23 C14DCE2F 3C86DEFC 46628BCD 0FE6B666 BC691297 54BAC344 5E7174A3 45B31F6F 5B0B6F7F 19A0D0BF 6EEA4413 F92AAE0C 695A7005 F468EC49 2069E04C C659D8DF 853F75A9 DFC38FDD 2EB5323C 12387507 35A9815F 94129124 B3EC051E 891422AE B27F6FF6 CEE85C50 08742C57 D47BE74C C247E472 D32EA652 8F2B9FA4 6E36903C F946F4E4 2FAF785D 7E6CD6C5 ACB9E4D2 515F63BC 5E617F7A 6AE423AB CF9B883F 2CF0DD30 D3ED1C43 2B326AE7 41D44A96 F114D803 6BEEB5A4 23E2B2AF 10A366BE A4515226 16D16332 5511EEAA DC1F4726 B00DD6E8 AF3D09EC 7FC7F413 FBB9AA87 29FA5D6C 9E7B4465 9F017427 9A6389E6 D335EF64 45EF330E 79F3F5D8 770C84AF C46A408D 4F45B277 658EEC37 88056CD6 65707B84 02D07034 A55CBE05 EE8E5A88 4B930B90 29B9C7B5 4771E0FC D756FF18 9402DD5E 299D2845 2C78FF0D 85101937 E4A45936 D000F055 DEFFA827 357C117E B1988642 2FD692A3 FED86571 50C6C26D AE352543 2D770322 836D6F35 CC9B44C4 D5C9CFE0 3CE52F12 AF9D5B9D 642BD0AE 3C65A778 CECD0B5E 850D00A7 831B2020 2D94CFEA 1B0AF9C2 40221EEF B638B121 1DBB34B1 8A43E580 0824C702 9DD90B5F ECDF5421 E36A105C C2082DF9 D9D1B496 B0B03CCA 3EB34004
u	0xF56D7E1C 8880F5D3 44FDE7B8 A07A4844 4E0407AA CA05B7C7 8C33AA8B B7F36363 75BB1497 F13CD439 565683B3 ABEF8FFE 7F74801A 5DC1A929 F453B9C9 F2DA8F02 DA624F5E 0BCDA365 8CA04AD9 8E8DB1AE E60FBDD6 6E000151 ACB89F9D 0207DA70 7EB8F3F8 0B1EA00C 8AFD6A76 33311708 2943A693 4FF68667 997EB823 17AA9A28 8BC1F0A6 128CBA90 E8C9EA56 969C5F13 33A0E89F D8CC0522 33D1BE84 124786C4 3D42A36E 18254FD2 177B93DA 08D7FAEE F7CD8EE6 5CD8FEA6 CF5302DB DEE2CB79 E3F8E855 30EB2FCA 123C0327 133A59DA D540ABAE 57DE0F0B 4C629CB7 EC8EB886 79C7DC39 4D6ACC88 EB0C482A 7A1AA619 B180BFBA 955A23D7 FDF60410 4C6B9046

恢复攻击实例。根据对等式(4)的分析过程以及实验硬件设备性能,为了平衡空间复杂度和攻击的时间复杂度,取 $\varepsilon=0.05$,计算可得 $m=15, t=7, d=16$,构造对应多项式的系数矩阵,通过攻击可以恢复底数 u 如表4所示。该攻击实例敌手约化格基的时间成本小于2 min,求解方程组的时间小于0.1 s,总时间合计不超过2 min。

根据对等式(2)的分析以及实验硬件设备性能,为了平衡空间复杂度和攻击的时间复杂度,取 $\varepsilon=0.2$,得 $m=8, t=2, d=45$,构造对应多项式的系数矩阵,通过分析可得如下含有 (a,k) 两个未知数的线性多项式,且总能找到所需的根:

$$\frac{a}{X_A}=\frac{T_1+T_2\frac{k_1}{X_K}}{T_3}$$

其中

T_1 =0x1362073439E3352CB556DF42A5E9697C1359937708306A9668A2E092A46AFEDC

T_2 =0x1DCF

T_3 =0x1362073439E3352CB556DF42A5E9697C1359937708306A9668A2E092A46B1CAB

该攻击实例敌手约化格基的时间成本小于2 min,求解方程组的时间小于0.1 s,总时间合计不超过2 min。

下面给出 N 为合数时的一个攻击实例。表5列出了使用的参数及结果。同样由于复杂度的问题,只给出底数 u 的恢复攻击。分别取两个1 536位的大素数 q 、q' ,生成模数 $N=qq'$,选择基数 u 为2 048位。对ExpSOS进行仿真之后,根据对等式(4)的分析过程以及实验硬件设备性能,为了平衡空间复杂度和攻击的时间复杂度,取 $\varepsilon=0.05$,有 $m=24, t=16, d=25$,构造对应多项式的系数矩阵,通过攻击可以恢复底数 u 如表5所示。

该实例中敌手约化格基的时间成本小于15 min,求解方程组的时间小于1 s,总时间合计不超过15 min。

5 结论

本文对Zhou等在IEEE TIFS 2017提出的模指数外包方案ExpSOS的安全性进行了全面的分析。通过格基约化技术对其方案进行攻击,并对其中的参数选择提出了建议以抵抗这种攻击。

参考文献:

[1] MORIARTY K M, KALISKI B, JONSSON J, et al. PKCS #1: RSA cryptography specifications version 2.2[J]. Internet Engineering Task Force, Request for Comments, 2016, 8017: 1-78.

[2] KERRY C F, GALLAGHER P D. Digital signature standard (DSS)[R]. Federal Information Processing Standards, 2013.

[3] HOHENBERGER S, LYSYANSKAYA A. How to securely outsource cryptographic computations[C]//LNCS 3378: Proceedings of the 2nd Conference on Theory of Cryptography, Cambridge, Feb 10-12, 2005. Berlin, Heidelberg: Springer, 2005: 264-282.

[4] CHEN X, LI J, MA J, et al. New algorithms for secure outsourcing of modular exponentiations[J]. IEEE Transactions on Parallel and Distributed Systems, 2013, 25(9): 2386-2396.

[5] YE J, XU Z, DING Y. Secure outsourcing of modular exponentiations in cloud and cluster computing[J]. Cluster Computing, 2016, 19(2): 811-820.

[6] FU S, YU Y, XU M. A New efficient algorithm for secure outsourcing of modular exponentiations[J]. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, 2020, 103(1): 221-224.

[7] VAN DIJK M, CLARKE D, GASSEND B, et al. Speeding up exponentiation using an untrusted computational resource [J]. Designs, Codes and Cryptography, 2006, 39(2): 253-273.

[8] WANG Y, WU Q, WONG D S, et al. Securely outsourcing exponentiations with single untrusted program for cloud storage[C]//Proceedings of the 19th European Symposium on Research in Computer Security, Wroclaw, Sep 7-11, 2014. Cham: Springer, 2014: 326-343.

[9] CHEVALIER C, LAGUILLAUMIE F, VERGNAUD D. Privately outsourcing exponentiation to a single server: cryptanalysis and optimal constructions[J]. Algorithmica, 2021, 83(1): 72-115.

[10] DING Y, XU Z, YE J, et al. Secure outsourcing of modular exponentiations under single untrusted programme model [J]. Journal of Computer and System Sciences, 2017, 90: 1-13.

[11] LI S, HUANG L, FU A, et al. CExp: secure and verifiable outsourcing of composite modular exponentiation with single untrusted server[J]. Digital Communications and Networks, 2017, 3(4): 236-241.

[12] FU A, LI S, YU S, et al. Privacy-preserving composite modular exponentiation outsourcing with optimal checkability in single untrusted cloud server[J]. Journal of Network and Computer Applications, 2018, 118: 102-112.

[13] RANGASAMY J, KUPPUSAMY L. Revisiting single-server algorithms for outsourcing modular exponentiation[C]//LNCS 11356: Proceedings of the 19th International Conference on Cryptology in India, New Delhi, Dec 9-12, 2018.

Cham: Springer, 2018: 3-20.
[14] ZHOU K, AFIFI M H, REN J. ExpSOS: secure and verifiable outsourcing of exponentiation operations for mobile cloud computing[J]. IEEE Transactions on Information Forensics and Security, 2017, 12(11): 2518-2531.
[15] KANNAN R. Algorithmic geometry of numbers[J]. Annual Review of Computer Science, 1987, 2(1): 231-267.
[16] AJTAI M. Generating hard instances of lattice problems[C]// Proceedings of the 28th Annual ACM Symposium on the Theory of Computing, Philadelphia, May 22-24, 1996. New York: ACM, 1996: 99-108.
[17] BONEH D. Twenty years of attacks on the RSA cryptosystem[J]. Notices of the AMS, 1999, 46(2): 203-213.
[18] REGEV O. On lattices, learning with errors, random linear codes, and cryptography[J]. Journal of the ACM, 2009, 56(6): 1-40.
[19] MINKOWSKI H. Geometrie der Zahlen[J]. Monatsh. f. Mathematik und Physik, 1910, 22: A30.
[20] LENSTRA A K, LENSTRA H W, LOVÁSZ L. Factoring polynomials with rational coefficients[J]. Mathematische Annalen, 1982, 261: 515-534.
[21] MAY A. New RSA vulnerabilities using lattice reduction methods[D]. Paderborn: University of Paderborn, 2003.
[22] HOWGRAVE-GRAHAM N. Finding small roots of univariate modular equations revisited[C]//LNCS 1355: Proceedings of the 6th International Conference on Cryptography and Coding, Cirencester, Dec 17-19, 1997. Berlin, Heidelberg: Springer, 1997: 131-142.
[23] HOWGRAVE-GRAHAM N. Approximate integer common divisors[C]//LNCS 2146: Proceedings of the International Conference on Cryptography and Lattices, Providence, Mar 29-30, 2001. Berlin, Heidelberg: Springer, 2001: 51-66.
[24] HERRMANN M, MAY A. Solving linear equations modulo divisors: on factoring given any bits[C]//LNCS 5350: Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security, Melbourne, Dec 7-11, 2008. Berlin, Heidelberg: Springer, 2008: 406-424.

郑云海（1995—），男，山东潍坊人，硕士研究生，主要研究方向为云计算安全、密码学。
ZHENG Yunhai, born in 1995, M.S. candidate. His research interests include cloud computing security and cryptography.

田呈亮（1983—），男，山东莱芜人，博士，副教授，主要研究方向为基于格的密码学、云/边缘计算中的隐私保护问题。
TIAN Chengliang, born in 1983, Ph.D., associate professor. His research interests include lattice-based cryptography and privacy preservation problems in cloud/edge computing.