中国科学: 信息科学 2021 年 第 51 卷 第 8 期: 1375–1390
SCIENTIA SINICA Informationis
论文

基于格的口令散列方案

李增鹏[1], 汪定[2,3*]

1. 山东大学网络空间安全学院, 青岛 266237
2. 南开大学网络空间安全学院, 天津 300071
3. 天津市网络与数据安全技术重点实验室 (南开大学), 天津 300350
* 通信作者. E-mail: wangding@nankai.edu.cn

收稿日期: 2020–06–16; 接受日期: 2020–08–19; 网络出版日期: 2021–08–09

国家自然科学基金 (批准号: 61802006, 61802214)、山东省国家自然科学基金 (批准号: ZR2019BF009) 和青岛市应用基础研究计划青年专项基金 (批准号: 19-6-2-6-cg) 资助项目

摘要 在可预见的未来, 口令仍将是最主要的身份认证方法. 口令认证密钥交换协议 (password authenticated key exchange, PAKE) 是口令认证的重要组成部分, 它允许通信双方在不安全的通话信道上建立一个安全的会话密钥. 为了缓解服务器被入侵后对存储在服务器上口令的影响, 将口令散列之后再存储被广泛推荐, 例如使用传统的口令散列函数, 如 PBKDF2, Bcrypt, 和 Scrypt. 然而, 这些口令散列函数依赖复杂的数学问题, 安全性证明建立在随机预言机模型 (random oracle model, ROM) 之上, 且需要较大内存支持. 为解决上述问题, 基于离散对数假设的口令散列方案陆续被提出, 如 Benhamouda-Pointceva 方案 (IACR ePrint2013/833)、Kiefer-Manulis 方案 (ESORICS'14)、Pointcheval-Wang 方案 (ASIACCS'17) 与平滑投影散列函数 (smooth projective hash function, SPHF) 集成, 但这些方案无法实现后量子安全且仍依赖于 ROM 模型. 因此, 本文着重研究如何在标准模型下设计后量子安全的口令散列方案, 并给出可证明安全性分析. 尽管所提方案尚不能应用于实际, 但为构造实际的后量子安全的口令认证及密钥交换协议奠定了基础.

关键词 抗量子, 口令认证密钥交换, 口令散列方案, 平滑投影散列函数, 基于格的密码学

1 引言

自 20 世纪 70 年代计算机诞生以来, 口令始终是最主要的身份认证方法之一. 然而, 口令通常是由用户生成的低熵字符串, 易被攻击者猜测. 并且, 随着用户网络服务的增多, 需要记忆和管理的口令也随之增多, 使得 "可记忆 vs. 抗猜测" 的这一矛盾日益突出. 为此, 学者们陆续提出了众多的新型身份认证方案, 如图形口令、基于硬件、基于生物特征等 [1~3] 这些新型认证技术. 虽然这些新型的认证技术在安全性或可用性方面优于传统口令认证技术, 但在可部署性及便携性等方面劣于口令 [4]. 比如,

引用格式: 李增鹏, 汪定. 基于格的口令散列方案. 中国科学: 信息科学, 2021, 51: 1375–1390, doi: 10.1360/SSI-2020-0177
Li Z P, Wang D. Achieving password-hashing scheme over lattices (in Chinese). Sci Sin Inform, 2021, 51: 1375–1390, doi: 10.1360/SSI-2020-0177

基于硬件的认证方法使用不方便, 基于生物特征的认证方法存在隐私泄露问题 [2,3], 这些固有缺陷使得学术界逐渐形成一个共识 [4,5]: 即在可预见的未来, 口令仍将是最主要的身份认证方法.

当前, 口令认证主要面临的一大威胁是猜测攻击, 包括在线猜测攻击和离线猜测攻击. 尽管口令被散列后存储, 攻击者仍可以对散列后的口令发起猜测攻击, 如使用基于上下文无关文法的攻击 [6]、基于深度学习的定向攻击 [7] 等方法. 为防止在线猜测攻击, 学者们提出了 Honeywords [8,9] 的概念, 它通过将真实的口令嵌入在 Honeyword 中, 迫使在线猜测攻击失效; 为抵抗离线猜测攻击, 采用带盐的口令散列函数来散列输入的口令 [10]. 执行口令散列功能的基本目的是使攻击者获取口令文件后无法直接得到明文口令, 同时保证了口令在服务器端的机密性; 执行加盐口令散列函数从一定程度上增加了攻击者猜测口令的难度, 但攻击者仍然可以通过离线猜测攻击反向猜测出口令. 需说明的是, 盐值通常为长度固定、中等熵值的随机字符串 (如 NIST SP800-63B 推荐使用 32 bit 熵值的盐) [10]. 盐作为口令散列函数的一个输入, 使得相同的口令得到不同的散列输出值 [11,12]. 攻击者实施离线猜测攻击时, 首先生成一个猜测序列; 然后, 按从前向后的顺序, 将该猜测序列中的每一个猜测与目标账户的盐值一起输入到认证系统使用的散列函数, 将计算得到的散列值与直接从系统得到的散列值进行比对; 如果二者匹配成功, 则口令猜测成功, 停止; 否则, 继续将猜测序列中的下一个猜测, 与目标账户的盐值一起输入到散列函数, 这样循环直到猜出口令或猜测序列尝试完毕 (即猜测失败). 值得注意的是, 口令 (加盐) 散列后存储的系统中, 需要设计非对称的口令认证协议 (asymmetric PAKE) [13,14].

一般来说, 设计口令认证密钥交换 (password authenticated key exchange, PAKE) 协议有两种不同的技术路线. 一种是基于随机预言机模型中的数论假设, 利用抗碰撞伪随机函数 (pseudorandom function, PRF). 另一种是在标准模型中, 利用散列证明系统, 如平滑投影散列函数 (smooth projective hash function, SPHF) [15]. 现存的口令散列函数 (如 PBKDF2 和 Bcrypt) 多讨论如何在随机预言机模型下基于数论假设的构造, 以及如何集成到随机预言机模型下的 PAKE 协议中 [16~19]. 但对于平滑投影散列函数是否可以有效地集成到标准模型下 PAKE 协议的讨论则相对较少 [15]. Benhamouda-Pointcheval [20] 提出了一种在随机预言机模型下的口令散列方案 (password hashing scheme, PHS), 并首次讨论了将该方案集成到基于平滑投影散列函数的 PAKE 的可能性. 后续工作 [21~23] 去除了随机预言机, 提出了在离散对数假设下基于 Pedersen 承诺方案的口令散列方案及优化方案, 但是这些方案 [21~23] 的构造依赖于随机预言机模型, 尚无法保证在量子时代的安全性. Nguyen 等 [24] 利用基于格上短整数解 (short integer solution, SIS) 困难问题的 Kawachi-Tanaka-Xagawa (KTX) 承诺方案, 提出了一种基于格的零知识口令策略检测协议, 该协议可以看作是一种基于格的口令散列方案的雏形. 受该方案启发, 本文研究了如下问题:

有没有可能在标准模型下设计一种抗量子的口令散列方案?

为填补这一空白, 本文尝试在格上构造 3 种有效的口令散列方案, 为后续进一步集成到 PAKE 协议提供选择. 基于格的口令散列方案的主要优点如下: (1) 系统化口令认证协议, 并集成到任意的对称式 PAKE 的认证阶段, 从而得到非对称 PAKE 协议. (2) 抗量子的口令散列方案, 并抵抗已知攻击 (如离线字典攻击). 本文主要工作概述如下: 第 2 节讨论口令散列函数构造所需的基础密码学组件. 第 3 节讨论已有的基于数论假设的口令散列方案. 第 4 节将具体讨论所提出的 3 种口令散列方案, 其中, 4.1 小节通过 CDGLW 承诺方案实现口令哈希, 4.2 小节通过 BDLOP 承诺方案实现口令散列, 4.3 小结通过 KTX 承诺方案实现口令散列. 第 5 节对格上基于承诺的口令散列方案进行系统性的对比分析, 包括具体的特性、安全问题和挑战. 第 6 节总结基于格的口令散列方案, 并展望下一步研究工作.

1.1 本文贡献

• 重新定义口令散列方案的接口. 观察到, Kiefer-Manuals 方案 [21,22] 提出的口令散列方案, 其盐值不具有不可区分的特性, 与 Benhanmouda-Pointcheval 方案 [20] 提出的原始定义不同. 这在一定程度上降低了方案安全性, 为在标准模型中获得抗量子口令散列方案, 结合 Benhanmouda-Pointcheval 方案 [20]、Kiefer-Manuals 方案 [21,22], 本文重新修订口令散列方案的定义, 并修正了口令散列方案的接口, 以便可以有效地集成到非对称 PAKE 协议中.

• 提出基于格的 Pedersen 承诺口令散列方案. 观察到, 现有的基于离散对数的口令散列方案 [21,22] 均使用 Pedersen 承诺方案设计而来. Nguyen 等 [24] 利用格上基于 SIS 问题的 KTX 承诺, 提出了一种基于格的口令散列协议, 该协议具有零知识口令策略检查的特性, 其构造依赖于复杂的随机排列, 但放松了安全性要求, 使其盐值不具有不可区分性的特性. 此外, 非对称 (或称基于验证因子的, verifier-based) PAKE 协议 [20,25,26] 要求服务器端存储与口令相关的验证信息 (例如, 带有与盐值相关的随机口令散列). 因此, 为保证基于格的口令散列方案满足 PAKE 协议的要求, 本文遵循基于 Pedersen 类口令散列方案 [21,22] 的技术路线, 提出了 3 种基于格上承诺口令散列方案. 相较于 Nguyen 等 [24] 复杂的 permutation 操作, 本文方案具有更高的计算效率.

• 无损安全性节省存储空间. 综合考虑安全性、灵活性和性能等因素, 口令散列函数应满足: (1) 输入为 0 到 128 字节之间的任意长度的口令字符串, (2) 至少 16 字节的盐值, (3) 32 字节的输出长度, (4) 配置时间 (t_{cost}) 和内存需求 (m_{cost}) 等一个或多个参数. 现有散列函数的构造多依赖于巨大内存空间, 在标准模型下, 基于格的承诺式口令散列结构, 在不影响安全性的前提下, 输出长度小于 32 字节, 可节省更多的内存空间.

1.2 相关工作

1.2.1 (同态) 承诺方案

非交互式承诺方案意味着在承诺阶段, 秘密消息 m 可被很好地隐藏, 在随后的打开阶段, 消息 m 和随机值 r 则会被公开. 承诺方案的绑定性质意味着不能使用两个不同的消息打开同一个承诺. 此外, 同态承诺意味着如果对两个承诺进行运算 (加或乘), 将得到一个新的承诺, 其中包含两个消息的结果 (和或积). 同态密码体制, 如 ElGamal [27], Paillier [28], BGN [29], 可以被视为具有完美绑定性和计算隐藏性的同态承诺方案. 此外, 还有许多现有的同态承诺方案, 如 Fujisaki-Okamoto 方案 [30]、Damgard-Fujisaki 方案 [31]、Damgard-Nielsen 方案 [32]、Boyen-Waters 方案 [33]、Groth 方案 [34] 以及 Sahai 方案 [35] 等等. 然而, 所有这些承诺方案都不能在量子时代保证安全.

1.2.2 主流口令散列方案

密钥散列消息认证码 (hash-based message authentication codes, HMAC) 是密钥派生函数 (key derivation functions, KDFs) 的主要组成部分, KDFs 以口令为输入并输出一个或多个密钥. Bcrypt, Scrypt 和 Argon2 是当前经典的口令散列函数解决方案, 在口令加密和存储等应用中应用广泛. 根据调查, 文献 [36] 的表 1 显示了密码泄漏问题, 文献 [37] 的表 2 显示了近 59% 的 web 服务采用不安全的口令散列函数 (如 MD5 和 SHA1) 或直接以明文的方式存储口令. 目前, 绝大多数应用程序只使用 SHA2 或 MD5 来散列用户口令, 只有少数应用程序使用慢散列高耗内存的散列函数, 如 PBKDF2 [38], 该类函数具有以下结构: $\mathsf{key} = \text{PBKDF2}(\mathsf{hLen}, \mathsf{pwd}, \mathsf{salt}, c, \mathsf{dkLen})$, 其输出一个派生秘钥 key. PBKDF2 密钥派生函数有 5 个输入参数来派生密钥: (1) 输出长度 hLen (例如主要的 HMAC); (2) 口令 (即 pwd); (3)

盐值 salt; (4) 所需的迭代次数 c; (5) 派生密钥的比特位长度 dkLen. 值得注意的是, Argon2 是口令散列竞赛 (password hashing competition, PHC) 的最终赢家 [39], 它适合于需要可证明的高内存使用的场景, 例如口令散列、密钥派生和密码学货币.

总之, 口令散列函数主要应用于 web 服务、主流系统和嵌入式应用的通用服务中 [40]. 在随机预言机模型下, 上述 KDFs (例如 PBKDF2, Bcrypt, Scrypt 和 Argon2) 的数学结构非常精炼 (相比非对称加密方案), 通常认为它们 (与其他对称算法相同) 没有具体针对它们的量子攻击算法. 此外, Leurent-Nguyen 方案 [41] 指出了随机预言机模型的一些潜在风险, 因此最近提出的口令散列函数 (例如文献 [21,22]) 多集中在标准模型下, 除继承传统 KDFs 的主要特性, 还应满足口令隐藏、原像性、弱抗碰撞性、预散列熵和熵保持等基本特性. 但标准模型下口令散列函数无法实现量子安全.

2 预备知识

定义安全参数 λ, 粗体小写字母表示向量, $\boldsymbol{x}$, 粗体大写字母表示矩阵, $\boldsymbol{A}$.

定义1 (承诺方案) 一个非交互的承诺方案包含 3 个概率多项式 (Setup, Commit, Open) 子算法, 针对消息空间 $\mathcal{M}$, 3 个子算法的接口定义如下:

- $\mathrm{ck} \leftarrow \mathrm{Setup}(1^{\lambda})$ 生成公共承诺密钥 ck.
- $(c,d) \leftarrow \mathrm{Commit}_{\mathrm{ck}}(m)$ 针对承诺的消息 m 生成承诺值 c 和打开值 d.
- $\bar{m} \in \mathcal{M} \cup \{\perp\} \leftarrow \mathrm{Open}_{\mathrm{ck}}(c,d)$, 承诺正确打开则输出消息, 否则输出 $\perp$, 此时 c 为无效的承诺值.

注意在下文中, 在上下文清楚的前提下, 我们通常省略公共承诺密钥 ck. 对于正确性和安全性的定义如下, 其中安全性通过隐藏性和绑定性来刻画.

正确性意味着对于任意的消息 m, $\mathrm{Open}_{\mathrm{ck}}(\mathrm{Commit}_{\mathrm{ck}}(m)) = m$.

隐藏性意味着对于敌手生成的任意两个消息 m_0 和 m_1, 敌手区分出它们相应承诺值 $c(m_0)$ 和 $c(m_1)$ 的概率是计算不可区分的, 记为 $c(m_0) \approx c(m_1)$, 对于任意由敌手选取的消息 m_0 和 m_1.

绑定性意味着对于同一个承诺值 c 不能被不同的消息 m_0 和 m_1 打开, 即, 如果敌手生成一个三元组 (c,d,d') 使得对于消息 m 和 m', (c,d) 和 (c,d') 都是有效的承诺值, 但 $m \neq m'$.

如 Pedersen 承诺, 设 $\mathbb{G}$ 是素数阶 $r \approx 2^{\lambda}$ 的群, 一个承诺方案包含以下两个阶段.

- **承诺阶段.** 接收者首先选择两个取自群 $\mathbb{G}$ 的生成元 (用 g 和 h 表示) 并将 g,h 发送给承诺者. 然后承诺者随机选择一个 $r \in [1,\ldots,r]$, 承诺消息 m 并发送 $(c,(r,m)) = \mathrm{Com}(m;r) = g^r \cdot h^m$.
- **打开阶段.** 承诺者向接收者发送消息 m 和随机数 r 用以打开承诺. 然后, 接收者验证 $g^r \cdot h^m \stackrel{?}{=} c$. 如果验证成功, 则输出 $m = \mathrm{Open}(c,(r,m))$, 否则输出 $\perp$.

定义2 (判定性 $\mathsf{LWE}_{n,q,\chi,m}$) 如果存在一个独立的分布 $(\boldsymbol{A},\boldsymbol{b}) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^{m\times 1}$, 那么存在如下两种不同的方式来生成此分布:

(1) 对于一个随机的 $\boldsymbol{s} \in \mathbb{Z}_q^n$, $\{(\boldsymbol{A},\boldsymbol{b}) : \boldsymbol{a} \leftarrow \mathbb{Z}_q^{m\times n}, \boldsymbol{s} \leftarrow \mathbb{Z}_q^{n\times 1}, \boldsymbol{e} \leftarrow \chi^{m\times 1}, \boldsymbol{b} = \boldsymbol{A}\cdot\boldsymbol{s} + \boldsymbol{e}\}$;

(2) 均匀分布 $(\{(\boldsymbol{A},\boldsymbol{b}) : \boldsymbol{A} \leftarrow \mathbb{Z}_q^{m\times n}, \boldsymbol{b} \leftarrow \mathbb{Z}_q^{m\times 1}\})$.

因此, 分布 (1) 与 (2) 是计算上不可区分的.

2.1 口令

口令是计算机系统中用户认证的第一道防线. 口令通常为一个低熵密钥, 由几个方便用户记忆的字符组成. 系统为每个注册用户分配一个帐户, 并存储每个帐户的用户名和散列的口令. 随后, 用户使用自己的用户名和口令登录, 服务器在对用户进行身份验证后即可访问相应的服务. 口令的另一个

重要应用是密钥生成, 特别是, 密钥派生函数 (key derivation function, KDF) [42] 生成的密钥用户加密会话数据. 文献 [43] 表示超过 90% 的用户选择不超过 10 个字符的口令. 此外, 用户选择的口令遵循 Zipf 定律 [44], 这就解释了为什么口令具有低猜测熵, 即大约 10 位的在线猜测安全性和大约 20∼22 位的离线猜测安全性 (参见文献 [45]). 此外, Wang 等 [46] 报告称, 人类选择的 4 字节口令 (一种特殊的口令) 可以分别提供 6.6 比特位的在线猜测安全性和 8.4 比特位的离线猜测安全性. 有关口令的详细解释参考文献 [40].

2.2 基于安全和威胁模型的口令散列方案

实现一个安全的口令散列方案并不是一个简单的任务. 为了设计一个合格的口令散列方案, 需要考虑口令散列方案是否满足 (1) 抗原像性, (2) 弱抗碰撞性, (3) 抗碰撞性等. 此外, 在设计新的口令散列方案时, 应避免 Merkle-Damgard 方案结构中的长度扩展攻击和部分消息碰撞等已知的密码学弱点. 本文采用了 Benhamouda-Pointcheval 方案 [20] 的技术路线, 结合作者提出的口令散列方案. 重新修订的口令散列方案定义包含以下 5 个多项式算法:

- $\mathsf{params} \leftarrow \mathsf{Setup}(1^\lambda, 1^\ell)$ 是一种概率算法, 输入安全参数 λ 和长度为 $\ell \leqslant \lambda$ 位的口令. 输出口令散列参数 params, 其中 params 包含随机盐空间 $\mathbb{S}_P$ 和 $\mathbb{S}_H$ 的隐式描述.
- $\overline{s} \leftarrow \mathsf{PreSalt}(\mathsf{params})$ 是一种概率算法, 输入参数 params, 并输出一个属于 $\mathbb{S}_P$ 的预盐值.
- $s \leftarrow \mathsf{Salt}(\mathsf{params})$ 是一种概率算法, 输入参数 params, 并在盐空间 $\mathbb{S}_H$ 中输出一个盐值 $s \in \mathbb{S}_P$ (取决于 params).
- $\overline{y} \leftarrow \mathsf{PreHash}(\mathsf{params}, \overline{s}, \mathsf{pwd})$ 是一种确定性算法, 输入参数 params、一个预盐值 $\overline{s}$ 和口令 pwd, 并输出预散列值 $\overline{y}$.
- $y \leftarrow \mathsf{PHash}(\mathsf{params}, \overline{s}, s, \mathsf{pwd})$ 是一种确定性算法, 输入参数 params、盐值 s 和预盐值 $\overline{s}$ 以及口令 pwd, 然后输出散列值 y.

为了方便描述, 在上下文清楚的前提下, 会省略 params.

安全属性. 算法 PreHash 不允许将盐值 s 作为输入. 因为 PreHash 是客户端执行的私有算法, 而盐值 s 是客户端在附加流中无法记忆的. 因此, 这将促使客户端在口令认证执行开始时从服务器请求获得盐值, 因此, 这将引入另一个从客户端到服务器的初始化轮次.

(1) 口令隐藏. 对于概率多项式时间 (或 PPT) 算法 $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, 其中 $\mathcal{A}_1(\mathsf{params})$ 输入 $\mathsf{params} \leftarrow \mathsf{Setup}(1^\lambda, 1^\ell)$ 并输出两个等长的口令 pwd_0 和 pwd_1, 然后对于 $y \leftarrow \mathsf{PreHash}(\mathsf{params}, \overline{s}, s, \mathsf{pwd})$ 令 $\mathcal{A}_2(y)$ 输出一个比特 b', 其中 $\overline{s} \leftarrow \mathsf{PreSalt}(\mathsf{params})$, $s \leftarrow \mathsf{Salt}(\mathsf{params})$ 和 $\overline{y} \leftarrow \mathsf{PreHash}(\mathsf{params}_{\mathsf{ph}}, \overline{s}, \mathsf{pwd}_b)$ 对于一个随机比特 $b \in \{0,1\}$, 存在一个可忽略的函数 $\mathrm{negl}(\lambda)$, 使得 $|\Pr[b = b'] - \frac{1}{2}| \leqslant \mathrm{negl}(\lambda)$.

(2) 盐值不可区分性. 对于任意参数 $\mathsf{params} \leftarrow \mathsf{Setup}(1^\lambda, 1^\ell)$, 以下两个分布是不可区分的, $\overline{s} \approx s$, 这里 $\overline{s} \leftarrow \mathsf{PreSalt}(\mathsf{params})$, $s \leftarrow \mathsf{Salt}(\mathsf{params})$.

(3) 抗原像性. 抗原像性又称之为紧单向性 (tight one-wayness), 即对于所有的 PPT 算法 $\mathcal{A}$ 至多运行 t 次, 存在一个可忽略的函数 $\mathrm{negl}(\lambda)$, 使得

$$\Pr[(i, \bar{y}) \leftarrow \mathcal{A}^{\mathsf{PHash}_{\bar{y}}(\cdot)}(\mathsf{params}); \mathsf{Finalise}(i, \bar{y}) = 1] \leqslant \frac{\alpha t}{2^\beta t_{\mathsf{PreHash}}} + \mathrm{negl}(\lambda),$$

其中 α 为较小的常数, t_{PreHash} 为 PreHash 的运行次数, 参数 $\mathsf{params} \leftarrow \mathsf{Setup}(\lambda)$. 此外, 参数 β 及其 2^β 是用来刻画所猜测的随机口令与真实口令相匹配的概率的. $\mathsf{Finalise}(i, \bar{y}) = 1$ 意味着当且仅当求第 i 个哈希函数 PHash 前向值时, PreHash 的输出等于 $\bar{y}$.

(4) 弱抗碰撞性. 存在一个可忽略的函数 $\mathrm{negl}(\lambda)$, 对于任意 $\mathsf{pwd} \in \{0,1\}^{\ell}$ 和任意无限制的敌手 $\mathcal{A}$, 使得能从均分分布中区分出 $\bar{y} \leftarrow \mathsf{PreHash}(\mathsf{params}, \mathsf{pwd}, \bar{s})$ 的概率是可忽略的 $\mathrm{negl}(\lambda)$.

(5) 预散列熵保持. 对于具有最小熵 θ 的多项式时间可抽样的所有口令分布 $\mathcal{D}$ 和任何 PPT 对手 $\mathcal{A}$, 存在一个可忽略的函数 $\mathrm{negl}(\lambda)$, 使得

$$\Pr\left[(\bar{s}, \bar{y}) \leftarrow \mathcal{A}(\mathsf{params}); s \in \mathbb{S}_H \bigwedge \bar{y} = \mathsf{PreHash}(\mathsf{params}, \bar{s}, \mathsf{pwd} \leftarrow \mathcal{D})\right] \leqslant 2^{-\theta} + \mathrm{negl}(\lambda).$$

(6) 熵保持. 存在一个可忽略的函数 $\mathrm{negl}(\lambda)$, 使得对于最小熵 θ 的任何口令分布 $\mathcal{D}$ (不一定在多项式时间内可抽样), 任意 (无限制) 的对手 $\mathcal{A}$ 满足:

$$\Pr\left[(s, y) \leftarrow \mathcal{A}(\mathsf{params}); (\bar{s} \in \mathbb{S}_H) \bigwedge (s \in \mathbb{S}_H \bigwedge y = \mathsf{PHash}(\mathsf{params}, s, \mathsf{pwd} \leftarrow \mathcal{D}))\right] \leqslant 2^{-\theta} + \mathrm{negl}(\lambda).$$

2.3 口令构成策略

为了防止口令猜测攻击, 一个简单的方法是强制用户使用安全性更强的口令, 并要求服务提供商部署更强大的口令散列函数. 在本文中口令合成策略定义遵循文献 [22] 的方法. 称元组 $f = (R, n_{\min}, n_{\max})$ 为口令策略, 其中 $n_{\min}$ 和 $n_{\max}$ 是口令的最小和最大长度, R 是 $\sum = \{d, u, l, s\}$ 上口令策略表达的简化正则表达式. R 与指定要实现正则表达式的字符串集相关, 其中这些字符串集包含 d 个数字, u 大写字母、l 小写字母和 s 符号. 例如, $R = dl$ 要求 pwd 至少包含一个数字和一个小写字母, $R = ssd$ 要求 pwd 至少包含两个符号和一个数字. 如果口令字符串 pwd 满足策略, 那么有 $f(\mathsf{pwd}) = \mathrm{TRUE}$ 来表示结果. 注意, $\sum$ (如 d, u, l 和 s) 是指编码字符集或 ASCII 字符集, 具体取决于上下文.

2.4 字典、口令分布和最小熵

为了便于解释口令字典的特性, 使用 $\mathcal{D}$ 来表示包含所有可能的 ASCII 字符组合的口令字典. $\mathcal{D}_{f,\ell}$ 包含所有口令组合策略 $\mathcal{D}_f = \{\mathsf{pwd} \in \mathcal{D} : f(\mathsf{pwd}) = \mathrm{TRUE}\}$ 以及其子集 (口令长度为 $\ell \in \mathbb{N}$ 的字典), 即 $\mathcal{D}_{f,\ell} = \{\mathsf{pwd} \in \mathcal{D} : f(\mathsf{pwd}) = \mathrm{TRUE} * |\mathsf{pwd}| = \ell\}$. 另外, 将口令 pwd 中的概率分布表示为 D_ω, 这意味着字符来自于字符集 $\omega \in \sum, d, u, l, s$. 然后根据 Shannon [47] 定义口令 $\mathsf{pwd} = (c_0, \ldots, c_{n-1})$ 的最小熵如 $\theta_{\mathcal{D}_{f,\ell}} = -\max_{\mathsf{pwd} \in \mathcal{D}_{f,\ell}}[D_{\sum}(c_i) \lg(D_{\sum}(c_i))]$.

注意, 为方便表述, 在上下文清楚的前提下, 本文忽略了最小熵定义中 $\mathcal{D}_{f,\ell}$ 的下标, 并简写成 $\mathcal{D}$ 或 $\mathcal{D}_f$.

2.5 基于口令散列方案的认证

本小节详细介绍优化的基于口令散列方案的认证方案, 包括注册和身份验证的各个阶段. 在正式介绍口令身份认证之前, 首先概述常规口令身份认证.

• 注册阶段, 客户机使用其用户名和口令 (即 uid 和 pwd) 在身份服务器上注册, 然后标识服务器在其本地数据库中存储相应的用户名和口令的散列值 $h = \mathsf{H}(\mathsf{pwd})$.

• 登录阶段, 客户端使用其 uid 和口令的散列值 h' 在服务器进行身份验证. 当服务器接收到 uid 和 h' 时, 服务器会检查在相应的 uid 下是否有 $h' = h$.

• 如果 (uid,h) 与 (uid,h') 匹配成功, 即 $h' = h$, 则服务器使用主密钥 msk 计算令牌 $\mathsf{tk}_{\mathsf{msk}} \leftarrow \mathsf{Auth}_{\mathsf{msk}}(x)$, 并将 $\mathsf{tk}_{\mathsf{msk}}$ 发送到客户端, x 包含客户端的信息、属性及 $\mathsf{tk}_{\mathsf{msk}}$ 中的附加信息 (例如, 有效时间等).

传统的口令认证方法中, 攻击者能够做到 (i) 恢复主密钥并伪造任意令牌, 以获得访问系统中任意资源和信息的权限; (ii) 获取口令散列, 以用作离线字典攻击的一部分, 恢复客户端凭据.

相较于如上概述常规口令身份认证, 基于口令散列方案的口令认证工作流程如下. 当用户想要登录服务时, 服务可以检查用户名、附加的盐和提供的口令. 然后, 服务将密码和盐一起做散列运算, 最后验证计算出的散列是否与存储的散列相匹配 [48].

注册阶段. 注册阶段之前, 需要初始化公共参数以及由服务器和公共参数 $\mathsf{params}_{\mathsf{ph}} \leftarrow \mathsf{Setup}(1^{\lambda}, 1^{\ell})$ 建立的口令策略 f. 最后, 对于客户机选择的口令, 仅当 $f(\mathsf{pwd}) = \mathrm{TRUE}$ 时, 服务器存储口令 pwd 的散列值 y. 需说明的是, 可以通过执行口令散列方案而不是对口令进行单向转换来获得口令散列.

(1) 服务器分配口令策略. 服务器 S 首先将其密码策略 f 分配给用户, 需要注册服务的客户 C 将根据制定的口令策略建立自己的口令.

(2) 客户通过其关联的口令进行注册. 合法的客户机 C 选择其相应的用户登录凭据并开始注册执行. 然后客户机使用唯一的 id 和相关联的口令 pwd 进行注册. 在接收到唯一的 id 和相关的口令 pwd 后, 服务器用盐 $s \leftarrow \mathsf{PHS.Salt}(\mathsf{params}_{\mathsf{ph}})$ 和预盐 $\bar{s} \leftarrow \mathsf{PHS.PreSalt}(\mathsf{params}_{\mathsf{ph}})$ 作为输入, 计算口令散列 $y_S \leftarrow \mathsf{PHash}(\mathsf{params}_{\mathsf{ph}}, s, \mathsf{pwd})$. 随后, 服务器存储 (y_S, s) 并删除 pwd, 并返回预盐 $\bar{s}$ 作为登录凭证.

认证阶段. 认证阶段发生在当客户端希望获得服务器所提供的服务时, 当客户端试图登录到服务器时, 客户端通过注册的 id 和相关的 pwd 向服务器进行身份验证. 随后, 客户机在通过身份验证的密钥交换阶段与服务器建立会话密钥.

(1) 当客户机 C 希望使用 pwd_C 向服务器 S 进行身份验证时, 它将生成一个登录命令 $\mathsf{cmd}_{\mathsf{login}} := (\mathrm{id}, \bar{y}, \bar{s})$ 对应于其口令的预散列值 $\bar{y}_C = \mathsf{PreHash}(\bar{s}_C, \mathsf{pwd}_C)$. 然后, 客户端将 $\mathsf{cmd}_{\mathsf{login}}$ 发送到服务器.

(2) 在接收到 $\mathsf{cmd}_{\mathsf{login}}$ 后, 服务器将打开承诺以获取 $\bar{y}_C$, 并从其密码数据库中获取散列值 $y_C := \mathsf{PHash}(s_S, \mathsf{pwd})$. 接下来, 如果客户端由服务器进行身份验证, 则它们是由发送到客户端的服务器生成的令牌 $\mathsf{tk}_{\mathsf{msk}}$. 如果接收到的令牌 $\mathsf{tk}_{\mathsf{msk}}$ 是合法的, 则表示该用户有资格登录.

3 基于数论假设的口令散列方案

本节着重讨论基于数论假设的口令散列方案, 首先回顾一下 Benhamouda-Pointcheval [20] 在随机预言机模型下的朴素口令散列方案和代数口令散列方案. 随后回顾 Kiefer-Manulis 基于 Pedersen 承诺的随机口令散列方案 [21].

3.1 朴素口令散列方案

本小节重新审视传统的口令散列方案. 本质上, 朴素口令散列方案类似常规口令散列方案 [49, 50], 其具有的基本接口定义如下:

- $\mathsf{params} \leftarrow \mathsf{Setup}(1^{\lambda})$ 输出 params.
- $(\bar{s}, \mathcal{T}_{\mathsf{ph}}) \leftarrow \mathsf{PreSalt}(\mathsf{params})$ 输出 $\bar{s} \in \mathcal{S} \in \{0,1\}^{\lambda}$ 和 $\mathcal{T}_{\mathsf{ph}} = \perp$.
- $s \leftarrow \mathsf{Salt}(\mathsf{params})$, 这里 $s := \mathcal{S} \in \{0,1\}^{\lambda}$.
- $\bar{y} \leftarrow \mathsf{PreHash}(\mathsf{params}, \bar{s}, \mathsf{pwd})$ 输出 $\bar{y} = \mathsf{pwd}$.
- $y \leftarrow \mathsf{PHash}(\mathsf{params}, s, \mathsf{pwd})$ 输出 $\mathsf{H}(s, \mathsf{pwd}) = y$.

3.2 随机预言机模型下的代数口令散列方案

Benhamouda-Pointcheval [20] 在随机预言模型中引入了一种代数密码散列方案. 该方案具有以下属性: 口令隐藏性、盐值不可区分性、抗原像性、弱抗碰撞性、预散列熵保持和熵保持性.

- $\mathsf{params}_{\mathsf{ph}} \leftarrow \mathsf{Setup}(1^\lambda)$ 输出一个参数 $\mathsf{params}_{\mathsf{ph}} := (\mathbb{G}, g)$, 其中 $\mathbb{G}$ 是一个 r (大于 2λ 位的素数) 阶的乘法循环群, g 是群 $\mathbb{G}$ 的生成元, 且设 H 是一个在 $\mathbb{Z}p$ 中有值的随机预言机.
- $s \leftarrow \mathsf{Salt}(\mathsf{params}_{\mathsf{ph}})$ 选择一个随机的 $h \in \mathbb{S}_H = \mathbb{G} \setminus \{1\}$ 并且输出 $s := h$.
- $(\bar{s}, \mathcal{T}_{\mathsf{ph}}) \leftarrow \mathsf{PreSalt}(\mathsf{params}_{\mathsf{ph}})$ 选取随机标量 $\mathcal{T}_{\mathsf{ph}} \in \mathbb{Z}_p^*$ 并且输出 $\bar{s} = h := g^{\mathcal{T}_{\mathsf{ph}}}$ 和一个陷门 $\mathcal{T}_{\mathsf{ph}}$, 这里 $h = g^{\mathcal{T}_{\mathsf{ph}}}$ 表示 $s := \bar{s}$.
- $\bar{y} \leftarrow \mathsf{PreHash}((\bar{s}, \mathcal{T}_{\mathsf{ph}}), \mathsf{pwd})$ 计算并输出 $\bar{y} := (\bar{s}^{(\mathcal{T}_{\mathsf{ph}}^{-1})})^{\mathsf{H}(\mathsf{pwd})} = g^{\mathsf{H}(\mathsf{pwd})}$.
- $y \leftarrow \mathsf{PHash}(s, \mathsf{pwd})$ 计算并输出 $y := s^{\mathsf{H}(\mathsf{pwd})} = h^{\mathsf{H}(\mathsf{pwd})}$.
- $\mathsf{Checkable}(\mathsf{params}, \mathcal{T}_{\mathsf{ph}}, s, \bar{y}, y)$ 检查 $\bar{y}^{\mathcal{T}_{\mathsf{ph}}} = (\bar{s})^{\mathsf{H}(\mathsf{pwd})} = h^{\mathcal{H}_{\mathsf{pwd}}} = y$.

3.3 来自基于 Pedersen 承诺的随机口令散列

Kiefer-Manulis 方案 [21] 在 ESORICS'14 利用 Pederson 承诺方案构造了一种随机口令散列方案. 该方案同样具有: 口令隐藏性、盐值不可区分性、抗原像性、弱抗碰撞性、预散列熵保持性和熵保持性.

- $\mathsf{params}_{\mathsf{ph}} \leftarrow \mathsf{Setup}(1^\lambda)$ 输入一个安全参数 λ, 输出公共参数 $\mathsf{params}_{\mathsf{ph}} := (\mathbb{G}, g, h, p)$, 其中 $\mathbb{G}$ 是一个 r (大于 2λ 位的素数) 阶的乘法循环群, g, h 是两个长度为 λ 的 r 阶素数群 $\mathbb{G}$ 的生成元, 设 H 是一个在 $\mathbb{Z}_p$ 上的随机预言机.
- $s \leftarrow \mathsf{Salt}(\mathsf{params}_{\mathsf{ph}})$ 输出一个散列盐值 s.
- $\bar{s} \leftarrow \mathsf{PreSalt}(\mathsf{params}_{\mathsf{ph}})$ 输出一个预散列盐值 $\bar{s}$.
- $\bar{y} \leftarrow \mathsf{PreHash}(\mathsf{params}_{\mathsf{ph}}, \bar{s}, \mathsf{pwd})$ 计算并输出预散列值 $\bar{y} := g^{\bar{s} \cdot \mathsf{pwd}}$.
- $y \leftarrow \mathsf{PHash}(\mathsf{params}_{\mathsf{ph}}, s, \bar{s}, \mathsf{pwd})$ 计算并输出散列值 $y := (y_1, y_2) = (g^{\bar{s}}, g^{\bar{s} \cdot \mathsf{pwd}} \cdot h^s)$.

4 基于格的口令散列方案通用构造

本节着重介绍本文所提出的基于格的口令散列方案. 为了获得标准模型下 (无随机预言机) 的高效后量子安全的口令方案, 本文在前序工作 Kiefer-Manulis 方案 [21] 的基础上, 首先引入了一个基于格上标准假设的同态承诺方案作为重要的密码学组件. 然后, 设计了 3 种不同的比特承诺式口令散列方案. 分别是基于 CDGLW 承诺方案 [51] 的口令散列方案、基于 BDLOP 承诺方案 [52] 的口令散列方案和基于 KTX 承诺方案 [53] 的口令散列方案.

4.1 基于 CDGLW 承诺的口令散列方案

为实现后量子安全的密码学承诺方案, 各种各样的基于格的承诺方案相继被提出, 如文献 [51~53] 等. 本小节着重介绍提出的基于 CDGLW 承诺方案 [51] 的口令散列方案, 该方案遵循 Kiefer-Manulis 方案 [21] 的研究路线, 将基于格的口令散列方案实例化. 具体构造如下.

4.1.1 CDGLW 承诺方案

Cabarcas 等 [51] 提出的具有无条件隐藏和后量子安全的承诺方案构造如下.

• $\mathsf{params} \leftarrow \mathsf{ComGen}(1^\lambda, 1^k)$. 设置合适的参数 $n, m, q \in \mathbb{Z}$ 和 $B, \sigma \in \mathbb{R}^+$. 取两个矩阵 $\boldsymbol{A}_1 \leftarrow \mathbb{Z}_q^{m\times n}$, $\boldsymbol{A}_2 \leftarrow \mathbb{Z}_q^{m\times k}$ 使得 $\boldsymbol{A} = (\boldsymbol{A}_1, \boldsymbol{A}_2) \in \mathbb{Z}_q^{m\times(n+k)}$ 具有平凡内核. 输出 $\mathsf{params} := (n, m, q, \boldsymbol{A})$.

• $(\mathsf{com}, \boldsymbol{s}) \leftarrow \mathsf{Com}(\boldsymbol{m} \in \mathbb{Z}_q^n; \boldsymbol{s})$. 取 $\boldsymbol{s} \leftarrow \mathbb{Z}_q^k$ 和 $\boldsymbol{e} \leftarrow \mathcal{D}_\sigma^m$, 计算 $\mathsf{com} := \boldsymbol{A}_1 \cdot \boldsymbol{m} + \boldsymbol{A}_2 \cdot \boldsymbol{s} + \boldsymbol{e} \in \mathbb{Z}_q^m$, 输出 $(\mathsf{com}, \boldsymbol{s})$.

• $\mathsf{Open}(\mathsf{com}, \boldsymbol{s}, \boldsymbol{m})$. 如果 $\|\mathsf{com} - \boldsymbol{A}_1\boldsymbol{m} - \boldsymbol{A}_2\boldsymbol{s}\| \leqslant B$ 则返回 $\boldsymbol{m}$, 否则返回 $\perp$.

4.1.2 基于 CDGLW 承诺的口令散列方案实例

在 CDGLW 承诺方案的基础上, 提出一种新的口令散列方案, 具体构造如下.

• $\bar{s} \leftarrow \mathsf{PreSalt}(\mathsf{params})$ 选择一个随机向量 $\boldsymbol{s} \leftarrow \mathbb{Z}_q^k$ 并且设置预盐 $\bar{s} := \boldsymbol{s}$.

• $s \leftarrow \mathsf{Salt}(\mathsf{params})$ 在高斯分布 $\mathcal{D}_\sigma^k$ 选择一个随机矩阵 $\boldsymbol{B} \in \mathbb{Z}_q^{m\times k}$ 和一个误差向量 $\boldsymbol{e}_0$, 然后输出盐 $s := \boldsymbol{B} \cdot \boldsymbol{e}_0 \in \mathbb{Z}_q^m$.

• $\bar{y} \leftarrow \mathsf{PreHash}(\bar{s}, \mathsf{pwd})$. 算法以 $\mathsf{pwd} \in \mathbb{Z}_q^n$ 和预盐 $\bar{s} := \boldsymbol{s} \in \mathbb{Z}_q^k$ 为输入获得预散列值. 然后生成 $\boldsymbol{b} := (\mathsf{pwd}, \boldsymbol{s})^{\mathrm{T}} \in \mathbb{Z}_q^{n+k}$, 最后计算预散列值

$$\bar{y} := \boldsymbol{p} = \boldsymbol{A} \cdot \boldsymbol{b} \ = \ (\boldsymbol{A}_1, \boldsymbol{A}_2) \cdot \begin{bmatrix} \mathsf{pwd} \\ \bar{s} \end{bmatrix} \ = \ \boldsymbol{A}_1 \cdot \mathsf{pwd} + \boldsymbol{A}_2 \cdot \boldsymbol{s} (\mathrm{mod}\ q) \in \mathbb{Z}_q^m.$$

• $y \leftarrow \mathsf{PHash}(\bar{s}, s, \mathsf{pwd})$. 算法取盐 $s := \boldsymbol{e} \leftarrow \mathcal{D}_\sigma^m$ 和 $\bar{s} := \boldsymbol{s} \leftarrow \mathbb{Z}_q^k$ 获得散列值, 然后计算并输出

$$y \ := \ \boldsymbol{h} = (\boldsymbol{A}_1, \boldsymbol{A}_2) \cdot \begin{bmatrix} \mathsf{pwd} \\ \bar{s} \end{bmatrix} + 2\boldsymbol{e} (\mathrm{mod}\ q) \ = \ \boldsymbol{A}_1 \cdot \mathsf{pwd} + \boldsymbol{A}_2 \cdot \boldsymbol{s} + 2\boldsymbol{e} (\mathrm{mod}\ q) \in \mathbb{Z}_q^m.$$

4.1.3 安全分析

基于 CDGLW 承诺方案的口令散列方案, 很好地继承了承诺方案的隐藏和绑定的属性, 并满足口令散列函数的属性: 口令隐藏性、抗原像性、弱抗碰撞性、预散列墒保持性和墒保持性. 分析如下:

(1) 口令隐藏性. 由于 CDGLW 承诺方案的隐藏特性, 口令散列方案继承了承诺方案的隐藏特性使其具有口令隐藏的特性. 严格地说, 假设在 $\mathbb{Z}_{95^n}$ 中有两个口令 pwd_0 和 pwd_1, 分别映射到整数 π_0 和 π_1, 那么这两个口令可以在承诺方案的帮助下实现完美的隐藏特性. 事实上, 如果敌手 $\mathcal{A}$ 知道消息 π 上的承诺 $y = \boldsymbol{A} \cdot \boldsymbol{b} + \boldsymbol{e} (\mathrm{mod}\ q)$, 那么能够以压倒性的能力区分 π_0 和 π_1 的敌手就意味着承诺方案的隐藏性可以被敌手 $\mathcal{A}$ 打破.

(2) 弱抗碰撞性. 对于任意两个口令 pwd, pwd' 和任意盐值 s, 如果存在 $\mathsf{PHash}(s, \mathsf{pwd}) = \mathsf{PHash}(s, \mathsf{pwd}')$, 则满足完美的弱抗碰撞性. 也就是说, 如果可以通过使用两个不同的预散列值 $\bar{y}$ 和 $\bar{y}'$ (即 $\bar{y}' \neq \bar{y}$) 获得相同的散列值 y, 那么可以使用这些值 (即 $\bar{y}$, $\bar{y}'$ 和 y) 的敌手就可以破坏承诺方案的计算绑定属性. 因此, 可以通过格上 SIS 假设保证弱抗碰撞性.

(3) 抗原像性. 如果找到一个 $\mathsf{PHash}(\cdot)$ 碰撞的概率是可忽略的, 那么抗原像性保持. 显然, 算法 $\mathsf{PHash}(\cdot)$ 的每次调用都需要取随机参数 $\bar{s}$ 和 s, 此外, Cabarcas 等 [51] 承诺方案的隐藏性质已经证明了 $y := \boldsymbol{h}$ 是一个完美的隐藏承诺, 因此, $\mathsf{PreHash}(\cdot)$ 意味着给定一个随机矩阵 $\boldsymbol{A}$、一个口令 pwd 和一个预盐 $\bar{s} := \boldsymbol{s}$, 存在一个 $\mathsf{PreHash}(\cdot)$ 的输出 $\bar{y} := \boldsymbol{A}\boldsymbol{b}$. 如果敌手试图解决 $y := \boldsymbol{A}\boldsymbol{b} = \boldsymbol{h} - 2\boldsymbol{e}$, 那么对敌手 $\mathcal{A}$ 必须对每个候选 $\mathsf{pwd}*$ 执行 2^θ 次 $\boldsymbol{A} \cdot \boldsymbol{b}$ (由 PreHash 表示) 运算. 也就是说, 这大致相当于对 $\mathsf{PreHash}$ 调用了 2^θ 次. 因此, 如果对手能够解决 $y := \boldsymbol{A}\boldsymbol{b} = \boldsymbol{h} - 2\boldsymbol{e}$, 则意味着最终有可能找到与矩阵 $\boldsymbol{A}$ 相关联的 SIS 问题的解决方案.

(4) 预散列熵保持性和熵保持性. 如果预盐是隐藏的, 由于其随机性, $\bar{y}$ 的最小熵大于 pwd 的最小熵.

4.2 基于 BDLOP 承诺的口令散列方案

本小节在 BDLOP 承诺方案的基础上, 提出第 2 种口令散列方案. 首先回顾一下 BDLOP 承诺方案.

4.2.1 BDLOP 承诺方案

Baum 等 [52] 利用零知识证明, 给出了一个格上加法同态承诺方案的构造.

• $\mathrm{pk}_{\mathsf{com}} \leftarrow \mathsf{ComGen}(1^{\lambda})$ 输出一个公共参数 $\mathrm{pk}_{\mathsf{com}}$. 事实上, 这个公共参数 $\mathrm{pk}_{\mathsf{com}}$ 接下来将被用于承诺消息 $\boldsymbol{m} \in \mathbb{R}_q^{\ell}$, 其具体包括

$$\begin{aligned} \boldsymbol{A}_1 &= [\mathbf{I}_n, \boldsymbol{A}_1'] \in \mathbb{R}_q^{n\times k}, \quad \boldsymbol{A}_1 \leftarrow \mathbb{R}_q^{n\times(k-n)}, \\ \boldsymbol{A}_2 &= [\mathbf{0}^{\ell\times n}, \mathbf{I}_{\ell}, \boldsymbol{A}_2'] \in \mathbb{R}_q^{\ell\times k}, \quad \boldsymbol{A}_2' \leftarrow \mathbb{R}_q^{\ell\times(k-n-\ell)}. \end{aligned}$$

输出 $\mathrm{pk}_{\mathsf{com}} = \boldsymbol{A} := \left[\begin{smallmatrix}\boldsymbol{A}_1\\ \boldsymbol{A}_2\end{smallmatrix}\right] \in \mathbb{R}_q^{(n+\ell)\times k}$.

• $\mathsf{com} \leftarrow \mathsf{Commit}(\mathrm{pk}_{\mathsf{com}}, \boldsymbol{m}; \mathsf{d})$ 为了承诺消息 $\boldsymbol{m} \in \mathbb{R}_q^{\ell}$, 选择一个随机向量 $\mathsf{d} := \boldsymbol{r} \in S_{\beta}^{k}$, 这里 S_{β} 是由 $m \in \mathbb{R}$ 的所有元素构成的集合, 其中 $m \in \mathbb{R}$ 的 ℓ_{∞} 范数最多为 β, β 是诚实证明者在 ℓ_{∞} 范数中随机性的范数界. 然后算法计算并输出承诺:

$$\mathsf{Commit}(\mathrm{pk}_{\mathsf{com}}, \boldsymbol{m}; \boldsymbol{r}) := \begin{bmatrix} \mathsf{ct}_1 \\ \mathsf{ct}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{bmatrix} \cdot \boldsymbol{r} + \begin{bmatrix} \mathbf{0}^n \\ \boldsymbol{m} \end{bmatrix} \in \mathbb{R}_q^{(n+\ell)\times 1}.$$

• $\mathsf{Open}(\mathrm{pk}_{\mathsf{com}}, \mathsf{com}, (\boldsymbol{m}, \mathsf{d}))$. 打开算法 Open (验证算法 Vrfy) 输入公钥 $\mathrm{pk}_{\mathsf{com}}$, 消息 $\boldsymbol{m} \in \mathbb{R}_q^{\ell}$, 承诺 $\mathsf{com} = [\mathsf{ct}_1^{\mathrm{T}}, \mathsf{ct}_2^{\mathrm{T}}]^{\mathrm{T}}$ 和一个公开值 (揭露值) $\mathsf{d} := \boldsymbol{r} = [r_1, r_2, \ldots, r_k]^{\mathrm{T}} \in \mathbb{R}_q^{k}$ 以及多项式 $f \in \bar{\mathcal{C}}$. 为了打开承诺, 验证者需要检查以下等式:

$$f \cdot \begin{bmatrix} \mathsf{ct}_1 \\ \mathsf{ct}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{bmatrix} \cdot \boldsymbol{r} + f \cdot \begin{bmatrix} \mathbf{0}^n \\ \boldsymbol{m} \end{bmatrix},$$

对于所有 i, $\|r_i\|_2 \leqslant 4\sigma\sqrt{N}$.

4.2.2 基于 BDLOP 承诺的口令散列方案实例

基于 BDLOP 承诺的口令散列方案的具体构造如下:

• $\bar{s} \leftarrow \mathsf{PreSalt}(\mathsf{params})$ 选择一个随机向量 $\boldsymbol{s} \leftarrow \mathbb{Z}_q^{k}$ 并且设置预盐 $\bar{s} := \boldsymbol{s}$.

• $s \leftarrow \mathsf{Salt}(\mathsf{params})$ 选择一个随机矩阵 $\boldsymbol{B} \in \mathbb{Z}_q^{k\times k}$ 和一个预盐 $\boldsymbol{s} \in \mathbb{Z}_q^{k}$, 然后输出盐 $s := \boldsymbol{e} = \boldsymbol{s}\cdot\boldsymbol{B} \in \mathbb{Z}_q^{k}$.

• $\bar{y} \leftarrow \mathsf{PreHash}(\bar{s}, \mathsf{pwd})$. 算法以 $\mathsf{pwd} \in \mathbb{Z}_q^{\ell}$ 和预盐 $\bar{s} := \boldsymbol{s} \in \mathbb{Z}_q^{k}$ 作为输入获得预散列值, 然后计算并输出

$$\bar{y} := \boldsymbol{p} = \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{bmatrix} \cdot \boldsymbol{s} + \begin{bmatrix} \mathbf{0}^n \\ \mathsf{pwd} \end{bmatrix} (\mathrm{mod}\ q) \in \mathbb{Z}_q^{m}.$$

• $y \leftarrow \mathsf{PHash}(\bar{s}, s, \mathsf{pwd})$. 算法以盐 $s := \boldsymbol{e} \in \mathbb{Z}_q^k$ 和 $\bar{s} := \boldsymbol{s} \leftarrow \mathbb{Z}_q^k$ 作为输入获得散列值, 然后计算并输出

$$y \ := \ \boldsymbol{h} = \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{bmatrix} \cdot \boldsymbol{e} + \begin{bmatrix} \boldsymbol{0}^n \\ \mathsf{pwd} \end{bmatrix} (\mathrm{mod}\ q) \in \mathbb{Z}_q^m.$$

4.2.3 安全性分析

基于 BDLOP 承诺方案的口令散列方案, 很好地继承了 BDLOP 承诺方案的隐藏和绑定的属性, 并满足口令散列函数的属性: 盐值不可区分性、口令隐藏性、抗原像性、弱抗碰撞性、预散列熵保持性和熵保持性. 详细分析类似于基于 CDGLW 的口令散列方案的安全性分析, 故细节省略.

4.3 基于 KTX 承诺的口令散列方案

KTX 承诺方案 [53] 是第一个基于格的承诺方案, 虽然 KTX 承诺方案存在一些缺点, 但不影响口令散列方案的构造.

4.3.1 KTX 承诺方案

KTX 承诺方案 [53] 采用了基于格上 SIS 的假设, 选取素数模 $q = \mathcal{O}(\lambda)$ 和维数 $m = 2\lambda\lceil \log q \rceil$. 为了承诺固定的 $\ell = \mathrm{poly}(\lambda)$ 位, 承诺秘钥是 $\mathrm{pk}_{\mathsf{com}} := \boldsymbol{P} = (\boldsymbol{A}, \boldsymbol{B}) \in \mathbb{Z}_q^{n \times (\ell+m)}$, 其中 $\boldsymbol{A} \leftarrow \mathbb{Z}_q^{n \times \ell}$, $\boldsymbol{B} \leftarrow \mathbb{Z}_q^{n \times m}$.

• $\mathsf{com} \leftarrow \mathsf{Commit}(\mathrm{pk}_{\mathsf{com}}, \boldsymbol{m}; \boldsymbol{r})$. 承诺 ℓ 比特的消息 $\boldsymbol{m} \in \{0,1\}^\ell$, 算法首先选择一个随机数 $\boldsymbol{r} \leftarrow \{0,1\}^m$, 然后计算并输出

$$\mathsf{com} := \mathsf{ct} \leftarrow \mathsf{Commit}(\mathrm{pk}_{\mathsf{com}}, \boldsymbol{m}; \boldsymbol{r}) \ = \ \boldsymbol{P} \cdot \begin{bmatrix} \boldsymbol{m} \\ \boldsymbol{r} \end{bmatrix} \ = \ \boldsymbol{A} \cdot \boldsymbol{m} + \boldsymbol{B} \cdot \boldsymbol{r} (\mathrm{mod}\ q) \in \mathbb{Z}_q^{n \times 1}.$$

• $\mathsf{Open}(\mathrm{pk}_{\mathsf{com}}, \mathsf{com}, (\boldsymbol{m}, \boldsymbol{r}))$. 打开 $\mathsf{com} \in \mathbb{Z}_q^{n \times 1}$, 揭示 $\boldsymbol{m} \in \{0,1\}^\ell$ 和 $\boldsymbol{r} \in \{0,1\}^m$.

如果一个承诺 com 与 $\boldsymbol{r}_1 = \boldsymbol{r}_2$ 可以被两个不同的有效打开串 $(\boldsymbol{m}_1, \boldsymbol{r}_1)$ 和 $(\boldsymbol{m}_2, \boldsymbol{r}_2)$ 打开, 那么这就意味着存在一个敌手可以找到一个与均匀随机分布 $\boldsymbol{P} := [\boldsymbol{A} \| \boldsymbol{B}] \in \mathbb{Z}_q^{n \times (m+\ell)}$ 相关的 $\mathsf{SIS}_{n,(m+\ell),q,q}^{\infty}$ 假设的解. 此外, 剩余散列引理保证了该方案满足统计上的隐藏性质, 这意味着一个有效的承诺 com 在 $\mathbb{Z}_q^n$ 上的分布在统计学上接近一致.

4.3.2 基于 KTX 承诺的口令散列方案实例

基于 KTX 承诺的口令散列方案的具体构造如下:

• $\bar{s} \leftarrow \mathsf{PreSalt}(\mathsf{params})$ 选择一个随机向量 $\boldsymbol{s} \leftarrow \mathbb{Z}_q^k$ 并设置预盐 $\bar{s} := \boldsymbol{s}$.

• $s \leftarrow \mathsf{Salt}(\mathsf{params})$ 选择一个随机向量 $\boldsymbol{r} \in \mathbb{Z}_q^m$ 并将其作为盐 $s := \boldsymbol{r} \in \mathbb{Z}_q^m$.

• $\bar{y} \leftarrow \mathsf{PreHash}(\bar{s}, \mathsf{pwd})$ 算法以随机矩阵 $\boldsymbol{A} \leftarrow \mathbb{Z}_q^{n+(\ell+k)}$, 口令 $\mathsf{pwd} \in \mathbb{Z}_q^\ell$ 和预盐 $\bar{s} := \boldsymbol{s} \in \mathbb{Z}_q^k$ 为输入获得预散列值, 然后计算并输出

$$\bar{y} := \boldsymbol{p} \ = \ \boldsymbol{A} \begin{bmatrix} \boldsymbol{s} \\ \mathsf{pwd} \end{bmatrix} (\mathrm{mod}\ q) \in \mathbb{Z}_{\mathrm{q}}^{\mathrm{n}}.$$

表 1 基于承诺方案的口令散列性能比较

Table 1 Comparison with commitment-based password hashing schemes

Character	Cabarcas et al. [51]	Baum et al. [52]	Kawachi et al. [53]
Assumption	RLWE	RLWE	RLWE
Commitment key	$\boldsymbol{A}=(\boldsymbol{A}_1,\boldsymbol{A}_2)\in\mathbb{Z}_q^{m\times(n+k)}$	$\boldsymbol{A}:=\left[\begin{smallmatrix}\boldsymbol{A}_1\\ \boldsymbol{A}_2\end{smallmatrix}\right]\in\mathbb{R}_q^{(n+\ell)\times k}$	$\boldsymbol{P}=(\boldsymbol{A},\boldsymbol{B})\in\mathbb{Z}_q^{n\times(\ell+m)}$
Plaintext space	$\boldsymbol{m}\in\mathbb{Z}_q^n$	$\boldsymbol{m}\in\mathbb{R}_q^{(n+\ell)\times 1}$	$\boldsymbol{m}\in\{0,1\}^m$
Commitment value	$\boldsymbol{A}_1\cdot\boldsymbol{m}+\boldsymbol{A}_2\cdot\boldsymbol{s}+\boldsymbol{e}\in\mathbb{Z}_q^m$	$\left[\begin{smallmatrix}\boldsymbol{A}_1\\ \boldsymbol{A}_2\end{smallmatrix}\right]\cdot\boldsymbol{r}+\left[\begin{smallmatrix}\boldsymbol{0}^n\\ \boldsymbol{m}\end{smallmatrix}\right]\in\mathbb{R}_q^{(n+\ell)\times 1}$	$\boldsymbol{A}\cdot\boldsymbol{m}+\boldsymbol{B}\cdot\boldsymbol{r}\in\mathbb{Z}_q^{n\times 1}$
Pre-salt value	$\bar{s}:=\boldsymbol{s}\in\mathbb{Z}_q^k$	$\bar{s}:=\boldsymbol{s}\in\mathbb{Z}_q^k$	$\bar{s}:=\boldsymbol{s}\in\mathbb{Z}_q^k$
Salt value	$s:=\boldsymbol{B}\cdot\boldsymbol{e}_0\in\mathbb{Z}_q^m$	$s:=\boldsymbol{e}=\boldsymbol{s}\cdot\boldsymbol{B}\in\mathbb{Z}_q^k$	$s:=\boldsymbol{r}\in\mathbb{Z}_q^m$
PreHash	$\boldsymbol{A}\cdot\left[\begin{smallmatrix}\mathsf{pwd}\\ \boldsymbol{s}\end{smallmatrix}\right]\in\mathbb{Z}_q^m$	$\left[\begin{smallmatrix}\boldsymbol{A}_1\\ \boldsymbol{A}_2\end{smallmatrix}\right]\cdot\boldsymbol{s}+\left[\begin{smallmatrix}\boldsymbol{0}^n\\ \mathsf{pwd}\end{smallmatrix}\right]\in\mathbb{Z}_q^m$	$\boldsymbol{A}\left[\begin{smallmatrix}\boldsymbol{s}\\ \mathsf{pwd}\end{smallmatrix}\right]\in\mathbb{Z}_q^n$
PHash	$\boldsymbol{A}\cdot\left[\begin{smallmatrix}\mathsf{pwd}\\ \boldsymbol{s}\end{smallmatrix}\right]+2\boldsymbol{e}\in\mathbb{Z}_q^m$	$\left[\begin{smallmatrix}\boldsymbol{A}_1\\ \boldsymbol{A}_2\end{smallmatrix}\right]\cdot\boldsymbol{e}+\left[\begin{smallmatrix}\boldsymbol{0}^n\\ \mathsf{pwd}\end{smallmatrix}\right]\in\mathbb{Z}_q^m$	$\boldsymbol{A}\left[\begin{smallmatrix}\boldsymbol{s}\\ \boldsymbol{m}\end{smallmatrix}\right]+\boldsymbol{B}\boldsymbol{r}\in\mathbb{Z}_q^n$

- $y\leftarrow\mathsf{PHash}(\bar{s},s,\mathsf{pwd})$ 算法以预盐 $s:=\boldsymbol{r}\in\mathbb{Z}_q^m$ 和 $\bar{s}:=\boldsymbol{s}\leftarrow\mathbb{Z}_q^k$, $\boldsymbol{A}\leftarrow\mathbb{Z}_q^{n+(\ell+m)}$ 以及 $\boldsymbol{B}\leftarrow\mathbb{Z}_q^{n\times m}$ 为输入获得散列值, 然后计算并输出

$$y\ :=\ \boldsymbol{h}\ =\ (\boldsymbol{A},\boldsymbol{B})\cdot\begin{bmatrix}\boldsymbol{s}\\ \boldsymbol{m}\\ \boldsymbol{r}\end{bmatrix}=\boldsymbol{A}\begin{bmatrix}\boldsymbol{s}\\ \boldsymbol{m}\end{bmatrix}+\boldsymbol{B}\boldsymbol{r}(\mathrm{mod}\ q)\in\mathbb{Z}_q^n.$$

4.3.3 安全性分析

基于 KTX 承诺方案的口令散列方案, 很好地继承了 KTX 承诺方案的隐藏和绑定的属性, 并满足口令散列函数的属性: 盐值不可区分性、口令隐藏性、抗原像性、弱抗碰撞性、预散列熵保持性和熵保持性. 详细分析类似于基于 CDGLW 的口令散列方案的安全性分析, 因此本文省略了细节.

5 性能分析

本文对格上基于承诺的口令散列方案进行了系统性的研究, 包括具体的特性、安全问题和挑战. 如表 1 所示, 本文重点研究了承诺口令的结构和大小、承诺、预盐、盐、预散列值和散列值等主要性质. 分析表明, 3 种口令散列方案都具有相同大小的盐和预盐, 但基于 KTX 的口令散列方案将获得更短的散列和预哈希值 (即 $\mathcal{O}(n)$), 比基于 BDLOP 和基于 CDGLW 的口令散列方案 (即 $\mathcal{O}(m)$) 要好.

6 总结

本文给出了一种在标准模型下基于格的后量子安全口令散列方案的通用构造方法, 与先前方案 [38, 49, 50] 相比, 在性能上并不占优. 但是, 本文所设计的口令散列方案具有可集成到基于 SPHF 的 PAKE 协议中的优势, 从而可获得低交互的口令认证协议, 含注册、认证及密钥交换阶段. 因此, 为获得标准模型下后量子安全且高效的口令散列方案, 本文遵循文献 [20] 所提出的技术路线, 采用基于格的同态承诺方案为基本密码学组件, 给出后量子安全的口令散列方案的通用构造方法, 并设计 3 种不同的基于格的后量子安全口令散列方案, 最后探讨了将其集成到基于 SPHF 的 PAKE 协议中, 从而

获得低交互的口令认证协议的可能性 [54,55]. 实际上, 在该通用构造方法的基础上, 仍存在若干同态承诺方案可有效应用于构造口令散列方案, 该部分将留作下一步的研究.

参考文献

1 Pointcheval D, Zimmer S. Multi-factor authenticated key exchange. In: Proceedings of the 6th International Conference Applied Cryptography and Network Security, New York, 2008. 277–295

2 Gardham D, Manulis M, Dragan C C. Biometric-authenticated searchable encryption. IACR Cryptol ePrint Arch, 2020, 2020: 17

3 Dupont P, Hesse J, Pointcheval D, et al. Fuzzy password-authenticated key exchange. In: Proceedings of the 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, 2018. 393–424

4 Bonneau J, Herley C, van Oorschot P C, et al. Passwords and the evolution of imperfect authentication. Commun ACM, 2015, 58: 78–87

5 Wang P, Wang D, Huang X. Advances in password security (in Chinese). J Comput Res Develop, 2016, 53: 2173

6 Weir M, Aggarwal S, de Medeiros B, et al. Password cracking using probabilistic context-free grammars. In: Proceedings of the 30th IEEE Symposium on Security and Privacy. Oakland: IEEE Computer Society, 2009. 391–405

7 Hitaj B, Gasti P, Ateniese G, et al. PassGAN: a deep learning approach for password guessing. In: Proceedings of the 17th International Conference on Applied Cryptography and Network Security, Bogota, 2019. 217–237

8 Juels A, Rivest R L. Honeywords: making password-cracking detectable. In: Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security, 2013. 145–160

9 Wang D, Cheng H, Wang P, et al. A security analysis of honeywords. In: Proceedings of the Network and Distributed System Security Symposium, 2018. 1–16

10 Wu J. A Non-Technical History of Password Storage. Technical Report, 2017. https://analogist.net/post/password-storage/

11 Grassi P A, Newton E M, Perlner R A, et al. Digital Identity Guidelines: Authentication and Lifecycle Management. Technical Report, NIST 800-63B. 2017

12 Li Z, Wang D. Two-round PAKE protocol over lattices without NIZK. In: Proceedings of the 14th International Conference on Information Security and Cryptology, Fuzhou, 2018. 138–159

13 Hu X X, Zhang J, Zhang Z F, et al. Universally composable anonymous password authenticated key exchange. Sci China Inf Sci, 2017, 60: 052107

14 Li Z, Wang J, Choi C, et al. Multi-factor password-authenticated key exchange via Pythia PRF service. Comput Mater Continua, 2020, 63: 663–674

15 Benhamouda F, Blazy O, Chevalier C, et al. New techniques for SPHFs and efficient one-round PAKE protocols. In: Proceedings of Annual Cryptology Conference, 2013. 449–475

16 Bellovin S M, Merritt M. Encrypted key exchange: password-based protocols secure against dictionary attacks. In: Proceedings of the 1992 IEEE Symposium on Security and Privacy, 1992. 72–84

17 Katz J, Ostrovsky R, Yung M. Efficient password-authenticated key exchange using human-memorable passwords. In: Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques, 2001. 475–494

18 Li Z, Wang D. Achieving one-round password-based authenticated key exchange over lattices. IEEE Trans Serv Comput, 2019. doi: 10.1109/TSC.2019.2939836

19 Jarecki S, Krawczyk H, Xu J Y. OPAQUE: an asymmetric PAKE protocol secure against pre-computation attacks. In: Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2018. 456–486

20 Benhamouda F, Pointcheval D. Verifier-based password-authenticated key exchange: new models and constructions. Cryptology ePrint Archive, Report 2013/833. https://eprint.iacr.org/2013/833

21 Kiefer F, Manulis M. Zero-knowledge password policy checks and verifier-based PAKE. In: Proceedings of European Symposium on Research in Computer Security, 2014. 295–312

22 Kiefer F, Manulis M. Blind password registration for verifier-based PAKE. In: Proceedings of the 3rd ACM Interna-

tional Workshop on ASIA Public-Key Cryptography, 2016. 39–48
23 Pointcheval D, Wang G. VTBPEKE: verifier-based two-basis password exponential key exchange. In: Proceedings of ACM on Asia Conference on Computer & Communications Security, 2017. 301–312
24 Nguyen K, Tan B H M, Wang H. Zero-knowledge password policy check from lattices. In: Proceedings of International Conference on Information Security, 2017. 92–113
25 Bellovin S M, Merritt M. Augmented encrypted key exchange: a password-based protocol secure against dictionary attacks and password file compromise. In: Proceedings of the 1st ACM Conference on Computer and Communications Security, 1993. 244–250
26 Gentry C, MacKenzie P D, Ramzan Z. A method for making password-based key exchange resilient to server compromise. In: Proceedings of the 26th Annual International Conference on Advances in Cryptology, 2006. 142–159
27 Elgamal T. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans Inform Theor, 1985, 31: 469–472
28 Paillier P. Public-key cryptosystems based on composite degree residuosity classes. In: Proceedings of International Conference on the Theory and Application of Cryptographic Techniques, 1999. 223–238
29 Boneh D, Goh E, Nissim K. Evaluating 2-DNF formulas on ciphertexts. In: Proceedings of Theory of Cryptography Conference, 2005. 325–341
30 Fujisaki E, Okamoto T. Statistical zero knowledge protocols to prove modular polynomial relations. In: Proceedings of the 17th Annual International Cryptology Conference, Santa Barbara, 1997. 16–30
31 Damgård I, Fujisaki E. A statistically-hiding integer commitment scheme based on groups with hidden order. In: Proceedings of International Conference on the Theory and Application of Cryptology and Information Security, 2002. 125–142
32 Damgård I, Nielsen J B. Perfect hiding and perfect binding universally composable commitment schemes with constant expansion factor. In: Proceedings of the 22nd Annual International Cryptology Conference on Advances in Cryptology, 2002. 581–596
33 Boyen X, Waters B. Compact group signatures without random oracles. In: Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2006. 427–444
34 Groth J. Simulation-sound NIZK proofs for a practical language and constant size group signatures. In: Proceedings of the 12th International Conference on Theory and Application of Cryptology and Information Security, 2006. 444–459
35 Groth J, Sahai A. Efficient non-interactive proof systems for bilinear groups. In: Proceedings of the 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2008. 415–432
36 Bauman E, Lu Y, Lin Z. Half a century of practice: who is still storing plaintext passwords? In: Proceedings of the 11th International Conference on Information Security Practice and Experience, 2015. 253–267
37 Jaeger D, Pelchen C, Graupner H, et al. Analysis of publicly leaked credentials and the long story of password (re-) use. In: Proceedings of the 11th International Conference on Passwords (PASSWORDS'16), 2016
38 Kaliski B. PKCS #5: password-based cryptography specification version 2.0. Request for Comments: 2898. https://tools.ietf.org/html/rfc2898
39 Aumasson J P. Password Hashing Competition and Our Recommendation for Hashing Passwords: Argon2. Technical Report, 2019. http://www.password-hashing.net/
40 Hatzivasilis G. Password-Hashing status. Cryptography, 2017, 1: 10
41 Leurent G, Nguyen P Q. How risky is the random-oracle model? In: Proceedings of Annual International Cryptology Conference, 2009. 445–464
42 NIST. Recommendation for password-based key derivation. Special Publication 800-132. 2010. http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-13\2.pdf
43 Wang D, Wang P, He D, et al. Birthday, name and bifacial-security: understanding passwords of Chinese web users. In: Proceedings of the 28th USENIX Conference on Security Symposium, 2019. 1537–1555
44 Wang D, Cheng H, Wang P, et al. Zipf's law in passwords. IEEE Trans Inform Forensic Secur, 2017, 12: 2776–2791
45 Bonneau J. The science of guessing: analyzing an anonymized corpus of 70 million passwords. In: Proceedings of IEEE Symposium on Security and Privacy, 2012. 538–552
46 Wang D, Gu Q, Huang X, et al. Understanding human-chosen pins: characteristics, distribution and security. In: Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, 2017. 372–385

47 Shannon C E. A mathematical theory of communication. Mobile Comput Commun Rev, 2001, 5: 3–55
48 Li Z P, Wang J R, Zhang W Y. Revisiting post-quantum hash proof systems over lattices for Internet of Thing authentications. J Ambient Intell Human Comput, 2020, 11: 3337–3347
49 Percival C. A future-adaptable password scheme. The OpenBSD Project, 2009. https://www.tarsnap.com/scrypt/scrypt.pdf
50 Biryukov A, Dinu D, Khovratovich D. Argon2: the memory-hard function for password hashing and other applications. https://www.cryptolux.org/images/0/0d/Argon2.pdf
51 Cabarcas D, Demirel D, Göpfert F, et al. An unconditionally hiding and long-term binding post-quantum commitment scheme. IACR Cryptology ePrint Archive, Report 2013/833
52 Baum C, Damgård I, Lyubashevsky V, et al. More efficient commitments from structured lattice assumptions. In: Proceedings of International Conference on Security and Cryptography for Networks, 2018. 368–385
53 Kawachi A, Tanaka K, Xagawa K. Concurrently secure identification schemes based on the worst-case hardness of lattice problems. In: Proceedings of International Conference on the Theory & Application of Cryptology & Information Security: Advances in Cryptology, 2008. 372–389
54 Li Z P, Wang D, Morais E. Quantum-safe round-optimal password authentication for mobile devices. IEEE Trans Depend Secure Comput, 2020. doi: 10.1109/TDSC.2020.3040776
55 Li Z P, Yang Z, Szalachowsk P, et al. Building low-interactivity multi-factor authenticated key exchange for industrial Internet-of-Things. IEEE Internet Things J, 2020, 8: 844–859

Achieving password-hashing scheme over lattices

Zengpeng LI[1] & Ding WANG[2,3*]

1. *School of Cyber Science and Technology, Shandong University, Qingdao* 266237*, China*;
2. *College of Cyber Science, Nankai University, Tianjin* 300071*, China*;
3. *Tianjin Key Laboratory of Network and Data Security Technology (Nankai University), Tianjin* 300350*, China*
* Corresponding author. E-mail: wangding@nankai.edu.cn

Abstract Password-based authentication is the dominant form of access control and is likely to keep its status in the foreseeable future. Password authenticated key exchange (PAKE) protocols enable two parties to exchange a session key during password-based authentication over an insecure channel. To resist password compromise at the server-side, passwords are recommended to be stored in a salted hash form. However, conventional password hashing functions (e.g., PBKDF2, bcrypt, and scrypt) only support PAKE protocols based on specific number-theoretic assumptions, which can only be proved secure in the random oracle model, and the communication rounds are generally high. Furthermore, they demand a large memory size, i.e., the output is of length 32 bytes. To address these issues, several password hashing schemes based on discrete-logarithm assumptions, e.g., Benhamouda and Pointceva (IACR ePrint2013/833), Kiefer and Manulis (ESORICS'14), and Pointcheval and Wang (ASIACCS'17), have been proposed to be integrated with a smooth projective hash function (SPHF), but they are not secure in the coming quantum era and only can be proved security in the random oracle model. In this work, we focus on the question of how to design an efficient password hashing scheme that can be integrated into quantum-resistant SPHF-based PAKE while being secure in the standard model (but not the random oracle model). Following the research line of Kiefer and Manulis (ESORICS'14), we design three new types of lattice-based password hashing schemes based on homomorphic commitment schemes with provable security in the standard model. We show that they can be efficiently integrated with SPHFs to obtain low-interactive PAKE protocols. Although the proposed scheme is not ready to be deployed in practice, it is an important step for the quantum-resistant password-based authentication and authenticated key exchange.

Keywords quantum resistant, password authenticated key exchange, password hashing scheme, smooth projective hash function, lattice-based cryptography

Zengpeng LI received his Ph.D. degree from Harbin Engineering University (HEU), China, in 2018. During his doctoral program, he was a Ph.D. research assistant with University of Auckland, NZ and Virginia Commonwealth University, USA, respectively, from 2015 to 2017. Currently, he is a faculty member in Shandong University (SDU) Qingdao Campus. His research efforts mainly focus on secure computing on encrypted data, verifiable computation, and password-based cryptography.

Ding WANG received his Ph.D. degree in information security from Peking University, China, in 2017. Currently, he is a full professor at Nankai University, China. His research interests include password, authentication, and provable security.