可证明安全的抗量子两服务器口令认证密钥交换协议

尹安琪[1]，郭渊博[1]，汪定[2,3]，曲彤洲[1]，陈琳[1]
（1. 信息工程大学密码工程学院，河南 郑州 450001；2. 南开大学网络空间安全学院，天津 300350；
3. 南开大学天津市网络与数据安全技术重点实验室，天津 300350）

摘 要：针对基于格的单服务器口令认证密钥交换（PAKE）协议不能抵抗服务器泄露攻击，而目前基于格的多服务器 PAKE 协议的执行效率较低且不适用于两服务器场景的问题，利用带误差学习（LWE），提出了格上第一个非适应性两方平滑投影哈希函数（SPHF），具备不可区分适应性选择密文攻击（IND-CCA2）的安全性，并约束了所基于的公钥加密（PKE）方案中相关参数的大小。基于此，分别针对被动和主动敌手的攻击，提出了相应的格上可证明安全的两服务器 PAKE 协议。所提出的 2 个协议可抵御量子攻击且实现了唯口令设置，也不需要使用签名/验签、全同态加密、秘密共享等昂贵密码原语来保证安全性，被动敌手攻击下的协议还避免了零知识证明的使用。此外，在标准模型下，对所提出的 2 个协议进行了严格的安全性证明。实验结果表明，所提出的两方 SPHF 和两服务器 PAKE 协议的执行效率较高。

关键词：口令认证密钥交换协议；两服务器；平滑投影哈希函数；可证明安全；抗量子
中图分类号：TN918.1
文献标志码：A
DOI: 10.11959/j.issn.1000−436x.2022052

Provably secure quantum resistance two-server password-authenticated key exchange protocol

YIN Anqi[1], GUO Yuanbo[1], WANG Ding[2,3], QU Tongzhou[1], CHEN Lin[1]
1. Department of Cryptogram Engineering, Information Engineering University, Zhengzhou 450001, China
2. College of Cyber Science, Nankai University, Tianjin 300350, China
3. Tianjin Key Laboratory of Network and Data Security Technology, Nankai University, Tianjin 300350, China

Abstract: Aiming at the problem that the lattice-based single-sever password-authenticated key exchange (PAKE) protocols are not resistant to server compromise attack, while the existing lattice-based multi-server PAKE protocols are inefficient and incompatible with two-server scenarios. The first lattice-based two-party smooth projective hash function (SPHF) was proposed by utilizing the learning with errors (LWE), which was indistinguishability under adaptive chosen-ciphertext attack(IND-CCA2) secure. The parameters of the based public key encryption (PKE) scheme were also identified. On this basis, pertinent two-server PAKE protocols from lattices were designed countering both passive and active attackers. The two quantum resistance protocols were able to achieve password-only settings and the expensive cryptographic primitives were not used, including signature/verification, fully homomorphic encryption and secret sharing. The utilization of zero knowledge proofs were avoided by the protocol under the passive attackers. In the standard model, rigorous security proofs were provided for the two proposed protocols. Experimental results show that the proposed SPHF and PAKE protocols exhibit higher execution efficiency.

Keywords: password-authenticated key exchange protocol, two-server, smooth projective hash function, provably secure, quantum resistance

收稿日期：2021−12−16；修回日期：2022−02−15
通信作者：郭渊博，yuanbo_g@hotmail.com
基金项目：国家自然科学基金资助项目（No.61501515, No.61601515）
Foundation Item: The National Natural Science Foundation of China (No.61501515, No.61601515)

0　引言

口令认证密钥交换协议的通信方只需要使用低熵口令就可以在非安全信道上协商用于安全通信的高熵密钥[1]。与其他认证方式相比，口令认证可以避免使用公钥基础设施（PKI, public key infrastructure）和涉及个人隐私的生物特征，有效提高安全系统在部署上的低成本性和便利性[2]。

然而，当前口令泄露问题日益突出，如很多网站都发生过客户口令泄露问题，且口令泄露在短期内往往难以被发现，如 Yahoo 时隔 4 年才发现其泄露了 30 亿客户口令，这进一步加剧了口令泄露的危害。另一方面，基于传统困难问题（如大整数分解、离散对数）的密码体制无法抵抗量子算法的攻击[3]，且量子计算技术飞速发展，特别是商用量子计算机已经诞生[4]，因此基于传统困难问题的口令认证密钥交换（PAKE, password-authenticated key exchange）协议在量子时代将不再安全。

格密码体制是典型的抗量子密码体制，因其在计算上具有高效性、在困难实例选取上实现了随机性[5]且在全同态加密等领域应用前景广泛[6]，被认为是最有潜力的后量子密码体制之一。但格上 PAKE 协议的研究起步较晚且投入较少。如目前格上的 PAKE 协议多为单服务器设置，此类方案在单个服务器上存储了客户的认证信息（如口令或口令的哈希值）；一旦该服务器泄露，认证信息将全部泄露，即此类协议存在口令泄露问题。

两/多服务器认证是抵御服务器泄露攻击的有效方式，但基于格困难问题的此类研究相当有限。在基于传统困难问题的 PAKE 协议领域，相关方案可分为基于 PKI 模型[7-8]、基于身份（ID, identity）[9-10]和唯口令[11]3 种。但只有唯口令方案符合 PAKE 协议的设计初衷，即允许客户在只拥有低熵口令的前提下实现认证。其他 2 种方案分别需要客户管理系统公钥和服务器身份等信息。相比之下，目前在格上只存在个别基于 PKI 模型的多服务器口令认证密钥传输协议[6]，且不能应用于两服务器场景。虽然在格上实例化 ID2S 架构[10]是实现抗量子两服务器 PAKE 协议的可行技术路线，但只能得到基于 ID 的密钥传输方案，即上述 2 种基于格的方案虽然宣称是 PAKE 协议，但实际上都只实现了密钥传输。

此外，格上的 PAKE 协议目前还不能很好地发挥格密码体制的线性计算优势。主要原因之一是现有方案一般需要使用签名/验签[12-13]、零知识证明[14-15]等密码原语来保证安全性。而格上的多服务器 PAKE 方案[6]进一步需要使用全同态加密、秘密共享等密码原语。这些昂贵密码原语的使用降低了协议的执行效率，且增加了通信与存储开销。

据本文分析，实现格上高效的两服务器 PAKE 协议主要有三点挑战。1) 实现唯口令设置和密钥交换。一方面要使客户不需要存储和管理系统公钥、服务器身份等信息，从而提高安全系统的可部署性；另一方面要实现密钥交换而不局限于密钥传输。2) 避免使用签名/验签、全同态加密、零知识证明和秘密共享等昂贵密码原语，并降低协议通信轮次，以精简协议架构进而降低协议开销和简化协议的安全性分析。3) 设计格上 IND-CCA2 安全的两方平滑投影哈希函数（SPHF, smooth projective hash function），同时约束基于的公钥加密（PKE, public key encryption）方案中相关参数的大小，以消除带误差学习（LWE, learning with error）问题不完全同态性对 SPHF 正确性的影响。该类 SPHF 是构建唯口令两服务器 PAKE 协议的有效数学工具[11,16]，但据本文所知，目前还没有基于格的两方 SPHF，且此前即使是格上的集中式 SPHF 最多也只能保证不可区分性选择密文攻击（IND-CCA1, indistinguishability under chosen-ciphertext attack）的安全性。

因此，本文主要研究格上 IND-CCA2 安全的两方 SPHF，以解决现有分布式 SPHF 无法抵抗量子攻击和现有格上 SPHF 安全性不高的问题，并进一步提出格上可证明安全的唯口令两服务器 PAKE 协议，以抵抗服务器泄露、量子攻击，同时解决现有格上相关方案可部署性差且执行效率低的问题。实验结果表明，与其他相关方案相比，所提出的 SPHF 和 PAKE 协议具有较高的执行效率。

1　相关工作

在基于传统困难问题的 PAKE 协议领域，文献[17]首次给出了唯口令的方案，这使 PAKE 协议的研究不再局限于 PKI 模型。早期 PAKE 协议的开销一般很大，甚至不具备实用性。文献[18]在 2001 年提出了第一个高效的 PAKE 架构称为 KOY 架构。此后，为降低协议各方面的开销，协议的通信轮次不断降低，协议架构由最初的三轮（KOY[18]/GL[19]和 JG[20]/GK[21]架构）发展到两轮[22]以及一轮[23]架构。但上述方案都是单服务器方案，无法抵御服务器泄

露攻击。为此，两/多服务器 PAKE 协议应运而生。文献[9-10]虽然宣称提出了基于身份的通用两服务器 PAKE 协议架构，即可以基于任何两方 PAKE 方案实现两服务器 PAKE 协议，但实际上只能实现密钥传输。文献[11]基于秘密共享技术，提出了一种唯口令的门限 PAKE 协议。但一般情况下，多服务器 PAKE 协议不能用于两服务器场景，反之亦然。

与基于传统困难问题的 PAKE 协议相比，格上 PAKE 协议的研究起步较晚。文献[24]在 2009 年才在 KOY 架构下提出了第一个格上的 PAKE 协议。文献[15]基于非交互式零知识证明技术，在 2017 年给出了格上第一个两轮 PAKE 协议。直到 2018 年，文献[14]才提出了第一个格上的一轮 PAKE 协议，但该协议需要使用零知识证明、签名/验签以及平滑性放大等技术来保证 IND-CCA2 安全性，这导致协议的计算、存储和通信开销较大。文献[12]改进了文献[14]中的一轮 PAKE 协议，虽提高了效率却仍无法避免签名/验签算法的使用。与上述基于格的单服务器 PAKE 协议相比，格上两/多服务器 PAKE 协议的研究更加不足。文献[6]宣称提出了格上第一个多服务器 PAKE 协议，但该方案实际上只实现了密钥传输，并没有实现密钥交换；该方案基于 PKI 模型，且需要多次使用全同态加密、签名/验签、零知识证明、秘密共享等技术来保证安全性，开销较大。虽然可以在格上实例化通用的 ID2S 协议架构[9-10]，但只能得到基于 ID 的密钥传输方案，无法实现唯口令认证和密钥交换，且该架构无法避免签名/验签算法的使用，而基于格的签名/验签算法一般具有较大的算法参数，这会进一步增大协议开销。

SPHF 是构建 PAKE 协议的有效数学工具，但 SPHF 的应用不局限于该领域，其在零知识证明、不经意传输以及证据加密等领域[14]都具有相当广泛的应用。SPHF（实际上是精确 SPHF）的概念最初由文献[25]提出。由于在格上构建精确 SPHF 比较困难[26]，文献[24]首次提出了一种基于格的近似 SPHF，并利用纠错码技术，仅基于该近似 SPHF 实现了格上的第一个 PAKE 协议。随后，文献[14-15]基于上述技术路线，提出了基于不同格上 PKE 方案的近似 SPHF。文献[27]提出了一种格上 IND-CCA1 安全的公钥加密算法，一般称为 MP 公钥加密算法，由于其执行效率较高，因此成为格 PAKE 领域应用最广泛的公钥加密算法之一。文献[28]则基于 MP 公钥加密算法[28]提出了一种格上的精确 SPHF，但由于算法参数较大，计算开销成倍提高。

SPHF 可分为适应性[24]与非适应性[14]两类。在非适应性 SPHF 中，投影密钥的计算不需要使用密文，因而此类 SPHF 成为构建低轮次（特别是一轮）PAKE 协议的有效数学工具[14]。目前，格上的非适应性 SPHF 只能保证 IND-CCA1 或不可区分性选择明文攻击（IND-CPA, indistinguishability under chosen-plaintext attack）的安全性[13]。虽然个别文献[14]基于带标签的 IND-CCA2 安全的 PKE 方案设计了非适应性 SPHF，但由于要求标签固定，该 SPHF 只能保证 IND-CPA 安全。

此外，根据计算参与方的数量多少 SPHF 又可以分为集中式和分布式两类。在基于传统困难问题的 SPHF 领域，文献[11]提出了一种基于 DDH 假设的分布式 SPHF，但无法抵御量子攻击，也不适用于两方计算场景。文献[16]实际上隐式地给出了一种两方 SPHF，也无法抵抗量子攻击。据本文所知，目前格上还不存在分布式（包括两方）SPHF。

本文工作借鉴文献[29-30]的研究思路展开，主要贡献总结如下。

1) 基于 LWE 问题的加法同态属性，提出了一种格上 IND-CCA2 安全的非适应性两方 SPHF，支持一轮两方 PAKE 协议的构造。由于 LWE 问题的不完全同态性会影响 SPHF 的正确性，本文还约束了所基于的 PKE 方案中相关参数的大小。

2) 基于所提出的两方 SPHF，在被动和主动敌手的攻击下，分别提出了基于格的两服务器 PAKE 协议，以抵抗量子和服务器泄露攻击。所提出的 2 个协议都是唯口令设置，且不需要使用签名/验签、全同态加密、秘密共享等密码原语，被动敌手攻击下的协议还避免了零知识证明的使用，因而提高了安全系统的可部署性和效率。

3) 在文献[13,16]中安全模型的基础上，给出了一种更加现实的两服务器 PAKE 安全模型，该模型可以更加准确地评估 PAKE 协议面临的真实风险。在所提出的更加现实的 PAKE 安全模型下，严格证明了本文 2 个协议的安全性。证明基于标准模型，从而避免了随机预言机的使用。对于 PAKE 协议而言，使用随机预言机不仅存在实际应用中被一般哈希函数替代而导致的安全隐患，还可能造成 PAKE 协议遭受离线口令猜测攻击。

2 预备知识

在介绍本文所需的预备知识之前，首先给出本文必要的符号说明，如表 1 所示。

表 1　符号定义

符号	含义
κ	安全参数
q	LWE 问题的模数（为素数或某素数的幂）
$\leftarrow/\xleftarrow{r}$	取样/随机取样
$\lceil\cdot\rceil/\lfloor\cdot\rfloor$	向上/下取整
$\|\|\boldsymbol{x}\|\|/\|\|A\|\|$	向量 $\boldsymbol{x}$ 的模/集合 A 的大小
$\|\|$	向量或矩阵的纵向级联
$\|$	向量或矩阵的横向级联
pw	口令
$\mathbb{D}$	口令空间
$\perp$	非法标识
$\mathrm{Ham}(\cdot,\cdot)$	汉明距离函数
$\mathbf{0}$	零向量/零矩阵

2.1　基于格困难问题的公钥加密算法

格。设矩阵 $\boldsymbol{A}\in\mathbb{R}^{m\times n}$，且 $\boldsymbol{A}$ 的列向量线性无关，另设格用 Λ 表示，并定义格 Λ 为

$$\Lambda\{\Lambda_{\boldsymbol{A}}=\boldsymbol{A}\boldsymbol{t}\,|\,\boldsymbol{t}\in\mathbb{Z}_q^n\} \tag{1}$$

其中，$\boldsymbol{A}$ 为格 Λ 的基矩阵，m 为格 Λ 的维数，$\boldsymbol{t}$ 为 $\mathbb{Z}_q^n$ 上的随机向量。

判定性 LWE（$\mathrm{dLWE}_{n,m,q,\chi}$）问题。设 m，n，q，u 为正整数，其中 $m\geqslant n$，$q\geqslant 2$，$m=\mathrm{poly}(n)$，$q\leqslant 2^{\mathrm{poly}(n)}$。另设正整数 e 为 $\mathrm{dLWE}_{n,m,q,\chi}$ 问题误差，且 e 的概率分布为 χ。$\mathrm{dLWE}_{n,m,q,\chi}$ 问题定义为区分以下分布的问题：1) $A_{t,\chi}=(\boldsymbol{a},u=<\boldsymbol{t},\boldsymbol{a}>+e)$，其中 $\boldsymbol{a}\xleftarrow{r}\mathbb{Z}_q^n$，$\boldsymbol{t}\xleftarrow{r}\mathbb{Z}_q^n$，$e\leftarrow\chi$；2) $(\boldsymbol{a},u)\xleftarrow{r}\mathbb{Z}_q^n\times\mathbb{Z}_q$。

误差分布。设 $\boldsymbol{w}$ 为随机向量，并设概率密度函数 $\rho_{s,c}(\boldsymbol{w})=\exp(-\pi||\boldsymbol{w}-\boldsymbol{c}||^2/s^2)$。对于格 $\Lambda\in\mathbb{Z}^{m\times n}$，设 $\rho_{s,c}(\Lambda)=\sum_{\boldsymbol{w}\in\Lambda}\rho_{s,c}(\boldsymbol{w})$。$\Lambda$ 上的离散高斯分布为

$$D_{\Lambda,s,\boldsymbol{c}}(\boldsymbol{w})=\frac{\rho_{s,c}(\boldsymbol{w})}{\rho_{s,c}(\Lambda)} \tag{2}$$

其中，向量 $\boldsymbol{c}$ 和变量 s 分别为离散高斯分布的中心和平滑参数。进一步地，若将 $D_{\Lambda,s,\boldsymbol{c}}(\boldsymbol{w})$ 的定义域约束为 $||\boldsymbol{w}||\leqslant s\sqrt{m}$，那么 $D_{\Lambda,s,\boldsymbol{c}}(\boldsymbol{w})$ 为截断离散高斯分布。由于截断离散高斯分布和离散高斯分布在统计上相近[24]，为便于描述，下文直接省略截断二字。本文中 $\mathrm{dLWE}_{n,m,q,\chi}$ 问题的误差分布 χ 为截断离散高斯分布。

格上 IND-CCA2 安全的公钥加密方案。在格上，一般利用 IND-CPA 安全的 PKE[31]（在随机预言机模型下）或 IND-CCA1 安全的 PKE（利用哈希证明、零知识证明、陷门或 BCHK 等技术[32]）构建 IND-CCA2 安全的 PKE。因此，目前基于格上 PKE[28, 33]构建的 PAKE 协议需要使用额外的密码原语来保证 IND-CCA2 安全性。上述 BCHK 技术又分为基于签名和基于 Mac 这 2 种。文献[32]利用基于 Mac 的 BCHK 技术，提出了一种 IND-CCA2 安全的 PKE，本文将其记作 ZYF。接下来介绍该方案。

设 m，n，$\bar{m}$，k，κ 为正整数，其中 $n\,\mathrm{lb}d\geqslant 3\kappa$，$m=\bar{m}+nk$，$2\leqslant d\leqslant\sqrt{q}$，$k=\lceil\log_b q\rceil$。设 (encoded, decoded) 表示编/解码算法，$\mathbb{F}(2^\kappa)$ 表示阶为 2^κ 的有限域，$\mathrm{H}:\ \{0,1\}^*\rightarrow\mathbb{F}(2^\kappa)/\{0\}$ 表示定义域为 $\{0,1\}^*$、值域为 $\mathbb{F}(2^\kappa)/\{0\}$ 的抗碰撞哈希函数，$\mathrm{FRD}:\ \mathbb{F}(2^\kappa)\rightarrow\mathbb{Z}_q^{n\times n}$ 表示定义域为 $\mathbb{F}(2^\kappa)$、值域为 $\mathbb{Z}_q^{n\times n}$ 全秩差编码算法，KG 和 ENC 分别表示密钥生成算法和加密算法，那么 ZYF 方案的定义如下。

$\mathrm{KG}(1^\kappa)$：$\boldsymbol{A}\xleftarrow{r}\mathbb{Z}_q^{n\times\bar{m}}$，$\boldsymbol{R}\xleftarrow{r}(D_{\mathbb{Z}^{\bar{m}},\omega(\sqrt{\log n})})^{nk}$，$\boldsymbol{B}=-\boldsymbol{A}\boldsymbol{R}$，返回公私钥对 $(\mathbf{pk},\mathbf{sk})=((\boldsymbol{A},\boldsymbol{B}),\boldsymbol{R})$。

$\mathrm{ENC}(\mathbf{pk},p\in\mathbb{F}(2^\kappa))$：$\boldsymbol{s}\xleftarrow{r}D_{\mathbb{Z}^n,\alpha q}$，$\boldsymbol{e}_1\xleftarrow{r}D_{\mathbb{Z}^{\bar{m}},\alpha q}$，$\boldsymbol{e}_2\xleftarrow{r}D_{\mathbb{Z}^{nk},\gamma}$（$\gamma=\sqrt{||\boldsymbol{e}_1||^2+\bar{m}(\alpha q)^2}\,\omega(\sqrt{\log n})$）；$x,y,z\xleftarrow{r}\mathbb{F}(2^\kappa)$；令 $\boldsymbol{v}=x||y||z\in(\mathbb{F}(2^\kappa))^3$，$\mathrm{tag}=\mathrm{H}(\boldsymbol{c}_1)\in\mathbb{F}(2^\kappa)$，$\tau=\mathrm{H}(\boldsymbol{c}_2,c_3)\in\mathbb{F}(2^\kappa)$，计算

$$\begin{cases}\tilde{\boldsymbol{s}}=\boldsymbol{s}+\mathrm{encode}_d(\boldsymbol{v})\\ \boldsymbol{c}_1=\boldsymbol{A}^{\mathrm{T}}\tilde{\boldsymbol{s}}+\boldsymbol{e}_1\\ \boldsymbol{c}_2=(\boldsymbol{B}+\mathrm{FRD}(\mathrm{tag})\boldsymbol{G}_b)^{\mathrm{T}}\tilde{\boldsymbol{s}}+\boldsymbol{e}_2\\ c_3=x+p\\ c_4=\tau y+z\end{cases} \tag{3}$$

其中，$\tilde{\boldsymbol{s}}$ 为明密文对 $(\boldsymbol{C},p)$ 的证据。最后输出密文 $\boldsymbol{C}=(\boldsymbol{c}_1,\boldsymbol{c}_2,c_3,c_4)\in\mathbb{Z}_q^{\bar{m}}\times\mathbb{Z}_q^{nk}\times\mathbb{F}(2^\kappa)\times\mathbb{F}(2^\kappa)$。本文不需要使用解密算法，因此不再介绍解密过程[32]。

2.2　平滑投影哈希函数

设 pw 为口令，$\mathbb{D}$ 为口令空间，label 为有效标签，hash 为哈希函数，$\boldsymbol{C}$ 和 $\mathbb{C}_{\mathrm{pk}}$ 分别为与公钥 **pk** 对应的密文和 (label, $\boldsymbol{C}$) 对的集合。对给定的公钥 **pk**，定义集合 $\mathbb{X}$ 和 $\mathbb{L}_{\mathrm{pw}}$（$\mathrm{pw}\in\mathbb{D}$）分别为

$\mathbb{X}$:={(label, $\boldsymbol{C}$, pw) | (label, $\boldsymbol{C}$)$\in\mathbb{C}_{\mathrm{pk}}$&& pw$\in\mathbb{D}$}

$\mathbb{L}_{\mathrm{pw}}$:={(label, Enc(**pk**, hash(pw||label)), pw$\in\mathbb{D}$)}

一般称三元组 (label, $\boldsymbol{C}$, pw) 为单词，用 $\boldsymbol{W}$ 表示。根据 $\mathbb{X}$ 和 $\mathbb{L}_{\mathrm{pw}}$ 的定义可知，$\mathbb{L}=\{\mathbb{L}_{\mathrm{pw}}\}_{\mathrm{pw}\in\mathbb{D}}\subset\mathbb{X}$。

现给出近似平滑投影哈希函数的概念[24]。令 Ham 表示汉明距离函数，l 表示哈希密钥长度，参数 ε 表示错误的比例；**KH** 表示哈希密钥，$\mathbb{K}$表示 **KH** 的集合；**KP** 表示投影密钥，$\mathbb{S}$表示 **KP** 的集合；HASH 表示定义域为$\mathbb{X}$、值域为 $\mathbb{Z}_q$ 的哈希函数；$\mathcal{H}$ 表示关于 HASH 的哈希函数簇；projKG 表示投影函数，其定义域为$\mathbb{K}$，值域为$\mathbb{S}$。近似平滑投影哈希函数定义为输入公钥 **pk** 和集合$\mathbb{L}$、$\mathbb{X}$，输出 ($\mathbb{K}$, $\mathbb{G}$, $\mathcal{H}$ ={HASH($\boldsymbol{W}$, **KH**): $\mathbb{X} \to \{0,1\}^n$}, $\mathbb{S}$, projKG(**KH**): $\mathbb{K} \to \mathbb{S}$)，且满足以下条件。

1) 存在高效的取样算法 $\mathbf{KH} \xleftarrow{r} \mathbb{K}$、哈希函数 HASH($\boldsymbol{W}$, **KH**)和投影函数 projKG。

2) 正确性。对于$\forall \boldsymbol{W}$ = (label, $\boldsymbol{C}$, pw) $\in \mathbb{L}$，哈希值 $\boldsymbol{h}$ = HASH($\boldsymbol{W}$, **KH**) 可由投影密钥 **KP** 确定。即存在一个函数 projHASH，称为投影哈希函数，以投影密钥 **KP**、$\boldsymbol{W}$ 以及 $\boldsymbol{W}\in\mathbb{L}$的证据 r 为输入，投影哈希值 **ph** 为输出，且满足以下条件

$$\Pr(\mathrm{Ham}(\boldsymbol{h},\mathbf{ph}) \geqslant \varepsilon l) = \mathrm{negl}(\kappa) \tag{4}$$

3) 平滑性。对于$\forall \boldsymbol{W}$ = (label, $\boldsymbol{C}$, pw) $\in \mathbb{X}/\mathbb{L}$，$\mathbf{KH} \xleftarrow{r} \mathbb{K}$和 **KP** = projKG(**KH**)，以下 2 个分布在统计上不可区分

$$\begin{cases}(\mathbf{KP}, \mathrm{HASH}(\boldsymbol{W}, \mathbf{HK})) \\ (\mathbf{KP}, \boldsymbol{v} \leftarrow \{0,1\}^{\kappa})\end{cases} \tag{5}$$

近似 SPHF 只能实现近似正确性，文献[14]给出了通过纠错码算法实现正确性放大从而得到 SPHF 的通用技术，接下来介绍该技术。

令 $\{\mathrm{hashKG}', \mathrm{projKG}', \mathrm{HASH}', \mathrm{projHASH}'\}$ 表示近似 SPHF，其值域为 $\{0,1\}^{v}$；令 ECC 表示纠错码算法，其可以纠正 ε 比例的错误。那么以下定义的 $\{\mathrm{hashKG}, \mathrm{projKG}, \mathrm{HASH}, \mathrm{projHASH}\}$ 是 SPHF。

$\mathbf{KH} \leftarrow \mathrm{hashKG(params)}$：输入 $\mathbf{KH}_1 \leftarrow \mathrm{hashKG}'$ (params)，在 ECC 的定义域中随机选择 $\mathbf{KH}_2$，输出哈希密钥 $\mathbf{KH} := (\mathbf{KH}_1, \mathbf{KH}_2)$。

$\mathbf{KP} \leftarrow$ projKG (params, $\mathbf{KH}, \boldsymbol{W}$)：计算 $\mathbf{KP}_1 \leftarrow$ projKG (params, $\mathbf{KH}_1, \boldsymbol{W}$)，并计算 $\boldsymbol{c}$ = ECC ($\mathbf{KH}_2$)，$\boldsymbol{h}' \leftarrow$ HASH' (params, $\mathbf{KH}_1, \boldsymbol{W}$)，$\mathbf{KP}_2 = \boldsymbol{c} \oplus h'$，最后输出 $\mathbf{KP} := (\mathbf{KP}_1, \mathbf{KP}_2)$。

$\boldsymbol{h} \leftarrow \mathrm{HASH}(\mathrm{params}, \mathbf{KH}, \boldsymbol{W})$：输出 $\boldsymbol{h}$ = $\mathbf{KH}_2$。

$\mathbf{ph} \leftarrow \mathrm{projHASH}(\mathrm{params}, \mathbf{KP}, \boldsymbol{W}, r)$：计算 $\mathbf{ph}' \leftarrow$ projHASH(params, $\mathbf{KP}_1, \boldsymbol{W}, r$)，并输出 $\mathbf{ph}$ = ECC^{-1} ($\mathbf{ph}' \oplus \mathbf{KP}_2$)。

由于上述技术可以直接将近似 SPHF 转换为 SPHF，为方便描述下文直接称近似平滑投影哈希函数为平滑投影哈希函数。

3 两服务器 PAKE 安全模型

本节基于文献[13,16,34]中的模型，给出了一种更加现实的两服务器 PAKE 安全模型，该模型能更加准确地评估 PAKE 协议面临的真实风险。

3.1 通信模型

协议的参与方包括客户 $C \in$ Client，服务器 $A, B \in$ Sever 和敌手/攻击者 $\mathcal{A} \in$ Adv。客户持有口令 pw_C，服务器 A 和 B 分别存储口令份额 $\mathrm{pw}_{A,C}$ 和 $\mathrm{pw}_{B,C}$，且满足 $\mathrm{pw}_{A,C} + \mathrm{pw}_{B,C} (\mathrm{mod} q) = \mathrm{pw}_C$。一个服务器可以处理多个客户的认证请求，本文仅以一个客户为例，研究两服务器 PAKE 协议。

在协议执行过程中存在“客户−服务器”之间和“服务器−服务器”之间的通信。本文假设敌手完全控制“客户−服务器”之间的通信，因此敌手可以窃听、注入、篡改、延迟或拦截该通信网络上的任何消息。服务器的存储空间相对充足，可以存储长期密钥等其他信息。因此，本文假设“服务器−服务器”之间的通信是安全的，即敌手无法窃听或控制未泄露服务器之间的通信。实际上可通过设置可信服务器来实现秘密信息的分发。此外，假设服务器与客户之间的通信是异步的；并假设服务器之间的通信是半同步的，即若客户向服务器端发送消息，各服务器可以在规定的时间间隔内收到该消息。

文献[34]提出的 BPR 模型是应用最广泛的 PAKE 安全模型之一。在 BPR 模型中，每个参与方可以与其他参与方并行地执行多个协议。该模型用实例 Π 建模协议的执行，如 Π_C^i 表示客户 C 的第 i 个实例，$\Pi_A^{j_A}$ 表示服务器 A 的第 j_A 个实例，且一个实例只能使用一次。每个实例，如 Π_C^i，拥有一个状态变量 $(\mathrm{sid}_C^i, \mathrm{pid}_C^i, \mathrm{acc}_C^i, \mathrm{term}_C^i, \mathrm{sk}_C^i)$。其中，$\mathrm{sid}_C^i$ 表示实例的会话标识，它为 Π_C^i 接收和发送的消息统一编号，服务器之间的传输消息不计入在内；pid_C^i 表示客户 C 认为的其通信伙伴的标识，在两服务器 PAKE 协议中，对于客户 C 而言，pid_C^i 是一个集合，即 $\mathrm{pid}_C^i = \{A, B \mid A, B \in \mathrm{Sever}\}$；$\mathrm{sk}_C^i$ 表示会话密钥，同理 $\mathrm{sk}_C^i = \{\mathrm{sk}_{C,A}^i, \mathrm{sk}_{C,B}^i\}$；$\mathrm{term}_C^i$ 表示标识 Π_C^i 是否终止的二值变量，若 Π_C^i 终止，则 $\mathrm{term}_C^i = 1$，这表示 Π_C^i

完成了消息的发送和接收；acc_C^i 表示标识 Π_C^i 是否被接收的二值变量，若实例 Π_C^i 最终被接收，那么 $\mathrm{acc}_C^i=1$，这表示 Π_C^i 成功终止。

伙伴关系。若以下条件成立：1) $\mathrm{sid}_C^i=\mathrm{sid}_A^{j_A}\neq\perp$；2) $\mathrm{pid}_C^i=\{A,*\}$，$\mathrm{pid}_A^{j_A}=C$，则称实例 Π_C^i 和实例 $\Pi_A^{j_A}$ 互为伙伴关系。此时，若存在某服务器 B，满足 $\mathrm{pid}_C^i=\{A,B\}$，则称服务器 B 是服务器 A 的相关服务器，也称 $\Pi_A^{j_A}$ 与 $\Pi_B^{j_B}$ 是相关的。

3.2　敌手模型

本文假设敌手最多入侵某客户 2 个认证服务器中的一个，并称被入侵的服务器为泄露服务器。本文将敌手分为被动和主动 2 种类型。在被动敌手攻击下，泄露服务器仍按协议的定义执行，但敌手可以获得该服务器的内部状态信息。而主动敌手可以进一步控制泄露服务器向客户和相关服务器发送的消息。注意，在本文的通信模型下，被动敌手也可以控制服务器向客户发送的消息，但 2 种敌手都无法窃听或控制 2 个未泄露服务器之间的通信。

BPR 模型[34]用预言机建模敌手能力，敌手可以通过访问不同的预言机来执行不同的攻击。在 BPR 模型中，敌手与客户或服务器（实际上是客户或服务器的实例）之间的交互是通过访问预言机实现的。根据 BPR 和文献[16]中的安全模型，现将两服务器 PAKE 协议中的敌手能力建模为以下预言机。

Execute(C, i, A, B, j_A, j_B)。该预言机建模离线口令猜测攻击，并根据协议的定义向敌手返回实例 Π_C^i、$\Pi_A^{j_A}$、$\Pi_B^{j_B}$ 之间的传输信息。

以下 Send 预言机建模在线口令猜测攻击，敌手可以向客户或服务器发送 Send 预言机询问。

Send($C, i, (A, B)$ msg)。该预言机向实例 Π_C^i 发送消息 msg，并根据协议的定义向敌手返回传输消息。特别地，Send($C, i, (A, B)$)可以启动客户 C 与服务器(A, B)之间的一次协议执行。

Send(A, j_A, B, C, msg)。该预言机向实例 $\Pi_A^{j_A}$ 发送消息 msg，并根据协议的定义向敌手返回传输消息。特别地，若服务器 A（或 B）泄露，敌手可以额外获得 A（或 B）的内部状态信息。

Reveal(C, A, i)。该预言机（其中 $A\in\mathrm{pid}_C$）建模会话密钥泄露攻击，如会话密钥的非法擦除等，并向敌手返回会话密钥 $\mathrm{sk}_{C,A}^i$。

在介绍 Test 预言机前，首先给出新鲜性的概念。

新鲜性。设实例 Π_C^i 与 $\Pi_A^{j_A}$ 互为伙伴关系，且 $\mathrm{pid}_C^i=\{A,B\mid A,B\in\mathrm{Sever}\}$，若敌手 1) 未访问过 Reveal($C, A, i$) 预言机，2) 未访问过 Reveal($A, C, j_A$) 预言机，3) 服务器 A 与服务器 B 都未泄露，那么称会话密钥 $\mathrm{sk}_{C,A}^i$ 是新鲜的。

Test(C, A, i)。该预言机（其中 $A\in\mathrm{pid}_C$）不建模任何真实的敌手能力，只用于定义安全协议。该预言机选择一个随机比特 b，若 $b=1$，其向敌手返回真实会话密钥 $\mathrm{sk}_{C,A}^i$；否则，其向敌手返回与 $\mathrm{sk}_{C,A}^i$ 等长的随机串。敌手只能执行一次 Test(C, A, i) 询问，且只能执行关于新鲜会话密钥的 Test 询问。

在本文中，敌手仿冒某泄露服务器向相关服务器发送消息的行为也属于在线口令猜测攻击。

正确性。设实例 Π_C^i 与实例 $\Pi_A^{j_A}$、$\Pi_B^{j_B}$ 都互为伙伴关系，且 $\mathrm{pid}_C^i=\{A,B\}$，如果 1) $\mathrm{acc}_C^i=\mathrm{acc}_A^{j_A}=\mathrm{acc}_B^{j_B}=1$，2) $\mathrm{sk}_{C,A}^i=\mathrm{sk}_{A,C}^{j_A}$，3) $\mathrm{sk}_{C,B}^i=\mathrm{sk}_{B,C}^{j_B}$，则称两服务器 PAKE 协议是正确的。

在主动敌手攻击下（不妨设服务器 A 泄露），任何协议无法保证正确性条件 2) 一定成立，但本文在第 5.2 节提出的协议满足若 $\mathrm{acc}_C^i=\mathrm{acc}_B^{j_B}=1$，那么正确性条件 3) 一定成立。

敌手优势。口令认证密钥交换协议应该具备语义安全性，即敌手无法区分真实的会话密钥及与之等长的随机串。设敌手进行 Test 询问后，给出的猜测比特为 b'，若 $b'=b$，称敌手成功，并记该事件为 Success。定义敌手优势 Adv($\mathcal{A}$)为

$$\mathrm{Adv}(\mathcal{A})=2\Pr(\mathrm{Success})-1 \tag{6}$$

理想情况下，敌手成功的概率 Pr(Success)的上限为 $\frac{1}{2}+\mathrm{negl}(\kappa)$（$\mathrm{negl}(\kappa)$ 为可忽略函数）。此时敌手优势 Adv($\mathcal{A}$) 的上限为 $\mathrm{negl}(\kappa)$。但由于客户实际使用的口令空间较小，敌手总能通过在线口令猜测成功，敌手优势的上限必定大于 $\mathrm{negl}(\kappa)$。因此，如果敌手所能执行的最佳攻击方式为在线口令猜测攻击，则称 PAKE 协议是安全的。

设概率多项式时间（PPT, probabilistic polynomial time）的敌手执行在线攻击有次数上限，为 $Q(\kappa)$。文献[11]在定义安全 PAKE 协议时，一般假设口令服从均匀分布，此时敌手优势的上限为 $\frac{Q(\kappa)}{\|D\|}+\mathrm{negl}(\kappa)$。但文献[35]的研究表明，口令服从均匀分

布大大低估了敌手在线口令猜测的成功率，因而基于口令均匀分布假设的安全模型也大大低估了 PAKE 协议面临的真实风险。因此，本节假设口令分布服从 CDF-Zipf 定律[35]，并重新定义了文献[11]中的安全两服务器 PAKE 协议。

定义 1 安全两服务器 PAKE 协议。设 PPT 敌手$\mathcal{A}$执行在线口令猜测攻击的次数有上限，为$Q(\kappa)$；并设口令空间为$\mathbb{D}$，且口令分布服从 CDF-Zipf 定律。假设敌手最多入侵某客户 2 个认证服务器中的一个，如果对于所有的 PPT 敌手$\mathcal{A}$，存在可忽略函数 $\mathrm{negl}(\kappa)$，满足

$$\mathrm{Adv}_{\mathcal{A},\Pi}(\kappa)\leqslant C'Q(\kappa)^{s'}+\mathrm{negl}(\kappa) \tag{7}$$

称协议 Π 是安全的两服务器 PAKE 协议，其中，$C'=0.062\,239$ 和 $s'=0.155\,478$ 是 CDF-Zipf 分布的拟合参数[13]，κ是安全参数。

根据文献[36]，基于 CDF-Zipf 口令分布模型的敌手优势与真实敌手优势最多相差 0.491%，且与基于口令均匀分布模型进行口令猜测相比，基于 CDF-Zipf 口令分布模型进行口令猜测的成功率可提高 2～3 倍。因此，所提出的两服务器 PAKE 安全模型更加现实，其精确度可比基于口令均匀分布假设的安全模型提高 2～3 倍。

4 格上的 IND-CCA2 安全的两方 SPHF

两方 SPHF 是构建唯口令两服务器 PAKE 协议的有效数学工具。但据本文所知在格上还不存在此类 SPHF。在基于传统困难问题的 SPHF 领域，可以通过集中式 SPHF 构建两方 SPHF，但这在格上比较困难，原因有三。一是目前格上的 SPHF 多是近似 SPHF，利用此类 SPHF 设计的两方 SPHF 不能满足正确性要求。二是虽然存在个别精确 SPHF[27]，但一方面该 SPHF 基于强 LWE 问题，因而算法参数较大，执行效率成倍降低；另一方面，鉴于强 LWE 问题也只具备不完全加法同态属性，直接利用该 SPHF 构建的两方 SPHF 仍不能满足正确性要求。三是现有的格上集中式 SPHF 只具备 IND-CPA 或 IND-CCA1 安全性，基于这两类 SPHF 设计的两方 SPHF 不能保证 IND-CCA2 安全性。此外，为优化协议的通信轮次，本节研究非适应性 SPHF。

本节利用 LWE 问题的加法同态属性，基于 ZYF 方案（见第 2.1 节）研究格上 IND-CCA2 安全的非适应性两方 SPHF。

设ZYF方案的公钥为$\mathbf{pk}_Z$，公共本原矩阵为$\boldsymbol{G}_b$，哈希密钥长度为κ。记本文提出的格上非适应性两方 SPHF 为 WI-2SPHF，包括哈希密钥生成函数 hashKG、投影函数 proj、哈希函数 HASH 和投影哈希函数 projHASH 这 4 个子函数。WI-2SPHF (hashKG, proj, HASH, projHASH) 的定义如下。

$\mathrm{hashKG}(q,\bar{m},\kappa)$。首先输入算法参数$(q,\bar{m},\kappa)$，然后对于$1\leqslant i\leqslant\kappa$，根据算法参数取样$\mathbf{kh}_i\xleftarrow{\mathrm{r}}\mathbb{Z}_q^{\bar{m}}$，最后输出哈希密钥$\mathbf{KH}:=(\mathbf{kh}_1,\cdots,\mathbf{kh}_\kappa)\in(\mathbb{Z}_q^{\bar{m}})^\kappa$。

$\mathrm{proj}(\mathbf{KH}=(\mathbf{kh}_1,\cdots,\mathbf{kh}_\kappa),\mathbf{pk}_Z=(\boldsymbol{A},\boldsymbol{B}),\boldsymbol{G}_b)$。输入哈希密钥$\mathbf{KH}=(\mathbf{kh}_1,\cdots,\mathbf{kh}_\kappa)$、$\mathbf{pk}_Z=(\boldsymbol{A},\boldsymbol{B})$和$\boldsymbol{G}_b$；计算$\mathbf{kp}_i=(\boldsymbol{A}+(\boldsymbol{B}+\mathrm{F}(\mathrm{tag})\boldsymbol{G}_b)|\mathbf{0})\mathbf{kh}_i$（$1\leqslant i\leqslant\kappa$）；输出投影密钥$\mathbf{KP}:=(\mathbf{kp}_1,\cdots,\mathbf{kp}_\kappa)\in(\mathbb{Z}_q^n)^\kappa$。

$\mathrm{HASH}(\boldsymbol{C}=(\boldsymbol{c}_1,\boldsymbol{c}_2,\boldsymbol{c}_3,\boldsymbol{c}_4),\mathbf{KH}:=(\mathbf{kh}_1,\cdots,\mathbf{kh}_\kappa))$。首先输入密文$\boldsymbol{C}=(\boldsymbol{c}_1,\boldsymbol{c}_2,\boldsymbol{c}_3,\boldsymbol{c}_4)$和哈希密钥$\mathbf{KH}:=(\mathbf{kh}_1,\cdots,\mathbf{kh}_\kappa)$，然后计算

$$\begin{cases}h_i^1=\left(\boldsymbol{c}_1+\boldsymbol{c}_2\|\mathbf{0}\right)^{\mathrm{T}}\mathbf{hk}_i(\mathrm{mod}\,q)\\ h_i^2=(h_i^1+c_3+c_4)(\mathrm{mod}\,q)-\dfrac{q}{2}\\ h_i=1+\mathrm{RD}\left(\dfrac{2h_i^2}{q}\right)(\mathrm{mod}\,2)\end{cases} \tag{8}$$

其中，$\mathrm{RD}(x)$表示对x进行四舍五入操作。最后输出哈希值$h:=(h_1,\cdots,h_\kappa)$。

$\mathrm{projHASH}(\mathbf{KP}=(\mathbf{kp}_1,\cdots,\mathbf{kp}_\kappa),\boldsymbol{C}=(\boldsymbol{c}_1,\boldsymbol{c}_2,\boldsymbol{c}_3,\boldsymbol{c}_4),\tilde{\boldsymbol{s}})$。首先输入$\mathbf{KP}=(\mathbf{kp}_1,\cdots,\mathbf{kp}_\kappa)$、$\boldsymbol{C}=(\boldsymbol{c}_1,\boldsymbol{c}_2,\boldsymbol{c}_3,\boldsymbol{c}_4)$和$\boldsymbol{W}=(\mathrm{label},\boldsymbol{C},\mathrm{pw})\in\mathbb{L}$的证据$\tilde{\boldsymbol{s}}$，然后计算

$$\begin{cases}\mathrm{ph}_i^1=\mathbf{kp}_i^{\mathrm{T}}\tilde{s}(\mathrm{mod}\,q)\\ \mathrm{ph}_i^2=(\mathrm{ph}_i^1+c_3+c_4)(\mathrm{mod}\,q)-\dfrac{q}{2}\\ \mathrm{ph}_i=1+\mathrm{RD}\left(\dfrac{2\mathrm{ph}_i^2}{q}\right)(\mathrm{mod}\,2)\end{cases} \tag{9}$$

最后输出投影哈希值$\mathbf{ph}:=(\mathrm{ph}_1,\cdots,\mathrm{ph}_\kappa)$。

服务器 A 输入$(q,\bar{m},\kappa)$、$\mathbf{pk}_Z$和$\boldsymbol{G}_b$并计算

$$\begin{cases}\mathbf{KH}_A\leftarrow\mathrm{hashKG}(q,\bar{m},\kappa)\\ \mathbf{KP}_A\leftarrow\mathrm{proj}(\mathbf{KH}_A,\mathbf{pk}_Z=(\boldsymbol{A},\boldsymbol{B}),\boldsymbol{G}_b)\\ \boldsymbol{h}_A\leftarrow\mathrm{HASH}(\boldsymbol{C}=(\boldsymbol{c}_1,\boldsymbol{c}_2,\boldsymbol{c}_3,\boldsymbol{c}_4),\mathbf{KP}_A)\\ \mathbf{ph}_A\leftarrow\mathrm{projHASH}(\mathbf{KP}_A,\boldsymbol{C}=(\boldsymbol{c}_1,\boldsymbol{c}_2,\boldsymbol{c}_3,\boldsymbol{c}_4),\tilde{\boldsymbol{s}}_A)\end{cases} \tag{10}$$

服务器 B 输入$(q,\bar{m},\kappa)$、$\mathbf{pk}_Z$和$\boldsymbol{G}_b$并计算

$$\begin{cases}\mathbf{KH}_B \leftarrow \text{hashKG}(q,\bar{m},\kappa)\\ \mathbf{KP}_B \leftarrow \text{proj}(\mathbf{KH}_B,\mathbf{pk}_Z=(\boldsymbol{A},\boldsymbol{B}),\boldsymbol{G}_b)\\ \boldsymbol{h}_B \leftarrow \text{HASH}(\boldsymbol{C}=(\boldsymbol{c}_1,\boldsymbol{c}_2,\boldsymbol{c}_3,\boldsymbol{c}_4),\mathbf{KP}_B)\\ \mathbf{ph}_B \leftarrow \text{projHASH}(\text{KP}_B,\boldsymbol{C}=(\boldsymbol{c}_1,\boldsymbol{c}_2,\boldsymbol{c}_3,\boldsymbol{c}_4),\tilde{\boldsymbol{s}}_B)\end{cases} \tag{11}$$

最后计算 $\boldsymbol{h}_S=\boldsymbol{h}_A\oplus\boldsymbol{h}_B$， $\mathbf{ph}_S=\mathbf{ph}_A\oplus\mathbf{ph}_B$。

为确保 WI-2SPHF 的正确性，现确定相关参数。根据式(8)和式(9)可知，$|h_i^1-\text{ph}_i^1|=|(\boldsymbol{e}_1+\boldsymbol{e}_2)^{\text{T}}\text{hk}_i|$；又 $\boldsymbol{h}_S=\boldsymbol{h}_A\oplus\boldsymbol{h}_B$，$\mathbf{ph}_S=\mathbf{ph}_A\oplus\mathbf{ph}_B$，进一步，根据式(10)和式(11)可知，要确保 WI-2SPHF 的正确性，误差分布应该满足。

$$2|(\boldsymbol{e}_1+\boldsymbol{e}_2)^{\text{T}}\text{hk}_i|\leqslant\frac{\varepsilon q}{8} \tag{12}$$

又 $\boldsymbol{e}_1$ 和 $\boldsymbol{e}_2$ 取自截断离散高斯分布，所以 $\boldsymbol{e}_1\leqslant\alpha q\sqrt{\bar{m}\kappa}$， $\boldsymbol{e}_2\leqslant\gamma\sqrt{nk\kappa}$，那么根据式(12)可得

$$\left(\alpha q\sqrt{\bar{m}\kappa}+\gamma\sqrt{nk\kappa}\right)s\sqrt{\bar{m}}\leqslant\frac{\varepsilon q}{16} \tag{13}$$

由第 2.2 节中 SPHF 的定义可知，SPHF 要具备：1) 正确性，即哈希值可通过哈希密钥或者投影密钥 2 种途径计算，这保证协议双方可以协商一致的会话密钥；2) 平滑性，当单词 $\boldsymbol{W}\in\mathbb{X}/\mathbb{L}$ 时，即使公开投影密钥，敌手也不能学习到哈希值的任何信息，这保证了 SPHF 的安全性。下面证明本文提出的 WI-2SPHF 是平滑投影哈希函数。

定理 1　WI-2SPHF 是平滑投影哈希函数。

证明　详见附录 1。

5　格上可证明安全的两服务器 PAKE 协议

在格上，PAKE 协议目前多为单服务器设置，无法抵抗服务器泄露攻击。个别可抵抗该类攻击的方案也只能实现密钥传输，且需要基于 PKI 模型[6]，因而客户需要存储和管理系统公钥。虽然可以在格上实例化文献[9-10]中通用的 ID2S 架构，但只能得到基于身份的两服务器密钥传输方案，导致客户需要额外存储和管理服务器身份等信息。因此，本节研究格上唯口令的两服务器 PAKE 协议，以提高安全系统的可部署性。此外，为应对不同的性能和安全性需求，本节分别研究被动和主动敌手攻击下可证明安全的两服务器 PAKE 协议。

5.1　针对被动敌手攻击的两服务器 PAKE 协议

假设协议在客户 C 与服务器 A、B 之间执行，C 持有口令 pw_C，A 和 B 分别存储口令份额 $\text{pw}_{A,C}$ 和 $\text{pw}_{B,C}$，满足 $\text{pw}_{A,C}+\text{pw}_{B,C}(\text{mod}\,q)=\text{pw}_C$。

基于所提出的 WI-2SPHF，本节针对被动敌手攻击，提出了一种格上可证明安全的唯口令两服务器 PAKE 协议，记作 2S-PAKE，如图 1 所示。

协议所需的密码原语、初始化阶段及执行过程如下。

密码原语。1) 格上 IND-CCA2 安全的 ZYF 方案；2) 基于格的两方 SPHF、WI-2SPHF。

初始化阶段。协议参与方（包括客户 C 与服务器 A、B）在初始化阶段建立公共参考序列 $\text{CRS}=((\boldsymbol{A},\boldsymbol{B}),\boldsymbol{G}_b)$，其中 $\mathbf{pk}_Z=(\boldsymbol{A},\boldsymbol{B})$ 和 $\boldsymbol{G}_b$ 分别是 ZYF 方案的公钥和公共本原矩阵。

执行过程。客户 C 根据 ZYF 方案的定义选择 $x_{C,A},y_{C,A},z_{C,A},x_{C,B},y_{C,B},z_{C,B}\xleftarrow{\text{r}}\mathbb{F}(2^\kappa)$， $\boldsymbol{s}_{C,A},\boldsymbol{s}_{C,B}\xleftarrow{\text{r}}D_{\mathbb{Z}^n,\alpha q}$， $\boldsymbol{e}_{1A},\boldsymbol{e}_{1B}\xleftarrow{\text{r}}D_{\mathbb{Z}^{\bar{m}},\alpha q}$， $\boldsymbol{e}_{2A},\boldsymbol{e}_{2B}\xleftarrow{\text{r}}D_{\mathbb{Z}^{nk},\gamma}$，并计算 $\tilde{\boldsymbol{s}}_{C,A}$ 和 $\tilde{\boldsymbol{s}}_{C,B}$；为 A 和 B 计算哈希密钥 $\mathbf{KH}_{C,A}\leftarrow\text{hashKG}(q,\bar{m},\kappa)$、$\mathbf{KH}_{C,B}\leftarrow\text{hashKG}\ (q,\bar{m},\kappa)$，并根据ZYF方案的定义，为服务器$A$和$B$分别计算密文 $\boldsymbol{C}_{C,A}=\text{ZYF}\ (x_{C,A},\ y_{C,A},\ z_{C,A},\ \tilde{\boldsymbol{s}}_{C,A},\ \boldsymbol{e}_{1C,A},\boldsymbol{e}_{2C,A},\text{pw}_C)$ 和 $\boldsymbol{C}_{C,B}=\text{ZYF}\ (x_{C,B},y_{C,B},z_{C,B},\ \tilde{\boldsymbol{s}}_{C,B},\ \boldsymbol{e}_{1C,B},\boldsymbol{e}_{2C,B},\text{pw}_C)$；然后计算投影密钥 $\mathbf{KP}_{C,A}\leftarrow\text{proj}(\mathbf{KH}_{C,A},\mathbf{pk}_Z,\boldsymbol{G}_b)$、$\mathbf{KP}_{C,B}\leftarrow\text{proj}\ (\mathbf{KH}_{C,B},\ \mathbf{pk}_Z,\boldsymbol{G}_b)$；最后，$C$ 向 A 和 B 发送消息 $(C,\boldsymbol{C}_{C,A},\boldsymbol{C}_{C,B},\mathbf{KP}_{C,A},\mathbf{KP}_{C,B})$。

服务器 A 与服务器 B 的操作对称，下文仅以服务器 A 为例说明服务器端的操作。

服务器 A 根据 ZYF 方案的定义选择 $x_A,y_C,z_C,x_A',y_C',z_C'\xleftarrow{\text{r}}\mathbb{F}(2^\kappa)$，$\boldsymbol{s}_A,\boldsymbol{s}_A'\xleftarrow{\text{r}}D_{\mathbb{Z}^n,\alpha q}$，$\boldsymbol{e}_{1A},\boldsymbol{e}_{1A}'\xleftarrow{\text{r}}D_{\mathbb{Z}^{\bar{m}},\alpha q}$， $\boldsymbol{e}_{2A},\boldsymbol{e}_{2A}'\xleftarrow{\text{r}}D_{\mathbb{Z}^{nk},\gamma}$，并计算 $\tilde{\boldsymbol{s}}_A$ 和 $\tilde{\boldsymbol{s}}_A'$；计算哈希密钥 $\mathbf{KH}_A\leftarrow\text{hashKG}(q,\bar{m},\kappa)$ 和 $\mathbf{KH}_A'\leftarrow\text{hashKG}(q,\bar{m},\kappa)$；根据 ZYF 方案分别计算密文 $\boldsymbol{C}_A=\text{ZYF}(x_A,y_A,z_A,\tilde{\boldsymbol{s}}_A,\boldsymbol{e}_{1A},\boldsymbol{e}_{2A},\text{pw}_{C,A})$ 和 $\boldsymbol{C}_A'=\text{ZYF}\ (x_A',\ y_A',z_A',\tilde{\boldsymbol{s}}_A',\boldsymbol{e}_{1A}',\boldsymbol{e}_{2A}',\text{pw}_{C,A})$；然后为其本身和 B 计算投影密钥 $\mathbf{KP}_A\leftarrow\text{proj}(\mathbf{KH}_A,\mathbf{pk}_Z,\boldsymbol{G}_b)$ 和 $\mathbf{KP}_A'\leftarrow\text{proj}(\mathbf{KH}_A',\ \mathbf{pk}_Z,\boldsymbol{G}_b)$；最后，服务器 A 向客户 C 发送消息 $(A,\boldsymbol{C}_A,\boldsymbol{C}_A',\mathbf{KP}_A,\mathbf{KP}_A')$。

客户 C 在收到消息 $(A,\boldsymbol{C}_A,\boldsymbol{C}_A',\mathbf{KP}_A,\mathbf{KP}_A')$ 和 $(B,\boldsymbol{C}_B,\boldsymbol{C}_B',\mathbf{KP}_B,\mathbf{KP}_B')$ 后，为 A 计算哈希值 $h_{C,A}\leftarrow\text{HASH}(\boldsymbol{C}_A,\mathbf{KH}_{C,A})\oplus\text{HASH}(\boldsymbol{C}_B',\mathbf{KH}_{C,B})$，为 B 计算 $\boldsymbol{h}_{C,B}\leftarrow\text{HASH}(\boldsymbol{C}_B,\mathbf{KH}_{C,B})\oplus\text{HASH}(\boldsymbol{C}_A',\mathbf{KH}_{C,A})$；并为 A 计算投影哈希值 $\mathbf{ph}_{C,A}\leftarrow\text{projHASH}(\mathbf{KP}_A,\boldsymbol{C}_{C,A},\tilde{\boldsymbol{s}}_{C,A})\oplus\text{projHASH}(\mathbf{KP}_B',\boldsymbol{C}_{C,A},\tilde{\boldsymbol{s}}_{C,A})$，为服务器 B 计算

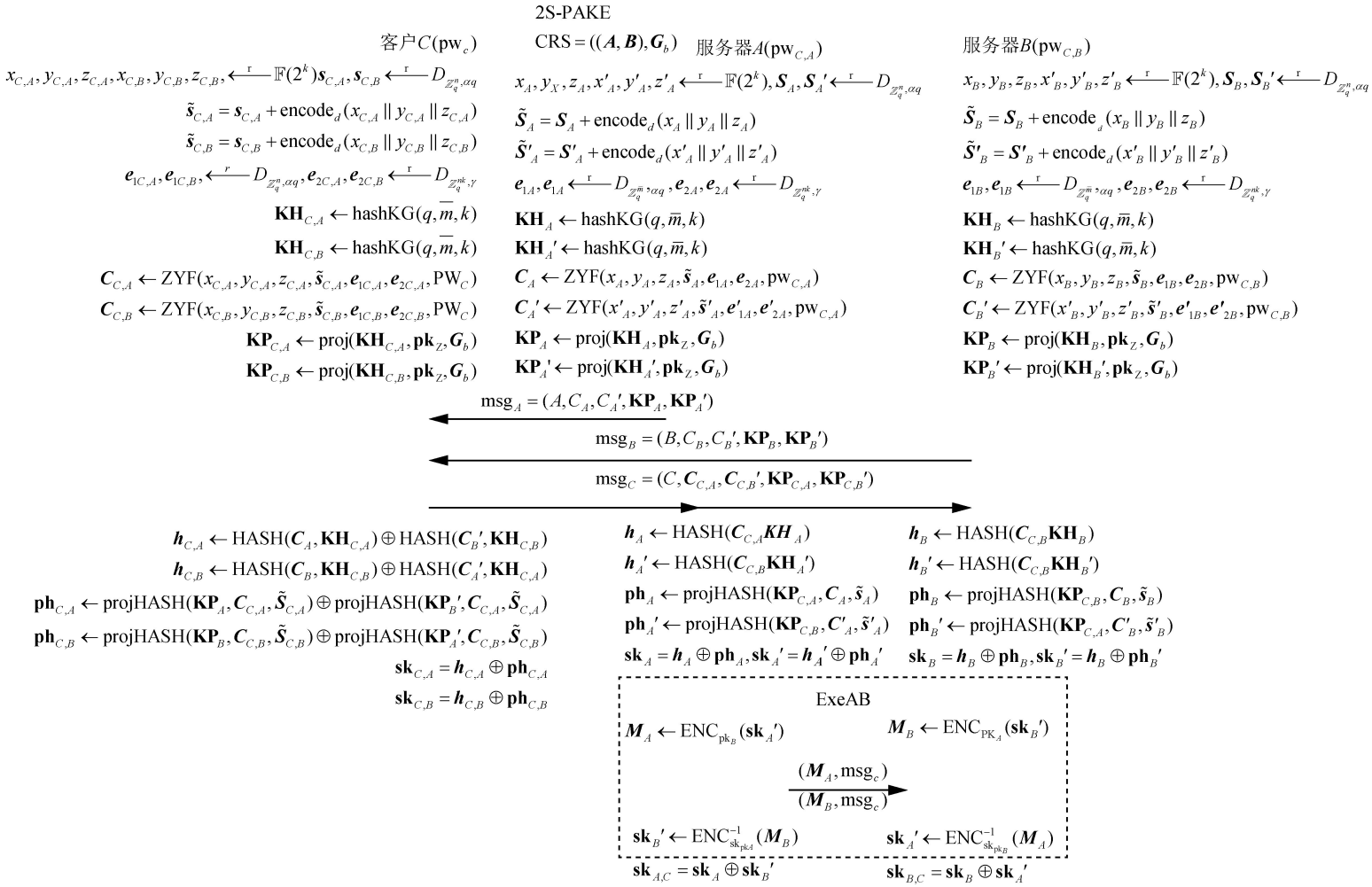

图 1　格上可证明安全的唯口令两服务器 PAKE 协议

$\mathbf{ph}_{C,B} \leftarrow$ projHASH $(\mathbf{KP}_B, \boldsymbol{C}_{C,B}, \tilde{\boldsymbol{s}}_{C,B}) \oplus$ proj $(\mathbf{KP}_A', \boldsymbol{C}_{C,B}, \tilde{\boldsymbol{s}}_{C,B})$，其中 $\tilde{\boldsymbol{s}}_{C,A}$ 和 $\tilde{\boldsymbol{s}}_{C,B}$ 分别是 $\boldsymbol{W}_{C,A} \in \mathbb{L}$ 和 $\boldsymbol{W}_{C,B} \in \mathbb{L}$ 的证据；最后，客户 C 分别计算与服务器 A 和服务器 B 之间的会话密钥 $\mathbf{sk}_{C,A} = \boldsymbol{h}_{C,A} \oplus \mathbf{ph}_{C,A}$ 和 $\mathbf{sk}_{C,B} = \boldsymbol{h}_{C,B} \oplus \mathbf{ph}_{C,B}$。

服务器 A 在收到客户 C 发送的消息 $(C, \boldsymbol{C}_{C,A}, \boldsymbol{C}_{C,B}, \mathbf{KP}_{C,A}, \mathbf{KP}_{C,B})$ 后，计算哈希值 $h_A \leftarrow \mathrm{HASH}(\boldsymbol{C}_{C,A}, \mathbf{KH}_A)$、$h_A' \leftarrow \mathrm{HASH}(\boldsymbol{C}_{C,B}, \mathbf{KH}_A')$ 和投影哈希值 $\mathbf{ph}_A \leftarrow \mathrm{projHASH}(\mathbf{KP}_{C,A}, \boldsymbol{C}_A, \tilde{\boldsymbol{s}}_A)$、$\mathbf{ph}_A' \leftarrow \mathrm{projHASH}(\mathbf{KP}_{C,B}, \boldsymbol{C}_A', \tilde{\boldsymbol{s}}_A')$；然后计算 $\mathbf{sk}_A = \boldsymbol{h}_A \oplus \mathbf{ph}_A$ 和 $\mathbf{sk}_A' = \boldsymbol{h}_A' \oplus \mathbf{ph}_A'$，并与服务器 B 执行 ExeAB 协议以获得 $\mathbf{sk}_B'$（在 ExeAB 协议中，A 用 B 的公钥加密 $\mathbf{sk}_A'$ 以得到 $\boldsymbol{M}_A$，并将 $(\boldsymbol{M}_{\mathrm{A}}, \mathrm{msg}_C)$ 发送给 B；A 收到 B 发送的 $(\boldsymbol{M}_B, \mathrm{msg}_C)$ 后，验证 B 发送的 msg_C 是否与 C 发送的一致，若一致，则用自己的私钥解密得到 $\mathbf{sk}_B'$；否则，结束协议的执行。）。最后计算会话密钥 $\mathbf{sk}_{A,C} = \mathbf{sk}_A \oplus \mathbf{sk}_B'$。

本文通过证明定理 2，证明所提出的 2S-PAKE 协议在第 3 节中更现实的安全模型下是被动敌手攻击下安全的两服务器 PAKE 协议。

定理 2　若 ZYF 方案是 IND-CCA2 安全的公钥加密方案，且 WI-2SPHF 是平滑投影哈希函数，那么图 1 所示的 2S-PAKE 协议在被动敌手攻击下是安全的两服务器 PAKE 协议。

证明　协议的正确性与安全性证明见附录 2。

5.2　针对主动敌手攻击的两服务器 PAKE 协议

假设协议仍在客户 C 与服务器 A、B 之间执行。与针对被动敌手攻击的协议相比，针对主动敌手攻击的协议要求服务器 A 和 B 在运行 ExeAB 协议时，向对方证明其本身正确使用了口令份额，本节使用证据不可区分协议实现此目标。

本节利用文献[37]中的证据不可区分协议 Σ，改进图 1 中的 ExeAB 协议，从而得到主动敌手攻击下安全的两服务器 PAKE 协议。由于服务器 A 与服务器 B 的操作具有对称性，下文仅以服务器 A 为例说明 ExeAB 协议的变化。

服务器 A 运行 Σ 协议，各参数设置如式(14)所示。服务器 A 同时作为验证者，根据协议 Σ 对服务器 B 进行验证，若验证失败，则结束协议的执行。

$$\begin{cases} \boldsymbol{Y}_1, \boldsymbol{Y}_2 \leftarrow D_\sigma \\ \boldsymbol{T}_1 = \mathbf{KP}_{C,B}{}^{\mathrm{T}} \tilde{\boldsymbol{s}}_A' \\ \boldsymbol{T}_2 = \left(\boldsymbol{c}_{1C,B} + \boldsymbol{c}_{2C,B}\right)^{\mathrm{T}} \mathrm{KH}_{C,A}' \\ \boldsymbol{W}_1 = \tilde{\boldsymbol{s}}_A'^{\mathrm{T}} \boldsymbol{Y}_1' \\ \boldsymbol{W}_2 = \mathbf{KH}_{C,A}^{\mathrm{T}} \boldsymbol{Y}_2' \end{cases} \tag{14}$$

定理 3　若 ZYF 方案是 IND-CCA2 安全的公钥加密方案，且 WI-2SPHF 是平滑投影哈希函数，那么经过上述改变后，图 1 所示的 2S-PAKE 协议在主动敌手攻击下是安全的两服务器 PAKE 协议。

证明　详见附录 2。

6　协议仿真与性能分析

本节将所提出的 WI-2SPHF 和 2S-PAKE 与其他相关方案进行对比。为实现公平对比，所有的仿真实验均使用 Python 语言，统一在 Intel(R) Core(TM) i5-4590 平台上进行，该目标平台的操作系统为 Windows7，频率为 3.3 GHz，内存为 8 GB。

为方便进行各类开销的对比，本节假设 κ=32 bit，m=800 bit，$\bar{m}$ =416 bit，n=32 bit，n_1=16 bit，n_2=16 bit，q=4 096 bit。该参数规模符合第 2.1 节中的参数设置要求。

6.1　格上平滑投影哈希函数的评估

本节首先对不同的 SPHF 进行仿真，以得到相应的执行时间；然后从安全级别和仿真时间等方面对比评估所提出的 WI-2SPHF。表 2 给出了格上不同 SPHF 的对比情况。

根据表 2 可知，只有 WI-2SPHF 具备 IND-CCA2 安全性。当哈希密钥长度大于 128 bit 时，WI-2SPHF-1S 计算开销最小，且 WI-2SPHF-2S 的计算总开销小于集中式的 KV.SPHF[24] 和 Reg.SPHF[12]；当哈希密钥长度为 512 bit 时，WI-2SPHF-1S 的计算开销比目前最优的 MP.SPHF[13]提高了约 27.6%。综上，本文方案具有高效性，这得益于本文的高效投影函数和两方 SPHF 避免了额外密码原语的使用。而且随着会话密钥长度的增加，WI-2SPHF-1S 计算开销的增幅最小，即其密钥长度可扩展性较好。

6.2　两服务器 PAKE 协议的评估

本节首先仿真不同的协议，以得到相应的执行时间、存储空间占用量和通信数据量；然后从执行

表 2 **格上不同 SPHF 的对比**

名称	集中式/两方	安全级别	仿真时间/s			
			1 bit	128 bit	256 bit	512 bit
KV.SPHF	集中式	IND-CCA1	0.001 243	0.159 105	0.318 210	0.636 420
MP.SPHF	集中式	IND-CCA1	0.000 034	0.004 343	0.008 686	0.017 373
Reg.SPHF	集中式	IND-CPA	0.004 779	0.611 750	1.223 500	2.447 000
SPKE.SPHF	集中式	IND-CPA	0.000 044	0.005 700	0.011 401	0.022 803
WI-2SPHF-1S	两方	IND-CCA2	0.000 303	0.003 352	0.006 430	0.012 584
WI-2SPHF-2S	两方	IND-CCA2	0.000 606	0.006 705	0.012 860	0.025 170

注：WI-2SPHF-1S 表示 WI-2SPHF 在一个服务器上的执行情况；WI-2SPHF-2S 表示 WI-2SPHF 在 2 个服务器上的执行情况

效率、存储与通信开销以及安全性等方面对比评估所提出的两服务器 PAKE 协议。

据本文所知，Roy 协议[6]是目前格上唯一可抵抗服务器泄露攻击的相关方案。若假设 Roy 协议是两服务器设置（实际上其不能用于两服务器场景），其需要调用 5 次全同态加密、2 次零知识证明、8 次签名/验签、2 次加/解密算法和 9 次矩阵运算，这对于资源受限的客户端而言，在很多应用场景中甚至不具备实用性。相比之下，在客户端，本文协议避免了全同态加密、零知识证明以及签名/验签算法的使用。在服务器端，本文协议在被动敌手攻击下进一步避免了秘密共享算法的使用，但所需的加/解密次数增加；在主动敌手攻击下，本文协议也需要使用零知识证明，但与 Roy 协议相比仍具有高效性。然而，本文协议中矩阵运算的次数高于 Roy 协议，这主要是因为本文协议是密钥交换协议，而 Roy 协议只是密钥传输协议。

文献[6]的主要贡献是提出一种秘密共享方案，Roy 协议则是由该方案导出的，协议架构复杂，需要反复使用大量额外的密码原语来保证安全性，这导致 Roy 协议的效率明显低于本文协议，且 Roy 协议仅是一种密钥传输方案，没有实现密钥交换。为进一步评估 2S-PAKE 协议的性能，表 3 给出了典型 PAKE 协议架构下不同协议之间的性能对比情况，其中仿真时间是指协议执行的总时间。

根据表 3 可知，虽然 2S-PAKE 是两服务器方案，但执行效率依然高于除 Li-Wang's PAKE1 外的所有单服务器协议。若简单地将 Li-Wang's PAKE1 协议扩展为两服务器协议，其在服务器与客户端的仿真时间都为 0.009 434 s，相比之下，2S-PAKE 在客户端和服务器端的仿真时间分别降低了 25.2%和 15.1%。这说明了本文方案的高效性，这主要得益于本文高效的 WI-2SPHF。

2S-PAKE 在客户与服务器间只需一轮通信，通信轮次最优，但其通信开销大于 KV.PAKE 外的其他协议，这主要是两服务器认证所致。因 Zhang-Yu's PAKE 和 Li-Wang's PAKE2 是两轮协议，其在通信时延上的优势小于其他协议。Katz et al.'s PAKE 的公钥长度较大，所以 2S-PAKE 的存储开销明显较小；除此之外，本文协议的存储开销在客户端与服务器端分别是其他协议的 1.12～2.05 倍和 1.12～2.01 倍，这主要是两服务器认证和公钥长度较大所致。在实

表 3 **典型 PAKE 协议架构下不同协议之间的性能对比**

协议	协议类型	仿真时间/s		轮数（次数）	通信开销/bit	存储开销/bit	
		客户端	服务器端			客户端	服务器端
Zhang-Yu's PAKE	单服务器	0.019 393	0.019 365	2(2)	80 652	810 696	810 696
Li-Wang's PAKE1	单服务器	0.004 717	0.004 717	1(2)	28 416	461 220	461 220
Li-Wang's PAKE2	单服务器	0.612 124	0.009 369	2(2)	28 416	452 016	461 616
Benhamouda et al.'s PAKE	单服务器	0.018 751	0.018 656	1(2)	85 248	825 636	825 636
Katz et al.'s PAKE	单服务器	0.102 555	0.102 583	3(3)	931 852	32 573 884	32 573 884
2S-PAKE	两服务器	0.007 706	0.008 010	1(4)	131 472	926 976	926 784

注：除本文的 2S-PAKE 协议外，其他协议的开销还需要在表中数据的基础上添加由签名/验签等算法引入的额外部分

际应用时，本文协议在各开销上的劣势将会减小，因为其他协议的开销要在表 3 中数据的基础上添加由签名/验签等算法引入的额外部分。在主动敌手攻击下，本文协议在服务器端也需要额外计入由零知识证明带来的开销，因此性能有所下降。

表 4 给出了不同协议的安全性及其他性能对比。其中轮数（次数）特指服务器与客户之间的通信轮次。轮数是指通信双方之间双向通信的数量，次数是指单向通信的数量。若是异步通信，二者相等；若是同步通信，后者是前者的两倍。

与 Yi1[9]、Yi2[10]和 Raimondo et al.'s PAKE[11]协议相比，本文协议和 Roy 协议[6]基于格困难问题，可以抵抗量子攻击。而本文协议与 Raimondo et al.'s PAKE 是唯口令的 PAKE 协议，不需要基于 PKI 模型（Roy 协议）与服务器身份（Yi1、Yi2 协议），因此这 3 个协议所对应的安全系统具有更强的可部署性。本文协议与 Yi1、Yi2 协议是两服务器认证协议，不适用于多服务器场景，而 Roy、Raimondo et al.'s PAKE 协议是阈值方案，不适用于两服务器及服务器数目小于相应阈值的应用场景。

根据表 4 可知，只有本文的 2S-PAKE 协议在被动敌手攻击下不需要使用零知识证明与签名/验签等来保证安全性，提高了执行效率并降低了通信与存储开销。而在主动敌手攻击下，本文协议只需要在服务器端使用零知识证明。此外，Raimondo et al.'s PAKE 和 Roy 协议在每次执行时都需要多次使用秘密共享算法，这进一步增加了协议的开销。

与密钥传输协议 Yi1、Yi2 和 Roy 协议相比，本文协议和 Raimondo et al.'s PAKE 协议可实现密钥交换，会话密钥由所有参与方协商而成，而不是由一方选定。此外，Yi1 与 Yi2 协议是否需要使用随机预言机取决于所基于的密码原语，而 Raimondo et al.'s PAKE、Roy 和本文协议基于标准模型，直接避免了随机预言机的使用。随机预言机的安全性本身存疑，因为在实际应用中需要使用具体的哈希函数替代随机预言机；而对于 PAKE 协议而言，使用随机预言机可能导致更加严重的安全威胁，因为这可能导致 PAKE 协议遭受离线口令猜测攻击。

综上，本文提出的 SPHF 和 PAKE 协议基于格困难问题，只需要采用矩阵、向量等线性运算，与传统困难问题（如大整数分解和离散对数问题）采用的模幂运算相比，在实现上具有高效性，且与传统困难问题相比，格困难问题具有最坏情况与平均情况下的安全性归约，因此本文协议在应用时可以随机选取困难实例，这大大提高了协议使用的便利性。此外，由于本文协议是唯口令设置，所对应的安全系统不需要硬件部署，也不涉及用户不可撤销的隐私泄露问题，可部署性强。

7　结束语

本文基于 LWE 问题的加法同态属性，提出了一种格上 IND-CCA2 安全的非适应性两方 SPHF，可支持一轮两方 PAKE 协议的构造。本文还约束了该 SPHF 所基于的 PKE 中相关参数的大小，以消除 LWE 问题的不完全加法同态属性对 SPHF 正确性的影响，且该 SPHF 具有相对独立的研究价值，可应用于证据加密、零知识证明和不经意传输等领域。

在此基础上，为抵抗量子和服务器泄露攻击，本文分别针对被动和主动敌手的攻击，提出了格上可证明安全的唯口令两服务器 PAKE 协议，前者执行效率较高，后者安全性更高。本文提出的 2 个协议在客户与服务器之间都只需要一轮通信，不需要使用签名/验签、全同态加密和秘密共享等昂贵的密码原语，被动敌手攻击下的协议还避免了零知识证明的使用，因此大大降低了计算、通信和存储开销。据本文所知，本文提出的 SPHF 和 PAKE 协议分别

表 4　　**不同协议的安全性及其他性能对比**

协议	量子攻击	唯口令	服务泄露	签名/验签	零知识证明	服务器数目	轮数（次数）	安全模型	协议类型
Yi1	×	×	√	√	√	2	2(4)	—	密钥传输
Yi2	×	×	√	√	√	2	3(6)	—	密钥传输
Roy	√	×	√	√	√	*N*	3(6*N*)	标准模型	密钥传输
Raimondo et al.'s PAKE	×	√	√	√	×	*N*	3(6*N*)	标准模型	密钥交换
2S-PAKE（被动）	√	√	√	×	×	2	1(4)	标准模型	密钥交换
2S-PAKE（主动）	√	√	√	×	√	2	1(4)	标准模型	密钥交换

是第一个基于格的两方 SPHF 和两服务器 PAKE 协议。本文还在标准模型下，基于一种更加现实的两服务器 PAKE 安全模型对本文的 2 个协议进行了严格的安全性证明。实验结果表明，本文提出的两方 SPHF 和两服务器 PAKE 协议执行效率较高。此外，在基于口令的安全方案中，口令更新是一个重要问题。但据本文所知，目前还不存在基于格的分布式口令更新方案，这也是下一步的研究方向。

附录 1 WI-2SPHF 的正确性与平滑性证明

证明 1) 正确性证明

根据 SPHF 的定义，要证明 WI-2SPHF 的正确性，需要证明对于$\forall \boldsymbol{W}=(\text{label},(\boldsymbol{c}_1,\boldsymbol{c}_2,c_3,c_4),\text{pw})\in\mathbb{L}$，式(15)成立。

$$\Pr[\text{Ham}(\boldsymbol{h}_S,\mathbf{ph}_S)\geqslant\varepsilon\kappa]\leqslant\text{negl}(\kappa) \tag{15}$$

根据 WI-2SPHF 的参数设置，要保证式(15)成立只要保证 h_i^1 和 ph_i^1 满足 SPHF 的正确性要求即可。

$$\begin{aligned}
&h_i^1=(\boldsymbol{c}_1+\boldsymbol{c}_2\,\|\,\mathbf{0})^{\mathrm{T}}\mathbf{kh}_i(\bmod q)=\\
&((\boldsymbol{A}^{\mathrm{T}}\tilde{\boldsymbol{s}}+\boldsymbol{e}_1)+((\boldsymbol{B}+\text{FRD}(\text{tag})\boldsymbol{G}_b)^{\mathrm{T}}\tilde{\boldsymbol{s}}+\\
&\boldsymbol{e}_2)\,\|\,\mathbf{0})^{\mathrm{T}}\mathbf{kh}_i(\bmod q)=\\
&((\boldsymbol{A}+(\boldsymbol{B}+\text{FRD}(\text{tag})\boldsymbol{G}_b)|\mathbf{0})^{\mathrm{T}}\tilde{\boldsymbol{s}})^{\mathrm{T}}\mathbf{kh}_i+\\
&(\boldsymbol{e}_1+\boldsymbol{e}_2\,\|\,\mathbf{0})^{\mathrm{T}}\mathbf{kh}_i(\bmod q)\approx\\
&\tilde{\boldsymbol{s}}^{\mathrm{T}}((\boldsymbol{A}+(\boldsymbol{B}+\text{FRD}(\text{tag})\boldsymbol{G}_b)\,|\,\mathbf{0})^{\mathrm{T}}\mathbf{kh}_i)(\bmod q)=\\
&\tilde{\boldsymbol{s}}^{\mathrm{T}}\mathbf{kp}_i(\bmod q)=\mathbf{kp}_i^{\mathrm{T}}\tilde{\boldsymbol{s}}(\bmod q)=\text{ph}_i^1
\end{aligned} \tag{16}$$

由式(16)可知，本文提出的 WI-2SPHF 具备正确性。

2) 平滑性证明

平滑性要求即使投影密钥泄露，哈希值与随机数不可区分。首先证明对于$\forall \boldsymbol{W}=(\text{label},(\boldsymbol{c}_1,\boldsymbol{c}_2,c_3,c_4),\text{pw})\in\mathbb{X}/\mathbb{L}$，式(17)中的 2 个分布在统计上不可区分。

$$\begin{cases}\{(\mathbf{KP},\boldsymbol{h})\,|\,\mathbf{KH}\leftarrow\text{hashKG},\mathbf{KP}\leftarrow\text{projKG},\boldsymbol{h}\leftarrow\text{HASH}(\boldsymbol{W},\mathbf{KP})\}\\ \{(\mathbf{KP},\boldsymbol{h})\,|\,\mathbf{KH}\leftarrow\text{hashKG},\mathbf{KP}\leftarrow\text{projKG},\boldsymbol{h}\xleftarrow{\text{r}}\{0,1\}^{\kappa}\}\end{cases} \tag{17}$$

根据式(16)可知，$h_i^1=((\boldsymbol{A}+(\boldsymbol{B}+\text{FRD}(\text{tag})\boldsymbol{G}_b)|\mathbf{0})^{\mathrm{T}}\tilde{\boldsymbol{s}}+(\boldsymbol{e}_1+\boldsymbol{e}_2\,\|\,\mathbf{0}))^{\mathrm{T}}\mathbf{kh}_i(\bmod q)$，且投影密钥 $\mathbf{kp}_i=(\boldsymbol{A}+(\boldsymbol{B}+\text{FRD}(\text{tag})\boldsymbol{G}_b)|\mathbf{0})\mathbf{kh}_i$，且 $\tilde{s}$ 是随机的，所以即使敌手已知 kp_i 也不能学习到 h_i^1 的任何信息，因而也不能学习到 h_i 的任何信息。综上，在已知投影密钥 $\mathbf{KP}$ 的情况下，哈希值 $\boldsymbol{h}=(h_1,h_2,\cdots,h_l)$ 是 $\{0,1\}^{\kappa}$ 上的随机分布，这表示式(17)中的 2 个分布在统计上不可区分。

不妨设 A 泄露，本文构建模拟器 Sim 为敌手模拟两方 SPHF 的执行，且要求敌手无法区分真实的和模拟的两方 SPHF。Sim 应该在已知 $(\boldsymbol{h}_A,\mathbf{ph}_A,\mathbf{KP}_A)$ 和 WI-2SPHF 的输出 $(\boldsymbol{h}_S,\mathbf{ph}_S,\mathbf{KP}_S)$，但不知哈希密钥 $\mathbf{KH}_A$ 和秘密 $\tilde{\boldsymbol{s}}_A$ 的前提下，成功模拟 WI-2SPHF 的执行。

根据第 4 节中 WI-2SPHF 的定义，对于 $i\in[1,\kappa]$，模拟器 Sim 可以计算

$$\begin{cases}\mathbf{kp}_B^i=\mathbf{kp}_A^i\oplus\mathbf{kp}_S^i\\ \boldsymbol{h}_B^i=\boldsymbol{h}_A^i\oplus\boldsymbol{h}_S^i,\\ \mathbf{ph}_B^i=\mathbf{ph}_A^i\oplus\mathbf{ph}_S^i\end{cases} \tag{18}$$

根据式(18)可知，B 在不知道 A 的哈希密钥 $\mathbf{KH}_A$ 与秘密 $\tilde{\boldsymbol{s}}_A$ 的前提下，可以正确计算投影密钥 $\mathbf{KP}_B$、哈希值 $\boldsymbol{h}_B$ 与投影哈希值 $\mathbf{ph}_B$。因此，本文提出的 WI-2SPHF 是平滑的。

综上，定理 1 成立。

证毕。

附录 2 格上两服务器 PAKE 协议的正确性与安全性证明

证明 1) 正确性证明

与被动敌手攻击下的协议相比，主动敌手攻击下的协议只改变了 ExeAB，二者关于会话密钥的计算方式是等价的。因此，本文只给出被动敌手攻击下协议的正确性证明。

要证明协议正确，需证 $\mathbf{sk}_{C,A}=\mathbf{sk}_{A,C}$ 且 $\mathbf{sk}_{C,B}=\mathbf{sk}_{B,C}$，又协议关于 A 与 B 对称，所以只需证明 $\mathbf{sk}_{C,A}=\mathbf{sk}_{A,C}$（或 $\mathbf{sk}_{C,B}=\mathbf{sk}_{B,C}$）即可，下文给出 $\mathbf{sk}_{C,A}=\mathbf{sk}_{A,C}$ 的相关证明。

根据两方 SPHF 的正确性可知式(19)成立。

$$\begin{cases}\text{HASH}(\boldsymbol{W}_A,\mathbf{KH}_{C,A})=\text{projHASH}(\mathbf{KP}_{C,A},\boldsymbol{W}_A,\tilde{\boldsymbol{s}}_A)\\ \text{HASH}(\boldsymbol{W}_B',\mathbf{KH}_{C,B})=\text{projHASH}(\mathbf{KP}_{C,A},\boldsymbol{W}_B',\tilde{\boldsymbol{s}}_B')\\ \text{projHASH}(\mathbf{KP}_A,\boldsymbol{W}_{C,A},\tilde{\boldsymbol{s}}_{C,A})=\text{HASH}(\boldsymbol{W}_{C,A},\mathbf{KH}_A)\\ \text{projHASH}(\mathbf{KP}_B',\boldsymbol{W}_{C,A},\tilde{\boldsymbol{s}}_{C,A})=\text{HASH}(\boldsymbol{W}_{C,A},\mathbf{KH}_B')\end{cases} \tag{19}$$

根据图 1 中 $\mathbf{sk}_{C,A}$ 与 $\mathbf{sk}_{A,C}$ 的定义可得式(20)。

$$\begin{aligned}
&\mathbf{sk}_{C,A}=\boldsymbol{h}_{C,A}\oplus\mathbf{ph}_{C,A}=\\
&\quad\Big(\big(\text{HASH}(\boldsymbol{W}_A,\mathbf{KH}_{C,A})\oplus\text{HASH}(\boldsymbol{W}_B',\mathbf{KH}_{C,B})\big)\oplus\\
&\quad\big(\text{projHASH}(\mathbf{KP}_A,\boldsymbol{W}_{C,A},\tilde{\boldsymbol{s}}_{C,A})\oplus\\
&\quad\text{projHASH}(\mathbf{KP}_B',\boldsymbol{W}_{C,A},\tilde{\boldsymbol{s}}_{C,A})\big)\Big)\oplus\\
&\quad\Big(\big(\text{projHASH}(\mathbf{KP}_{C,A},\boldsymbol{W}_A,\tilde{\boldsymbol{s}}_A)\oplus\\
&\quad\text{projHASH}(\mathbf{KP}_{C,A},\boldsymbol{W}_B',\tilde{\boldsymbol{s}}_B')\big)\oplus\\
&\quad\big(\text{HASH}(\boldsymbol{W}_{C,A},\mathbf{KH}_A)\oplus\text{HASH}(\boldsymbol{W}_{C,A},\mathbf{KH}_B')\big)\Big)=\\
&(\mathbf{ph}_A\oplus\mathbf{ph}_B')\oplus(\boldsymbol{h}_A\oplus\boldsymbol{h}_B')=\\
&\quad(\mathbf{ph}_A\oplus\boldsymbol{h}_A)\oplus(\mathbf{ph}_B'\oplus\boldsymbol{h}_B')=\\
&\quad\mathbf{sk}_A\oplus\mathbf{sk}_B'=\mathbf{sk}_{A,C}
\end{aligned} \tag{20}$$

由式(20)可知图 1 所示的 2S-PAKE 是正确的。

2) 安全性证明

① 被动敌手攻击下协议的安全性证明

给定敌手$\mathcal{A}$攻击协议，Sim 为$\mathcal{A}$模拟协议的执行。下文

假设 A 泄露，并对协议进行安全性证明。

Sim 首先执行初始化阶段，包括选择公共参数，为客户 C 选择口令 pw_C，并为 A 和 B 分发口令份额 $\mathrm{pw}_{A,C}$ 和 $\mathrm{pw}_{B,C}$（满足 $(\mathrm{pw}_{A,C}+\mathrm{pw}_{B,C}) \bmod q=\mathrm{pw}_C$）等。

本文通过一系列实验来评估$\mathcal{A}$的优势，其中 P_0 表示敌手与真实协议交互的实验。具体地，本文通过证明相邻 2 个实验中的$\mathcal{A}$的优势差是可忽略的和确定$\mathcal{A}$在最后一个实验中的优势上限，来确定$\mathcal{A}$在 P_0 中的优势上限，从而证明协议的安全性。

实验 P_1。实验 P_1 与 P_0 的区别如下。

a) 若发生以下情况，敌手不会成功，且实验中止：msg_C 发生了重复；由 Sim 产生的 msg_A 或 msg_B 发生了重复。

b) 将未泄露服务器 B 计算 $\mathbf{sk}_A'$ 的方式替换为 $\mathbf{sk}_A'=\mathrm{projHASH}(\mathbf{KP}_A,\boldsymbol{W}_{C,A},\tilde{\boldsymbol{s}}_{C,A})\oplus\mathrm{HASH}(\boldsymbol{W}_A',\mathbf{KH}_{C,A})$。

注意 Sim 可以执行上述 b)操作，因为此时是被动敌手攻击，b)操作所需的参数对于 Sim 而言是已知的。

上述 a)中情况发生的概率是可忽略的，因而其带来的敌手优势变化也是可忽略的。相对于实验 P_0，实验 P_1 中 $\mathbf{sk}_B'$ 计算方式的改变只是概念上的，这种改变不会带来敌手优势的变化，所以

$$|\mathrm{Adv}_{\mathcal{A},\mathrm{P}_1}(\kappa)-\mathrm{Adv}_{\mathcal{A},\mathrm{P}_0}(\kappa)|\leqslant\mathrm{negl}(\kappa) \tag{21}$$

实验 P_2。改变 $\mathbf{sk}_B'$ 的计算方式，令其为一个随机数。利用标准的混合证明法，易知，只要 ENC 加密方案是安全的，那么该变化引起的敌手优势变化是可忽略的，即

$$|\mathrm{Adv}_{\mathcal{A},\mathrm{P}_2}(\kappa)-\mathrm{Adv}_{\mathcal{A},\mathrm{P}_1}(\kappa)|\leqslant\mathrm{negl}(\kappa) \tag{22}$$

为方便证明，下文将 msg 分为敌手产生的和预言机产生的两类。进一步，若敌手产生的 msg 对应合法的口令，那么称该 msg 为合法的，否则称其为非法的。假设模拟器 Sim 生成的公私钥对为 $(\mathbf{pk}_Z,\mathbf{sk}_Z)$（相关定义见 ZYF 方案），注意私钥 $\mathbf{sk}_Z$ 只在安全性证明中用于验证敌手生成的 msg 是否合法，而真实协议的执行并不需要私钥 $\mathbf{sk}_Z$。

实验 P_3。若 msg_C 由之前的预言机生成，那么在客户端选择与实例 $\Pi_B^{j_B}$ 互为伙伴关系的实例 Π_C^i，并令 $\mathbf{sk}_{B,C}^{j_B}=\mathbf{sk}_{C,B}^i$；改变 $\mathbf{sk}_B'$ 的计算方式，令 $\mathbf{sk}_B'=\mathbf{sk}_{C,A}\oplus\mathrm{HASH}(\boldsymbol{W}_A,\mathbf{KH}_{C,A})\oplus\mathrm{projHASH}(\mathbf{KP}_A,\boldsymbol{W}_{C,A},\tilde{\boldsymbol{s}}_{C,A})$。

在实验 P_2 与实验 P_3 中，$\mathbf{sk}_{B,C}^{j_B}=\mathbf{sk}_{C,B}^i$ 都成立，且当实例 Π_C^i 与实例 $\Pi_B^{j_B}$ 互为伙伴关系时，2 个实验都不会中止；此外，在实验 P_2 与实验 P_3 中，M_B 都是同一随机变量 $\boldsymbol{g}$ 的加密值，且 $\boldsymbol{g}$ 满足式(23)。

$$\mathbf{sk}_{C,A}=\boldsymbol{g}\oplus\mathrm{HASH}(\boldsymbol{W}_A,\mathbf{KH}_{C,A})\oplus\mathrm{projHASH}(\mathbf{KP}_A,\boldsymbol{W}_{C,A},\tilde{\boldsymbol{s}}_{C,A}) \tag{23}$$

根据以上证明可知

$$\mathrm{Adv}_{\mathcal{A},\mathrm{P}_3}(\kappa)=\mathrm{Adv}_{\mathcal{A},\mathrm{P}_2}(\kappa) \tag{24}$$

实验 P_4。若 msg_C 由敌手产生且非法，将 $\mathbf{sk}_{B,C}$ 替换为随机数，同时将 $\boldsymbol{M}_B$ 替换为随机数关于 $\mathbf{pk}_A$ 的加密值。

根据平滑投影哈希函数的平滑性，并由 msg_C 非法（$\boldsymbol{W}_{C,A}\in\mathbb{X}/\mathbb{L}$）可知，$h_A\leftarrow\mathrm{HASH}(\boldsymbol{W}_{C,A},\mathbf{KH}_A)$ 与随机数不可区分，所以 $\mathbf{sk}_{B,C}=\mathbf{sk}_B\oplus\mathbf{sk}_A'=\boldsymbol{h}_B\oplus\mathbf{ph}_B\oplus\mathbf{sk}_A'$ 与随机数不可区分。同理 M_B 的输入也与随机数不可区分。所以

$$\mathrm{Adv}_{\mathcal{A},\mathrm{P}_4}(\kappa)=\mathrm{Adv}_{\mathcal{A},\mathrm{P}_3}(\kappa) \tag{25}$$

实验 P_5。若 msg_C 由敌手产生且合法，那么直接宣布敌手成功；在其他情况下，敌手成功的定义不变。

该改变只会增加敌手的优势，所以

$$\mathrm{Adv}_{\mathcal{A},\mathrm{P}_4}(\kappa)\leqslant\mathrm{Adv}_{\mathcal{A},\mathrm{P}_5}(\kappa) \tag{26}$$

此时，若 msg_C 由之前的预言机生成，那么 $\mathbf{sk}_{B,C}^{j_B}=\mathbf{sk}_{C,B}^i$；若 msg_C 由敌手产生且非法，那么 $\mathbf{sk}_{C,B}$ 与 $\boldsymbol{M}_B$ 是随机的；若 msg_C 由敌手产生且合法，那么敌手成功，且实验结束；若 msg_C 重用，那么实验中止。

实验 P_6。服务器 B 改变其生成 ZYF 密文的方式，用非法口令（口令服从 CDF-Zipf 分布）的份额生成 ZYF 的密文。

基于 ZYF 方案的 IND-CPA 安全性，利用标准的混合证明法，以及本文关于语义安全的定义易知

$$|\mathrm{Adv}_{\mathcal{A},\mathrm{P}_6}(\kappa)-\mathrm{Adv}_{\mathcal{A},\mathrm{P}_5}(\kappa)|\leqslant\mathrm{negl}(\kappa) \tag{27}$$

实验 P_7。若 msg_A 与 msg_B 由敌手产生且非法，改变 msg_A 与 msg_B 的回复方式，将 $\mathbf{sk}_{C,B}$ 和 $\mathbf{sk}_{C,A}$ 设置为随机数。

由于 msg_A 非法，即 $\boldsymbol{W}_A\in\mathbb{X}/\mathbb{L}$ 且 $\boldsymbol{W}_A'\in\mathbb{X}/\mathbb{L}$，根据 SPHF 的平滑性可知，$\mathrm{HASH}(\boldsymbol{W}_A,\mathbf{KH}_{C,A})$ 和 $\mathrm{HASH}(\boldsymbol{W}_A',\mathbf{KH}_{C,A})$ 都与随机数不可区分。根据图 1 进一步可知，在客户端哈希值 $\boldsymbol{h}_{C,A}\leftarrow\mathrm{HASH}(\boldsymbol{W}_A,\mathbf{KH}_{C,A})\oplus\mathrm{HASH}(\boldsymbol{W}_B',\mathbf{KH}_{C,B})$ 与哈希值 $\boldsymbol{h}_{C,B}\leftarrow\mathrm{HASH}(\boldsymbol{W}_B,\mathbf{KH}_{C,B})\oplus\mathrm{HASH}(\boldsymbol{W}_A',\ \mathbf{KH}_{C,A})$ 都与随机数不可区分，那么 $\mathbf{sk}_{C,A}=\boldsymbol{h}_{C,A}\oplus\mathbf{ph}_{C,A}$ 和 $\mathbf{sk}_{C,B}=\boldsymbol{h}_{C,B}\oplus\mathbf{ph}_{C,B}$ 也都与随机数不可区分。所以

$$\mathrm{Adv}_{\mathcal{A},\mathrm{P}_7}(\kappa)=\mathrm{Adv}_{\mathcal{A},\mathrm{P}_6}(\kappa) \tag{28}$$

实验 P_8。若 msg_A 与 msg_B 由敌手产生且合法，那么直接宣布敌手成功，并中止实验；在其他情况下，敌手成功的定义保持不变。该改变只能增加敌手的优势，所以

$$\mathrm{Adv}_{\mathcal{A},\mathrm{P}_7}(\kappa)\leqslant\mathrm{Adv}_{\mathcal{A},\mathrm{P}_8}(\kappa) \tag{29}$$

实验 P_9。改变服务器 B 关于 ZYF 密文的生成方式，将其替换为非法口令（服从 CDF-Zipf 分布）份额的加密值。

根据 ZYF 方案的 IND-CCA2 安全性，并利用标准的混合证明方法，易知

$$|\mathrm{Adv}_{\mathcal{A},\mathrm{P}_9}(\kappa)-\mathrm{Adv}_{\mathcal{A},\mathrm{P}_8}(\kappa)|\leqslant\mathrm{negl}(\kappa) \tag{30}$$

在实验 P_9 中，除了敌手$\mathcal{A}$知晓其生成了合法的 msg_C（msg_A/msg_B）的情况外，协议的执行与合法口令（口令份额）无关，这说明即使敌手$\mathcal{A}$获得了泄露服务器 A 的口令份额 $pw_{A,C}$（$pw_{A,C}$ 是 $\mathbb{Z}_q$ 上的随机数），pw_C 也是安全的。此外，模拟器 Sim 除了检验 msg_C（msg_A/msg_B）的合法性外，也不会使用合法的口令（口令份额）。

令 GuessPW 表示以下事件：敌手发送了有效的 msg_C（msg_A/msg_B）。本文假设口令分布服从 CDF-Zipf 定律，那么

$$\Pr[\text{GuessPW}] \leqslant C'Q(\kappa)^{s'} + \text{negl}(\kappa) \tag{31}$$

这说明，虽然对于一个 msg_C 而言，敌手$\mathcal{A}$可以进行两次在线口令猜测，但是敌手$\mathcal{A}$要想成功，需要将有效的 msg_C 同时发送给服务器 A 和服务器 B，因此敌手$\mathcal{A}$实际上执行了两次在线口令猜测。

若敌手$\mathcal{A}$未能猜测出合法的口令，那么敌手$\mathcal{A}$只能通过 Test 预言机询问获得成功，所以有

$$\Pr[\text{Success}] = \Pr[b' = b] + \Pr[b' \neq b]\Pr[\text{GuessPW}] \tag{32}$$

进一步，根据 b 的随机性可知，敌手$\mathcal{A}$在实验 P_9 中的优势满足

$$\text{Adv}_{\mathcal{A},P_9}(\kappa) = \frac{1}{2}\Pr[\text{Success}] - 1 \leqslant C'Q(\kappa)^{s'} + \text{negl}(\kappa) \tag{33}$$

那么，根据实验 P_1～实验 P_9 的证明过程可知

$$\text{Adv}_{\mathcal{A},\Pi}(\kappa) \leqslant C'Q(\kappa)^{s'} + \text{negl}(\kappa) \tag{34}$$

综上，定理 2 成立。

②主动敌手攻击下协议的安全性证明

给定敌手$\mathcal{A}$攻击协议，假设模拟器 Sim 为敌手$\mathcal{A}$模拟协议的执行。同样假设服务器 A 泄露，并对主动攻击下的协议进行安全性证明（假设服务器 B 泄露的相关证明与此相同）。下文只给出与被动敌手攻击下安全性证明的不同之处。

在实验 P_1 中，改变 sk_A' 的计算方式，直接令 HASH 函数中的 $h_i^1 = T_{1i}$，令 projHASH 函数中的 $ph_i^1 = T_{2i}$，然后计算 $\mathbf{sk}_A' = \text{projHASH}(\mathbf{KP}_A, \boldsymbol{W}_{C,A}, \tilde{\boldsymbol{s}}_{C,A}) \oplus \text{HASH}(\boldsymbol{W}_A', \mathbf{KH}_{C,A})$ 在 P_3 中，改变 $\mathbf{sk}_B'$ 的计算方式，令 HASH 函数中的 $h_i^1 = T_{1i}$，令 projHASH 函数中的 $ph_i^1 = T_{2i}$，然后计算 $\mathbf{sk}_B' = \mathbf{sk}_{C,A} \oplus \text{HASH}(\boldsymbol{W}_A, \mathbf{KH}_{C,A}) \oplus \text{projHASH}\ (\mathbf{KP}_A, \boldsymbol{W}_{C,A}, \tilde{\boldsymbol{s}}_{C,A})$。

由证据不可区分协议 Σ 可知

$$\begin{cases} \exists \tilde{\boldsymbol{s}}_A', \text{ s.t. } \boldsymbol{T}_1' = \mathbf{KP}_{C,B}{}^{\mathrm{T}} \tilde{\boldsymbol{s}}_A' \\ \exists \mathbf{KH}_{C,A}', \text{ s.t. } \boldsymbol{T}_2' = \left(\boldsymbol{c}_{1C,B} + \boldsymbol{c}_{2C,B}\right)^{\mathrm{T}} \mathbf{KH}_{C,A}' \end{cases} \tag{35}$$

综上，定理 3 成立。

证毕。

参考文献：

[1] SHIN J S, JO M, HWANG J Y, et al. A verifier-based password-authenticated key exchange using tamper-proof hardware[J]. The Computer Journal, 2021, 64(8): 1293-1302.

[2] 汪定. 口令安全关键问题研究[D]. 北京: 北京大学, 2017.
WANG D. Research on key issues in password security[D]. Beijing: Peking University, 2017.

[3] SHOR P W. Algorithms for quantum computation: discrete logarithms and factoring[C]//Proceedings of the 35th Annual Symposium on Foundations of Computer Science. Piscataway: IEEE Press, 1994: 124-134.

[4] 叶茂. 基于格的口令认证密钥交换协议和相关加密算法研究[D]. 郑州: 信息工程大学, 2013.
YE M. Research on password-based authenticated key exchange protocols and associated encryption algorithms from lattices[D]. Zhengzhou: Information Engineering University, 2013.

[5] ASIF R. Post-quantum cryptosystems for Internet-of-things: a survey on lattice-based algorithms[J]. IoT, 2021, 2(1): 71-91.

[6] ROY P S, DUTTA S, SUSILO W, et al. Password protected secret sharing from Lattices[C]//Applied Cryptography and Network Security. Berlin: Springer, 2021: 442-459.

[7] HALEVI S, KRAWCZYK H. Public-key cryptography and password protocols[J]. ACM Transactions on Information and System Security, 1999, 2(3): 230-268.

[8] GONG L, LOMAS M A, NEEDHAM R M, et al. Protecting poorly chosen secrets from guessing attacks[J]. IEEE Journal on Selected Areas in Communications, 1993, 11(5): 648-656.

[9] YI X, HAO F, BERTINO E. ID-based two-server password-authenticated key exchange[C]//Computer Security - ESORICS 2014. Berlin: Springer, 2014: 257-276.

[10] YI X, RAO F Y, TARI Z, et al. ID2S password-authenticated key exchange protocols[J]. IEEE Transactions on Computers, 2016, 65(12): 3687-3701.

[11] RAIMONDO D M, GENNARO R. Provably secure threshold password-authenticated key exchange[J]. Journal of Computer and System Sciences, 2006, 72(6): 978-1001.

[12] LI Z P, WANG D. Two-round PAKE protocol over lattices without NIZK[C]//Information Security and Cryptology. Berlin: Springer, 2019: 138-159.

[13] LI Z P, WANG D. Achieving one-round password-based authenticated key exchange over lattices[J]. IEEE Transactions on Services Computing, 2022, 15(1): 308-321.

[14] BENHAMOUDA F, BLAZY O, DUCAS L, et al. Hash proof systems over lattices revisited[C]//Public-Key Cryptography - PKC 2018. Berlin: Springer, 2018: 644-674.

[15] ZHANG J, YU Y. Two-round PAKE from approximate SPH and instantiations from lattices[C]//International Conference on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2017: 37-67.

[16] KATZ J, MACKENZIE P, TABAN G, et al. Two-server password-only authenticated key exchange[J]. Journal of Computer and System Sciences, 2012, 78(2): 651-669.

[17] BELLOVIN S M, MERRITT M. Encrypted key exchange: password-based protocols secure against dictionary attacks[C]//Proceedings 1992 IEEE Computer Society Symposium on Research in Security and Privacy. Piscataway: IEEE Press, 1992: 72-84.

[18] KATZ J, OSTROVSKY R, YUNG M. Efficient pass-

word-authenticated key exchange using human-memorable passwords[C]//Lecture Notes in Computer Science. Berlin: Springer, 2001: 475-494.

[19] GENNARO R, LINDELL Y. A framework for password-based authenticated key exchange[J]. ACM Transactions on Information and System Security (TISSEC), 2006, 9(2): 181-234.

[20] JIANG S Q, GONG G. Password based key exchange with mutual authentication[C]//International Workshop on Selected Areas in Cryptography. Berlin: Springer, 2004: 267-279.

[21] GROCE A, KATZ J. A new framework for efficient password-based authenticated key exchange[C]//Proceedings of the 17th ACM conference on Computer and communications security. New York: ACM Press, 2010: 516-525.

[22] ABDALLA M, BENHAMOUDA F, POINTCHEVAL D. Public-key encryption indistinguishable under plaintext-checkable attacks[J]. IET Information Security, 2016, 10(6): 288-303.

[23] KATZ J, VAIKUNTANATHAN V. Round-optimal password-based authenticated key exchange[C]//Theory of Cryptography. Berlin: Springer, 2011: 293-310.

[24] KATZ J, VAIKUNTANATHAN V. Smooth projective hashing and password-based authenticated key exchange from lattices[C]// Advances in Cryptology - ASIACRYPT 2009. Berlin: Springer, 2009: 636-652.

[25] CRAMER R, SHOUP V. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption[C]// Advances in Cryptology - EUROCRYPT 2002. Berlin: Springer, 2002: 45-64.

[26] PEIKERT C, VAIKUNTANATHAN V, WATERS B. A framework for efficient and composable oblivious transfer[C]//Advances in Cryptology - CRYPTO 2008 Berlin: Springer, 2008: 554-571.

[27] MICCIANCIO D, PEIKERT C. Trapdoors for lattices: simpler, tighter, faster, smaller[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2012: 700-718.

[28] BLAZY O, CHEVALIER C, DUCAS L, et al. Exact smooth projective hash function based on LWE[R]. 2013.

[29] 尹安琪, 曲彤洲, 郭渊博, 等. 格上基于密文标准语言的可证明安全两轮口令认证密钥交换协议[J]. 电子学报, 2021: doi.org/10.12263/DZXB.20210517.
YIN A Q, QU T Z, GUO Y B, et al. Provably secure two-round PAKE based on ciphertext standard language over lattices [J]. Acta Electronica Sinica, 2021: doi.org/10.12263/DZXB.20210517.

[30] BENHAMOUDA F, BLAZY O, CHEVALIER C, et al. New techniques for SPHFs and efficient one-round PAKE protocols[C]// Advances in Cryptology-CRYPTO 2013. Berlin: Springer, 2013: 449-475.

[31] CANETTI R, GOLDREICH O, HALEVI S. The random oracle methodology, revisited[J]. Journal of the ACM, 2004, 51(4): 557-594.

[32] ZHANG J, YU Y, FAN S Q, et al. Improved lattice-based CCA2-secure PKE in the standard model[J]. Science China Information Sciences, 2020, 63(8): 1-22.

[33] REGEV O. On lattices, learning with errors, random linear codes, and cryptography[J]. Journal of the ACM, 2009, 56(6): 1-40.

[34] BELLARE M, POINTCHEVAL D, ROGAWAY P. Authenticated key exchange secure against dictionary attacks[C]//International conference on the theory and applications of cryptographic techniques. Berlin: Springer, 2000: 139-155.

[35] WANG D, CHENG H B, WANG P, et al. Zipf's law in passwords[J]. IEEE Transactions on Information Forensics and Security, 2017, 12(11): 2776-2791.

[36] WANG D, WANG P. On the implications of Zipf's law in passwords[C]//Computer Security - ESORICS 2016. Berlin: Spinger, 2016: 111-131.

[37] BAUM C, BOOTLE J, CERULLI A, et al. Sub-linear lattice-based zero-knowledge arguments for arithmetic circuits[C]//Advances in Cryptology - CRYPTO 2018. Berlin: Springer, 2018: 669-699.

[作者简介]

尹安琪（1995− ），女，山东临沂人，信息工程大学博士生，主要研究方向为安全协议设计及格密码理论等。

郭渊博（1975− ），男，陕西周至人，博士，信息工程大学教授、博士生导师，主要研究方向为网络防御、数据挖掘、机器学习和人工智能安全等。

汪定（1985− ），男，湖北十堰人，博士，南开大学教授、博士生导师，主要研究方向为公钥密码学、系统安全、人工智能等。

曲彤洲（1994− ），男，辽宁铁岭人，信息工程大学博士生，主要研究方向为网络安全和安全专用芯片设计等。

陈琳（1975− ），女，河南开封人，博士，信息工程大学副教授、硕士生导师，主要研究方向为信息安全、安全专用芯片设计等。