Multi-hop Fine-Grained Proxy Re-encryption

Yunxiao Zhou[1,2], Shengli Liu[2,3(✉)], and Shuai Han[1,2(✉)]

[1] School of Cyber Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
cloudzhou@sjtu.edu.cn

[2] State Key Laboratory of Cryptology, P.O. Box 5159, Beijing 100878, China
dalen17@sjtu.edu.cn

[3] Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
slliu@sjtu.edu.cn

Abstract. Proxy re-encryption (PRE) allows a proxy to transform a ciphertext intended for Alice (delegator) to another ciphertext intended for Bob (delegatee) without revealing the underlying message. Recently, a new variant of PRE, namely fine-grained PRE (FPRE), was proposed in [Zhou et al., Asiacrypt 2023]. Generally, FPRE is designed for a function family $\mathcal{F}$: each re-encryption key $\mathsf{rk}^{f}_{A\to B}$ is associated with a function $f \in \mathcal{F}$, and with $\mathsf{rk}^{f}_{A\to B}$, a proxy can transform Alice's ciphertext encrypting m to Bob's ciphertext encrypting $f(m)$. However, their scheme only supports single-hop re-encryption and achieves only CPA security.

In this paper, we formalize *multi-hop* FPRE (mFPRE) that supports multi-hop re-encryptions in the fine-grained setting, and propose two mFPRE schemes achieving CPA security and stronger HRA security (security against honest re-encryption attacks), respectively.

- For multi-hop FPRE, we formally define its syntax and formalize a set of security notions including CPA security, HRA security, undirectionality and ciphertext unlinkablity. HRA security is stronger and more reasonable than CPA security, and ciphertext unlinkablity blurs the proxy relations among a chain of multi-hop re-encryptions, hence providing better privacy. We establish the relations between these security notions.
- Our mFPRE schemes support fine-grained re-encryptions for bounded linear functions and have security based on the learning-with-errors (LWE) assumption in the standard model. In particular, one of our schemes is HRA secure and enjoys all the aforementioned desirable securities. To achieve CPA security and HRA security for mFPRE, we extend the framework of [Jafargholi et al., Crypto 2017] and the technique of the [Fuchsbauer et al., PKC 2019].

1 Introduction

Proxy re-encryption (PRE) extends the functionality of public-key encryption with re-encryption capability [4]. Let $(pk^{(A)}, sk^{(A)})$ and $(pk^{(B)}, sk^{(B)})$ be Alice

Q. Tang and V. Teague (Eds.): PKC 2024, LNCS 14604, pp. 161–192, 2024.
https://doi.org/10.1007/978-3-031-57728-4_6

and Bob's public and secret keys, respectively. Then Alice can generate a re-encryption key $\mathsf{rk}_{A\to B}$ with her key pair $(pk^{(A)}, sk^{(A)})$ and Bob's public key $pk^{(B)}$, and issue $\mathsf{rk}_{A\to B}$ to a proxy. Later her proxy is able to transform Alice's ciphertext $ct^{(A)}$ encrypting a message m to Bob's ciphertext $ct^{(B)}$ encrypting the same message, but the proxy cannot learn any information about m from $ct^{(A)}$, $ct^{(B)}$ and $\mathsf{rk}_{A\to B}$. Since its introduction, PRE has found a variety of applications, like email forwarding systems [4], secure distributed file systems [3], digital rights management systems [19] and block chain systems.

If the re-encryption key $\mathsf{rk}_{A\to B}$ can implement ciphertext transform not only from Alice to Bob, but also vice verse, then the PRE scheme is a *bidirectional* one. In contrast, if $\mathsf{rk}_{A\to B}$ does not support ciphertext transformation from Bob to Alice, then the PRE scheme is a *unidirectional* one. Note that the unidirectional property captures a more precise re-encryption authorization than the bidirectional property. Meanwhile, a unidirectional PRE can support bidirectional re-encryption authorization by issuing both $\mathsf{rk}_{A\to B}$ and $\mathsf{rk}_{B\to A}$ to a proxy. Therefore, unidirectional PRE is more welcome. However, designing unidirectional PREs is more challenging than its bidirectional siblings. In this paper, we focus on unidirectional PRE.

After transformation from $ct^{(A)}$ to $ct^{(B)}$ with $\mathsf{rk}_{A\to B}$, if the resulting $ct^{(B)}$ cannot be further transformed, the PRE scheme is a *single-hop* one. Otherwise, the resulting $ct^{(B)}$ can be further transformed to Charlie's ciphertext $ct^{(C)}$ with $\mathsf{rk}_{B\to C}$ (and so on), then the PRE scheme becomes a *multi-hop* one. Multi-hop PRE schemes support ciphertext transformation chains and provide re-encryption services in a more convenient way.

Fine-Grained Proxy Re-encryption. Traditionally, PRE provides an all-or-nothing authorization with which either the receiver can decrypt the transformed ciphertext to obtain the whole message m, or it learns nothing about m. Recently, PRE was further extended to support fine-grained re-encryption authorization in [21], and this variant PRE is named *fine-grained* PRE (FPRE). In an FPRE scheme, the re-encryption key $\mathsf{rk}^{f}_{A\to B}$ is further equipped with a function f which captures the precise re-encryption ability granted to a proxy. With $\mathsf{rk}^{f}_{A\to B}$, the proxy can transform Alice's ciphertext $ct^{(A)}$ encrypting a message m to Bob's ciphertext $ct^{(B)}$ encrypting $f(m)$ under $pk^{(B)}$. The recent work in [21] constructed a single-hop unidirectional FPRE scheme w.r.t. bounded linear functions, and proved its CPA security based on the learning-with-errors (LWE) assumption. However, there are two limitations in the FPRE scheme [21].

- The scheme only supports single-hop re-encryption. Suppose that Alice's ciphertext $ct^{(A)}$ has been transformed to a re-encrypted ciphertext $ct^{(B)}$ for Bob. Now Bob wants to forward the underlying message to Charlie, but he can not ask his proxy to do the ciphertext transformation for him due to the single-hop limitation of the FPRE. Thus, he has to decrypt $ct^{(B)}$ to recover the message and encrypt that message under Charlie's public key by himself. The decrypt-then-encrypt operation imposes extra working load to Bob. With a multi-hop FPRE scheme, this job becomes easy. Bob can simply

forward the ciphertext $ct^{(B)}$ to his proxy and his proxy will be in charge of the ciphertext transformation.

– The scheme only achieves CPA security. In their CPA model, the adversary is not allowed to learn any re-encryptions from the target user to corrupted users. This is not reasonable. Consider such a scenario: Alice has sent a ciphertext $ct^{(A)}$ to her proxy and her proxy has transformed $ct^{(A)}$ to a re-encrypted ciphertext $ct^{(B)}$ for Bob. Now Bob is corrupted by an adversary. Later, Alice receives a new ciphertext $ct^{*(A)}$, and it is natural to require that the adversary learns nothing about the underlying message of $ct^{*(A)}$. However, this desired security cannot be guaranteed by CPA security since in the CPA model, the adversary is not allowed to learn any re-encryptions from the target user Alice to a corrupted user Bob.

  In fact, obtaining re-encryptions from the target user to a corrupted user is the so-called honest re-encryption attacks (HRA) [6]. When taking HRA attacks into account, the CPA security is lifted to HRA security. As demonstrated in [6], HRA security is more reasonable than CPA security.

The above two limitations lead to an interesting question:

Can we construct a multi-hop fine-grained PRE scheme, preferably also achieving HRA security?

Related Works on Multi-hop PRE Schemes. There already exist some unidirectional multi-hop PRE schemes in the literature. Chandran et al. [5] designed the first multi-hop unidirectional PRE scheme from program obfuscation and showed the selective obfuscation-based security of their schemes from the LWE assumption. Phong et al. [17] proposed a multi-hop PRE scheme with selective CPA security. However, their scheme is interactive, i.e., the re-encryption key generation algorithm requires both user i and user j's secret keys. Lai et al. [13] proposed a multi-hop PRE scheme achieving selective CCA security from indistinguishability obfuscation (iO). However, iO is a theoretical tool and far from being practical. Fan et al. [8] presented a latticed-based scheme, achieving selective tag-based CCA (tbCCA) security, but proxy relations (i.e., challenge graph of the adversary) are restricted to tree structure. Note that the tbCCA security and the HRA security are not comparable since tbCCA security model does not capture honest re-encryption attacks. Later, Fuchsbauer et al. [9] improved Chandran et al.'s scheme [5] to HRA security based on LWE. At the same time, they presented another multi-hop unidirectional scheme constructed from fully homomorphic encryption [10] and also achieved HRA security from LWE on the ideal lattices and circular-security assumption. Recently, Miao et al. [15] proposed a generic construction of multi-hop PRE with selective HRA security, and presented instantiations based on the decisional Diffie-Hellman (DDH) assumption.

All the existing multi-hop PRE schemes do not consider the fine-grained re-encryption, so the multi-hop *fine-grained* PRE with HRA security is still missing.

Our Contributions. In this work, we propose the first *multi-hop fine-grained* PRE scheme from LWE in the standard model.

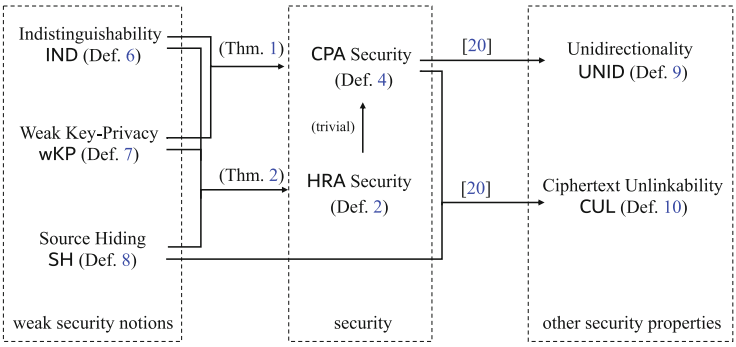

Fig. 1. Security notions of multi-hop FPRE and their relations, where [20] is the full version of this paper.

- *Formal Definitions for Multi-hop Fine-Grained PRE and Its Securities.* We formalize multi-hop fine-grained PRE (mFPRE) that supports multiple re-encryptions in the fine-grained setting. We also present the formal CPA and HRA security notions for multi-hop FPRE. In addition, we define unidirectionality (UNID) and ciphertext unlinkability (CUL) for mFPRE. The CUL security guarantees that the chain of multi-hop re-encryptions does not leak information about proxy relations among them, and hence provide better privacy. Moreover, we prove that UNID is implied by CPA, and CUL is implied by CPA and a weak security notion named source-hiding (SH).
- *Generic Framework for Achieving CPA and HRA Security for Multi-hop FPRE.* We extend the framework in [12] and adapt the techniques in [9] to the multi-hop FPRE setting for achieving (adaptive) CPA and HRA security. More precisely, we first define three weaker security notions including indistinguishability (IND), weak key-privacy (wKP) and source-hiding (SH). Then, we show that the CPA security of multi-hop FPRE is implied by IND and wKP, and the HRA security is implied by IND, wKP and SH. For proxy relations being chains or trees, our reduction only loses a quasi-polynomial factor. Note that the chain and tree topology have good applications in encrypted cloud storage, encrypted email forwarding, etc., as noted by [9].
- *Construction of Multi-hop FPRE from LWE.* We propose two unidirectional multi-hop FPRE schemes, including a CPA secure mFPRE_1 and an HRA secure mFPRE_2, for bounded linear functions[1]. More precisely, we prove that our first scheme mFPRE_1 has IND and wKP securities and hence achieves CPA

[1] Here "bounded" mean that the coefficients are of bounded norm. We note that the existing (single-hop) FPRE schemes [21] are also w.r.t. bounded linear functions.

security and UNID security, and prove that our second scheme mFPRE_2 has IND, wKP and SH securities and hence achieves HRA security, UNID security and CUL security. Both of the schemes are based on the LWE assumption in the standard model.

We refer to Fig. 1 for an overview of the security notions for multi-hop FPRE and their relations established in this work, and refer to Table 1 for a comparison of our schemes with known multi-hop unidirectional PRE schemes.

Table 1. Comparison of multi-hop unidirectional PRE schemes. The column **Standard Model?** asks whether the security is proved in the standard model. The column **Adaptive Corruptions?** asks whether all the security notions support adaptive corruptions. The column **Security** shows the type of security that the scheme achieves, where "HRA" refers to security against honest re-encryption attacks [6], and "tbCCA" refers to tag-based CCA [8] which is incomparable with HRA and restricts the proxy relations (i.e., challenge graph) to tree structure. The column **UNID** shows whether the scheme has unidirectionality. The column **CUL** shows whether the scheme has ciphertext unlinkability. The column **Assumption** shows the assumptions that the security of the scheme is based on, where "iO" refers to indistinguishability obfuscation. The column **Post Quantum?** asks whether the scheme is based on a post-quantum assumption. The column **Fine-Grained?** asks whether the scheme supports fine-grained re-encryptions. The column **Maximum Hops** shows the maximum re-encryption hops that the scheme supports, where "poly-log" refers to $\mathsf{poly}(\log \lambda)$, "sub-linear" refers to λ^{ε} with $0 < \varepsilon < 1$ in the security parameter λ, and "unbounded*" means that the PRE scheme in [15] can support any number of re-encryptions, but at the cost that the ciphertext length grows linearly with the number of re-encryptions. "–" means that no proof or discussion is provided.

PRE Scheme	Standard Model?	Adaptive Corruptions?	Security	UNID	CUL	Assumption	Post Quantum?	Fine–Grained?	Maximum Hops
FL19 [8]	✓	×	tbCCA	✓	–	LWE	✓	–	poly-log
LHAM20 [14]	✓	×	CCA	✓	–	iO	×	–	–
MPW23 [15]	✓	×	HRA	✓	–	DDH	×	–	unbounded*
FKKP19 [9]+ CCLNX14 [5]	✓	✓	HRA	✓	✓	LWE	✓	–	sub-linear
FKKP19 [9] +Gen09 [10]	✓	✓	HRA	✓	✓	LWE over ideal lattice + circular security	✓	–	–
mFPRE_1	✓	✓	CPA	✓	–	LWE	✓	✓	sub-linear
mFPRE_2	✓	✓	HRA	✓	✓	LWE	✓	✓	sub-linear

Technical Overview. Below we give a high-level overview of our multi-hop fine-grained PRE (mFPRE) scheme. We will first review the single-hop FPRE scheme proposed in [21]. Then we will explain how we realize multi-hop re-encryptions and how we achieve HRA security. For simplicity, we do not specify the dimensions of matrices/vectors.

Recap: The Single-Hop FPRE Scheme in [21] and Its Limitations. We give a brief description of the single-hop scheme in [21]. For user i, its public key $pk^{(i)}$ consists of two matrices $\mathbf{A}_1^{(i)} = \begin{pmatrix} \overline{\mathbf{A}}_1^{(i)} \\ \underline{\mathbf{A}}_1^{(i)} \end{pmatrix}$ and $\mathbf{A}_2^{(i)} = \begin{pmatrix} \overline{\mathbf{A}} \\ \underline{\mathbf{A}}_2^{(i)} \end{pmatrix}$, and its

secret key $sk^{(i)}$ contains a trapdoor $\mathbf{T}^{(i)}$ of $\overline{\mathbf{A}}_1^{(i)}$.[2] Here the upper part of $\mathbf{A}_2^{(i)}$ is a (fixed) matrix $\overline{\mathbf{A}}$ generated by a trusted setup and shared by all users, as required by the security of the scheme [21].

The ciphertexts of their scheme have two levels. The first-level/second-level ciphertext $ct_1^{(i)}/ct_2^{(i)}$ of user i is generated using $\mathbf{A}_1^{(i)}/\mathbf{A}_2^{(i)}$ in $pk^{(i)}$ according to the dual Regev encryption scheme [18], namely for level $b \in \{1, 2\}$,

$$ct_b^{(i)} = \mathbf{A}_b^{(i)}\mathbf{s} + \mathbf{e} + \begin{pmatrix} \mathbf{0} \\ \lfloor q/2 \rfloor \mathbf{m} \end{pmatrix} = \begin{pmatrix} \overline{\mathbf{A}}_b^{(i)}\mathbf{s} + \overline{\mathbf{e}} \\ \underline{\mathbf{A}}_b^{(i)}\mathbf{s} + \underline{\mathbf{e}} + \lfloor q/2 \rfloor \cdot \mathbf{m} \end{pmatrix}, \tag{1}$$

where $\mathbf{s}$ and $\mathbf{e} = \begin{pmatrix} \overline{\mathbf{e}} \\ \underline{\mathbf{e}} \end{pmatrix}$ are sampled according to a noise distribution χ.

To realize fine-grained re-encryptions w.r.t. a linear function $f_{\mathbf{M}} : \mathbf{m} \mapsto \mathbf{M} \cdot \mathbf{m}$, the re-encryption key is defined as $\mathsf{rk}_{i \to j}^{f_{\mathbf{M}}} := \left(\mathbf{R} \;\middle|\; \begin{smallmatrix} \mathbf{0} \\ \mathbf{M} \end{smallmatrix}\right)$, where $\mathbf{R}$ is a small norm matrix satisfying

$$\mathbf{R}\overline{\mathbf{A}}_1^{(i)} = \mathbf{A}_2^{(j)}\mathbf{S} + \mathbf{E} - \begin{pmatrix} \mathbf{0} \\ \mathbf{M} \end{pmatrix}\underline{\mathbf{A}}_1^{(i)} \tag{2}$$

with matrices $\mathbf{S}, \mathbf{E}$ following the noise distribution χ. Such $\mathbf{R}$ can be efficiently found by using the pre-image sampling algorithm $\mathsf{SamplePre}$ in [11] with the help of the trapdoor $\mathbf{T}^{(i)}$ of $\overline{\mathbf{A}}_1^{(i)}$ contained in $sk^{(i)}$ (cf. see Footnote 2). Now with $\mathsf{rk}_{i \to j}^{f_{\mathbf{M}}}$, user i's first-level ciphertext $ct_1^{(i)}$ of $\mathbf{m}$ can be converted to user j's second-level ciphertext $ct_2^{(j)}$ of the linear function $\mathbf{M} \cdot \mathbf{m}$ via multiplication

$$\begin{aligned} ct_2^{(j)} := \mathsf{rk}_{i \to j}^{f_{\mathbf{M}}} \cdot ct_1^{(i)} &= \left(\mathbf{R} \;\middle|\; \begin{smallmatrix} \mathbf{0} \\ \mathbf{M} \end{smallmatrix}\right) \cdot \begin{pmatrix} \overline{\mathbf{A}}_1^{(i)}\mathbf{s} + \overline{\mathbf{e}} \\ \underline{\mathbf{A}}_1^{(i)}\mathbf{s} + \underline{\mathbf{e}} + \lfloor q/2 \rfloor \cdot \mathbf{m} \end{pmatrix} \\ &= \underbrace{\left(\mathbf{R}\overline{\mathbf{A}}_1^{(i)} + \begin{pmatrix} \mathbf{0} \\ \mathbf{M} \end{pmatrix}\underline{\mathbf{A}}_1^{(i)}\right)}_{=\mathbf{A}_2^{(j)}\mathbf{S}+\mathbf{E} \text{ by } (2)} \cdot \mathbf{s} + \mathbf{R}\overline{\mathbf{e}} + \begin{pmatrix} \mathbf{0} \\ \mathbf{M}\underline{\mathbf{e}} \end{pmatrix} + \begin{pmatrix} \mathbf{0} \\ \lfloor q/2 \rfloor \cdot \mathbf{M}\mathbf{m} \end{pmatrix} \\ &= \mathbf{A}_2^{(j)} \underbrace{\mathbf{S}\mathbf{s}}_{:=\mathbf{s}'} + \underbrace{\mathbf{E}\mathbf{s} + \mathbf{R}\overline{\mathbf{e}} + \begin{pmatrix} \mathbf{0} \\ \mathbf{M}\underline{\mathbf{e}} \end{pmatrix}}_{:=\mathbf{e}'} + \begin{pmatrix} \mathbf{0} \\ \lfloor q/2 \rfloor \cdot \mathbf{M}\mathbf{m} \end{pmatrix}. \end{aligned} \tag{3}$$

Though a first-level ciphertext $ct_1^{(i)}$ can be re-encrypted to a second-level ciphertext $ct_2^{(j)}$, a second-level ciphertext $ct_2^{(j)}$ cannot be re-encrypted furthermore (no matter to first- or second-level ciphertexts), as explained below.

– To enable further re-encryptions of $ct_2^{(j)}$ to another user (say user k), user j need to compute a re-encryption key $\mathsf{rk}_{j \to k}^{f_{\mathbf{M}'}}$ similar to (2), and in particular, user j need to compute a small-norm $\mathbf{R}$ satisfying

$$\mathbf{R}\overline{\mathbf{A}} = \mathbf{A}_b^{(k)}\mathbf{S} + \mathbf{E} - \begin{pmatrix} \mathbf{0} \\ \mathbf{M}' \end{pmatrix}\underline{\mathbf{A}}_2^{(j)} \quad \text{for some } b \in \{1, 2\}, \tag{4}$$

where $\overline{\mathbf{A}}$ is the upper part of $\mathbf{A}_2^{(j)}$.

[2] With the trapdoor $\mathbf{T}^{(i)}$ of $\overline{\mathbf{A}}_1^{(i)}$, one can use the pre-image sampling algorithm $\mathsf{SamplePre}$ developed in [11] to sample a small-norm $\mathbf{R}$ such that $\mathbf{R} \cdot \overline{\mathbf{A}}_1^{(i)} = \mathbf{B}$ holds, given any $\mathbf{B}$.

– Note that $\overline{\mathbf{A}}$ is chosen by a trusted setup, so user j has no trapdoor of $\overline{\mathbf{A}}$. This is crucial to the security of their single-hop scheme [21], since their security proof needs to embed an LWE instance to $\overline{\mathbf{A}}$. But without knowing a trapdoor of $\overline{\mathbf{A}}$, user j *cannot* generate a $\mathbf{R}$ satisfying (4).[3]

Overall, it is the security that limits the scheme in [21] serving only for *single-hop* re-encryptions.

Achieving Multi-hop Re-encryptions. Note that in the single-hop scheme [21], the ciphertexts $ct_1^{(i)}, ct_2^{(i)}$ of two levels have an almost identical form (i.e., the dual Regev encryption) except for the matrix ($\mathbf{A}_1^{(i)}$ or $\mathbf{A}_2^{(i)}$) used in the encryption. The first-level ciphertext $ct_1^{(i)}$ can be re-encrypted since user i has the trapdoor of $\overline{\mathbf{A}}_1^{(i)}$, while the second-level ciphertext $ct_2^{(i)}$ cannot since user i does not have the trapdoor of $\overline{\mathbf{A}}$.

To enable multi-hop re-encryptions, the public key $pk^{(i)}$ in our scheme contains only one matrix $\mathbf{A}^{(i)} = \binom{\overline{\mathbf{A}}^{(i)}}{\underline{\mathbf{A}}^{(i)}}$, and the secret key $sk^{(i)}$ is the trapdoor $\mathbf{T}^{(i)}$ of $\overline{\mathbf{A}}^{(i)}$. (So our scheme has a transparent setup in contrast to [21].) The ciphertexts $ct^{(i)}$ in our scheme stick to $\mathbf{A}^{(i)}$ during encryption, i.e.,

$$ct^{(i)} = \mathbf{A}^{(i)}\mathbf{s} + \mathbf{e} + \begin{pmatrix} \mathbf{0} \\ \lfloor q/2 \rfloor \mathbf{m} \end{pmatrix}.$$

The re-encryption key $\mathsf{rk}_{i\to j}^{f_{\mathbf{M}}} := \left(\mathbf{R} \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}\end{smallmatrix}\right)$ in our scheme generates the small norm $\mathbf{R}$ according to

$$\mathbf{R}\overline{\mathbf{A}}^{(i)} = \mathbf{A}^{(j)}\mathbf{S} + \mathbf{E} - \binom{\mathbf{0}}{\mathbf{M}}\underline{\mathbf{A}}^{(i)}.$$

In a nutshell, we discard the subscripts $1, 2$ in our scheme.

Similar to the analysis (3), in our scheme, user i's ciphertext $ct^{(i)} = \mathbf{A}^{(i)}\mathbf{s} + \mathbf{e} + \binom{\mathbf{0}}{\lfloor q/2 \rfloor \cdot \mathbf{m}}$ of message $\mathbf{m}$ can be translated to user j's ciphertext with

$$ct^{(j)} := \mathsf{rk}_{i\to j}^{f_{\mathbf{M}}} \cdot ct^{(i)} = \mathbf{A}^{(j)} \underbrace{\mathbf{S}\mathbf{s}}_{:=\mathbf{s}'} + \underbrace{\mathbf{E}\mathbf{s} + \mathbf{R}\overline{\mathbf{e}} + \binom{\mathbf{0}}{\mathbf{M}\underline{\mathbf{e}}}}_{:=\mathbf{e}'} + \Big(\lfloor q/2 \rfloor \cdot \underbrace{\begin{smallmatrix}\mathbf{0}\\ \mathbf{M}\cdot\mathbf{m}\end{smallmatrix}}_{=f_{\mathbf{M}}(\mathbf{m})}\Big). \tag{5}$$

Now in our scheme, user j owns the trapdoor $\mathbf{T}^{(j)}$ of $\overline{\mathbf{A}}^{(j)}$ in its secret key, so it is able to generate $\mathsf{rk}_{j\to k}^{f_{\mathbf{M}'}} := \left(\mathbf{R}' \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}'\end{smallmatrix}\right)$ by sampling a small norm $\mathbf{R}'$ satisfying

$$\mathbf{R}'\overline{\mathbf{A}}^{(j)} = \mathbf{A}^{(k)}\mathbf{S}' + \mathbf{E}' - \binom{\mathbf{0}}{\mathbf{M}'}\underline{\mathbf{A}}^{(j)}.$$

[3] Otherwise, assuming that user j can generate a $\mathbf{R}$ satisfying (4) without knowing a trapdoor of $\overline{\mathbf{A}}$, then anyone (including user k) can generate such $\mathbf{R}$ and thus $\mathsf{rk}_{j\to k}^{f_{\mathbf{M}'}}$ without the help of user j. In this case, user k can translate all ciphertexts $ct_2^{(j)}$ intended for j to ciphertexts $ct_b^{(k)}$ ($b \in \{1, 2\}$) encrypted under $pk^{(k)}$ by itself, and then decrypt the re-encrypted ciphertexts using $sk^{(k)}$ to learn information about the message underlying $ct_2^{(j)}$, violating the confidentiality of encryption scheme.

Consequently, with $\mathsf{rk}_{j\to k}^{f_{\mathbf{M}'}}$, the re-encryption $ct^{(j)} = \mathbf{A}^{(j)}\mathbf{s}' + \mathbf{e}' + \left(\begin{smallmatrix}\mathbf{0}\\ \lfloor q/2\rfloor\cdot\mathbf{M}\cdot\mathbf{m}\end{smallmatrix}\right)$ generated by (5) can be further re-encrypted to user k's ciphertext

$$ct^{(k)} := \mathsf{rk}_{j\to k}^{f_{\mathbf{M}'}} \cdot ct^{(j)} = \mathbf{A}^{(k)} \underbrace{\mathbf{S}'\mathbf{s}'}_{:=\mathbf{s}''} + \underbrace{\mathbf{E}'\mathbf{s}' + \mathbf{R}'\overline{\mathbf{e}'} + \left(\begin{smallmatrix}\mathbf{0}\\ \mathbf{M}'\underline{\mathbf{e}'}\end{smallmatrix}\right)}_{:=\mathbf{e}''} + \Big(\lfloor q/2\rfloor\cdot \underbrace{\begin{smallmatrix}\mathbf{0}\\ \mathbf{M}'\cdot(\mathbf{M}\mathbf{m})\end{smallmatrix}}_{=f_{\mathbf{M}'}(f_{\mathbf{M}}(\mathbf{m}))}\Big),$$

which encrypts $f_{\mathbf{M}'}(f_{\mathbf{M}}(\mathbf{m})) := \mathbf{M}'\cdot\mathbf{M}\cdot\mathbf{m}$. In this way, the re-encryptions can be further extended with $ct^{(i)} \xrightarrow{\mathsf{rk}_{i\to j}^{f_{\mathbf{M}}}} ct^{(j)} \xrightarrow{\mathsf{rk}_{j\to k}^{f_{\mathbf{M}'}}} ct^{(k)} \xrightarrow{\mathsf{rk}_{k\to w}^{f_{\mathbf{M}''}}} \cdots$, and thus we achieve *multi-hop* fine-grained PRE for linear functions. Note that the norm of the errors $\mathbf{e}, \mathbf{e}', \mathbf{e}'', \cdots$ increases as the re-encryption continues, so to guarantee the correctness of decryption, the re-encryption can go on until the norm of errors reaches $\lfloor q/4\rfloor$. In fact, our multi-hop FPRE scheme supports constant hops of re-encryptions under polynomial modulus q and supports sub-linear hops of re-encryptions under sub-exponential modulus q.

Overall, since user j has the trapdoor $\mathbf{T}^{(j)}$ of $\overline{\mathbf{A}}^{(j)}$ in our scheme, this rescues our scheme from single-hop, but at the same time, it incurs an issue: we cannot embed an LWE instance to $\overline{\mathbf{A}}^{(j)}$ in the security proof. To avoid this issue, the scheme in [21] prohibits user j from having the trapdoor of both matrices in public key, which in turn limits it to supporting only single-hop re-encryption. To address this issue, we need new techniques to prove security for our mFPRE.

Below we will first show the high-level ideas of the selective CPA security proof of our scheme, and then explain how we upgrade the selective security to adaptive security by adapting the framework of [9,12] to the fine-grained setting, and explain how we achieve the stronger HRA security.

Selective CPA Security of Our Scheme. We give a high-level overview of the selective CPA security proof of our scheme. Roughly speaking, the (adaptive) CPA security asks the hardness of determining whether a ciphertext ct^* under $pk^{(i^*)}$ encrypts $\mathbf{m}_0$ or $\mathbf{m}_1$, even if an adversary $\mathcal{A}$ can get re-encryption keys $\{\mathsf{rk}_{i\to j}^{f}\}$ and secret keys $\{sk^{(i)}\}$ of some users. To prevent trivial attacks, $\mathcal{A}$ cannot corrupt the target user i^*, and cannot obtain a chain of re-encryption keys from i^* to some corrupted user j. Selective CPA security is weaker as it requires $\mathcal{A}$ to declare the target user i^* and the tuples (i, j) for which $\mathcal{A}$ wants to obtain the corresponding $\{\mathsf{rk}_{i\to j}^{f}\}$ at the beginning of the game.

The main ideas for the selective CPA security proof are: we first change the generations of re-encryption keys $\{\mathsf{rk}_{i\to j}^{f}\}$ so that it does not involve $sk^{(i^*)}$, and then the indistinguishability of ct^* essentially follows from the CPA security of the dual Regev encryption scheme (based on LWE). More precisely,

- **Step 1. Simulating the generation of $\{\mathsf{rk}_{i\to j}^{f}\}$ without knowing $sk^{(i^*)}$.** Let us take an (acyclic) chain of re-encryption keys $\mathsf{rk}_{i^*\to j_1}^{f_1}, \mathsf{rk}_{j_1\to j_2}^{f_2}, \cdots, \mathsf{rk}_{j_{d-1}\to j_d}^{f_{d-1}}$ as example to show how we simulate them in a computationally indistinguishable way without using $sk^{(i^*)}$.

Observe that only the generation of $\mathsf{rk}_{i^* \to j_1}^{f_1}$ involves $sk^{(i^*)}$, where the trapdoor $sk^{(i^*)} = \mathbf{T}^{(i^*)}$ of $\overline{\mathbf{A}}^{(i^*)}$ is used to sample $\mathbf{R}$ satisfying

$$\mathbf{R}\overline{\mathbf{A}}^{(i^*)} = \mathbf{A}^{(j_1)}\mathbf{S} + \mathbf{E} - \binom{\mathbf{0}}{\mathbf{M}}\underline{\mathbf{A}}^{(i^*)}.$$

Thus we need an indistinguishable way to sample it without trapdoor $\mathbf{T}^{(i^*)}$.

If we can embed an LWE instance to $\mathbf{A}^{(j_1)}\mathbf{S} + \mathbf{E}$ in the above equation, then it can be replaced by a uniform $\mathbf{U}$, and consequently, we have

$$\mathbf{R}\overline{\mathbf{A}}^{(i^*)} = \mathbf{A}^{(j_1)}\mathbf{S} + \mathbf{E} - \binom{\mathbf{0}}{\mathbf{M}}\underline{\mathbf{A}}^{(i^*)} \overset{c}{\approx} \mathbf{U} - \binom{\mathbf{0}}{\mathbf{M}}\underline{\mathbf{A}}^{(i^*)} \equiv \mathbf{U}.$$

As a result, we are able to sample $\mathbf{R}$ such that $\mathbf{R}\overline{\mathbf{A}}^{(i^*)} \equiv \mathbf{U}$ by simply choosing it according to a proper discrete Gaussian distribution.[4] However, we cannot embed the LWE instance, since the trapdoor of $\mathbf{A}^{(j_1)}$ is needed to generate $\mathsf{rk}_{j_1 \to j_2}^{f_2}$. This is exactly the issue we mentioned before.

To solve the problem without sacrificing the capability of multi-hop re-encryptions, we simulate the chain of re-encryption keys in reverse order. We will first change the generation of the very last $\mathsf{rk}_{j_{d-1} \to j_d}^{f_{d-1}}$ in the chain as follows. Since $\mathsf{rk}_{j_{d-1} \to j_d}^{f_{d-1}}$ lies in the very end of the chain, we do not need to generate re-encryption key from user j_d to any other users. Moreover, this chain starting from i^* contains only uncorrupted users to avoid trivial attacks. Consequently, the secret key $sk^{(j_d)}$ of user j_d is in fact not needed in the experiment, and now we can embed an LWE instance to $\mathbf{A}^{(j_d)}\mathbf{S} + \mathbf{E}$ such that

$$\mathbf{R}\overline{\mathbf{A}}^{(j_{d-1})} = \mathbf{A}^{(j_d)}\mathbf{S} + \mathbf{E} - \binom{\mathbf{0}}{\mathbf{M}}\underline{\mathbf{A}}^{(j_{d-1})} \overset{c}{\approx} \mathbf{U} - \binom{\mathbf{0}}{\mathbf{M}}\underline{\mathbf{A}}^{(j_{d-1})} \equiv \mathbf{U}.$$

Then $\mathbf{R}$ can be simply sampled following the proper discrete Gaussian distribution so that $\mathbf{R}\overline{\mathbf{A}}^{(j_{d-1})} \equiv \mathbf{U}$.

After the changing of $\mathsf{rk}_{j_{d-1} \to j_d}^{f_{d-1}}$, the secret key $sk^{(j_{d-1})}$ of user j_{d-1} is no longer involved, and thus through a similar analysis, we can then embed an LWE instance to $\mathbf{A}^{(j_{d-1})}\mathbf{S} + \mathbf{E}$ so that the $\mathbf{R}$ in the second last $\mathsf{rk}_{j_{d-2} \to j_{d-1}}^{f_{d-2}}$ can be sampled following discrete Gaussian. By changing the re-encryption keys one by one, we can eventually simulate all re-encryption keys in the chain by simply sampling them according to discrete Gaussian, without $sk^{(i^*)}$.

More generally, the re-encryption keys $\{\mathsf{rk}_{i \to j}^{f}\}$ queried by $\mathcal{A}$ might not be a chain. Nevertheless, we can simulate them in a similar way, roughly by processing all the chains simultaneously and for each chain in reverse order.

- **Step 2. Computationally hiding $\mathbf{m}_0/\mathbf{m}_1$ in $ct^{(i^*)}$.** After Step 1, $sk^{(i^*)}$ is not used at all, and thus for the challenge ciphertext $ct^{(i^*)} = \mathbf{A}^{(i^*)}\mathbf{s} + \mathbf{e} + \binom{\mathbf{0}}{\lfloor q/2 \rfloor \mathbf{m}_\beta}$ ($\beta \in \{0, 1\}$), we can embed an LWE instance to $\mathbf{A}^{(i^*)}\mathbf{s} + \mathbf{e}$, so that the underlying message $\mathbf{m}_\beta$ is hidden to the adversary $\mathcal{A}$.

[4] By [11], if $\mathbf{R}$ follows a proper discrete Gaussian distribution, then $\mathbf{R}\overline{\mathbf{A}}^{(i^*)}$ is statistically close to the uniform distribution $\mathbf{U}$.

Overall, this proof strategy works only in the selective setting, as it requires to know the tuples (i, j) for which $\mathcal{A}$ wants to obtain $\{\mathsf{rk}_{i\to j}^{f}\}$ in advance, so that they can be properly simulated (i.e., in reverse order for each chain).

To achieve adaptive security, if we guess the tuples (i, j) that $\mathcal{A}$ wants to query at the beginning of game, it will incur a security loss as large as $O(2^{\mathfrak{n}^2})$ with $\mathfrak{n}$ the number of users. To reduce the security loss of adaptive security, we extend the frameworks in [9,12] to multi-hop FPRE, as explained below.

Achieving Adaptive Security with Jafargholi et al.'s Framework. Jafargholi et al. [12] proposed a generic framework for upgrading selective security to adaptive security with a more fine-grained analysis. Later, Fuchsbauer et al. [9] applied the framework of [12] to the security of (traditional) PRE. In this work, we extend the framework of Jafargholi et al. [12] and the techniques of Fuchsbauer et al. [9] to our multi-hop FPRE.

Roughly speaking, the main observations are: although in the above selective proof strategy, we need the whole information (denoted by w) about the tuples (i, j) that $\mathcal{A}$ wants to query for re-encryption keys, only part of the information (denoted by u) is used in simulating the intermediate hybrids. For example, in the proof strategy shown above, Step 1 consists of many hybrids, while in each hybrid we only change the generation of a single re-encryption key in the chain, so a small amount of information u will be sufficient for the reduction to the LWE assumption; in Step 2, the information of $u := i^*$ is sufficient for the reduction. It is shown in [12] that the security loss in such cases can be limited to the maximum size of the information u used across any two successive hybrids, which might be much smaller than the size of w.

To apply their techniques [9,12], we abstract two useful yet weaker security notions for our multi-hop FPRE, including indistinguishability (IND) and weak key-privacy (wKP), and then establish a theorem by reducing the adaptive CPA security to IND and wKP with a smaller security loss. Concretely, the two weaker notions exactly correspond to Step 1 and Step 2 in the above proof strategy.

Weak Key-Privacy (wKP). It stipulates that the re-encryption key $\mathsf{rk}_{i\to j}^{f}$ honestly generated by $sk^{(i)}$ can be indistinguishably changed to a simulated one generated without $sk^{(i)}$ in the view of adversary who gets no secret keys $sk^{(i)}$.

Indistinguishability (IND). It requires the indistinguishability of ciphertext for adversary who gets no re-encryption keys $\mathsf{rk}_{i\to j}^{f}$ and no secret keys $sk^{(i)}$.

The theorem showing adaptive CPA security based on IND and wKP for our multi-hop FPRE is proved in a similar way as [9,12]. For an arbitrary adversary who can obtain re-encryption keys $\{\mathsf{rk}_{i\to j}^{f}\}$ for arbitrary tuples (i, j), the security loss of adaptive CPA security is $\mathfrak{n}^{O(\mathfrak{n})}$ in contrast to the naive guessing strategy $O(2^{\mathfrak{n}^2})$. In many realistic scenarios like key rotation for encrypted cloud storage or forwarding of encrypted mail, as demonstrated in [9], the proxy relations are in fact *trees, chains or low-depth graphs*. In these situations, an adversary can only obtain $\{\mathsf{rk}_{i\to j}^{f}\}$ for tuples (i, j) that form trees, chains or low-depth graphs, and the security loss is only quasi-polynomial $\mathfrak{n}^{O(\log \mathfrak{n})}$.

Achieving HRA Security. Security against honest re-encryption attacks (HRA) was first introduced by Cohen [6] and is a security notion stronger and more reasonable than CPA. Compared with CPA security, HRA also allows the adversary $\mathcal{A}$ to obtain re-encryptions of ciphertexts from the target user i^* to *corrupted* users, as long as the ciphertexts to be re-encrypted are honestly generated and are not (re-encryptions of) the challenge ciphertext ct^*. Note that HRA security is stronger than CPA: in the CPA experiment, $\mathcal{A}$ cannot obtain a chain of re-encryption keys from i^* to corrupted users in order to prevent trivial attacks, and thus cannot generate re-encryptions from i^* to corrupted users by itself.

In order to achieve HRA security, we need to enhance our aforementioned CPA proof strategy with a new computationally indistinguishable method for simulating the generation of re-encryptions of ciphertexts from the target user i^* to corrupted users without using $sk^{(i^*)}$. Note that the re-encryptions from i^* to corrupted users might be a chain $ct^{(i^*)} \to ct^{(j_1)} \to ct^{(j_2)} \to \cdots \to ct^{(j_d)}$, the generation of which involves a chain of re-encryption keys $\mathsf{rk}^{f_1}_{i^* \to j_1}, \mathsf{rk}^{f_2}_{j_1 \to j_2}, \cdots, \mathsf{rk}^{f_{d-1}}_{j_{d-1} \to j_d}$. However, we cannot use similar techniques as the CPA security proof strategy to replace this chain of re-encryption keys with simulated ones, since the involved users $j_1, j_2, \cdots, j_d$ might be corrupted by $\mathcal{A}$.

To bypass this problem, we will simulate the generation of the chain of re-encryptions $ct^{(i^*)} \to ct^{(j_1)} \to ct^{(j_2)} \to \cdots \to ct^{(j_d)}$ directly, without using any of the re-encryption keys $\mathsf{rk}^{f_1}_{i^* \to j_1}, \mathsf{rk}^{f_2}_{j_1 \to j_2}, \cdots, \mathsf{rk}^{f_{d-1}}_{j_{d-1} \to j_d}$, thus also without using $sk^{(i^*)}$. To this end, we abstract a (weak) security notion called source-hiding (SH) for multi-hop FPRE, by adapting the techniques in [9,12].

Source-Hiding (SH). It stipulates that the honestly generated re-encryption $ct^{(i)} \to ct^{(j)}$ by using $\mathsf{rk}^{f}_{i \to j}$ can be indistinguishably changed to a simulated one generated without $\mathsf{rk}^{f}_{i \to j}$.

The SH security is exactly what we need to upgrade our CPA security proof strategy to HRA security: roughly speaking, by the SH security, we can change all re-encryptions $ct^{(i)} \to ct^{(j)}$ queried by $\mathcal{A}$ to simulated ones without using re-encryption keys (thus $sk^{(i^*)}$ is not involved); then by the wKP security, we can change all re-encryption keys $\{\mathsf{rk}^{f}_{i \to j}\}$ queried by $\mathcal{A}$ to simulated ones without using $sk^{(i^*)}$; finally, by the IND security, the challenge ciphertext ct^* of the target user i^* hides the underlying message.

For achieving *adaptive* HRA security for multi-hop FPRE, we also extend the framework of Jafargholi et al. [12] and the techniques of Fuchsbauer et al. [9], and establish a theorem by reducing the adaptive HRA security to IND, wKP and SH, with similar security loss.

Finally, we give a high-level overview of our second multi-hop FPRE scheme which additionally satisfies SH security. More precisely, we augment each ciphertext with a level $v \in \mathbb{N}$, and use different noise distribution χ_v for the generation of ciphertexts of different levels. Namely, the v-level ciphertext of user i is now generated by

$$ct_v^{(i)} := \mathbf{A}^{(i)}\mathbf{s} + \mathbf{e} + \begin{pmatrix} \mathbf{0} \\ \lfloor q/2 \rfloor \mathbf{m} \end{pmatrix} \quad \text{with } \mathbf{s} \text{ and } \mathbf{e} \text{ following } \chi_v. \tag{6}$$

Moreover, we randomize the generation of re-encryption $ct_v^{(i)} \to ct_{v+1}^{(j)}$ with $\mathsf{rk}_{i\to j}^{f_{\mathbf{M}}}$ by adding noises, i.e., choosing $\tilde{\mathbf{s}}$ and $\tilde{\mathbf{e}}$ according to χ_{v+1} and computing

$$\begin{aligned} ct_{v+1}^{(j)} &:= \mathsf{rk}_{i\to j}^{f_{\mathbf{M}}} \cdot ct_v^{(i)} + \boxed{\mathbf{A}^{(j)}\tilde{\mathbf{s}} + \tilde{\mathbf{e}}} \\ &= \mathbf{A}^{(j)} \underbrace{\mathbf{S}\mathbf{s}}_{:=\mathbf{s}'} + \underbrace{\mathbf{E}\mathbf{s} + \mathbf{R}\bar{\mathbf{e}} + \begin{pmatrix} \mathbf{0} \\ \mathbf{M}\bar{\mathbf{e}} \end{pmatrix}}_{:=\mathbf{e}'} + \Big(\lfloor q/2 \rfloor \cdot \underbrace{\mathbf{M} \cdot \mathbf{m}}_{=f_{\mathbf{M}}(\mathbf{m})} \Big) + \boxed{\mathbf{A}^{(j)}\tilde{\mathbf{s}} + \tilde{\mathbf{e}}} \qquad (7) \\ &= \mathbf{A}^{(j)} \big(\boxed{\tilde{\mathbf{s}}} + \mathbf{s}' \big) + \big(\boxed{\tilde{\mathbf{e}}} + \mathbf{e}' \big) + \begin{pmatrix} \mathbf{0} \\ \lfloor q/2 \rfloor \cdot \mathbf{M} \cdot \mathbf{m} \end{pmatrix}, \qquad (8) \end{aligned}$$

where (7) follows from (5). By choosing the noise distribution χ_v carefully, we can ensure that $\boxed{\tilde{\mathbf{s}}}$ smudges $\mathbf{s}'$ and $\boxed{\tilde{\mathbf{e}}}$ smudges $\mathbf{e}'$. Consequently, the honestly generated re-encryption $ct_{v+1}^{(j)}$ in (8) is statistically indistinguishable from a freshly generated $(v+1)$-level ciphertext of user j that encrypts $\mathbf{M} \cdot \mathbf{m}$ according to (6), without using $\mathsf{rk}_{i\to j}^{f_{\mathbf{M}}}$. This shows the SH security of this scheme. Similar to our first scheme, this scheme also achieves IND and wKP securities, thus achieving adaptive HRA security via the generic theorem.

Interestingly, we also show that the SH security together with the CPA security (or HRA security) imply *ciphertext unlinkability* (CUL), which can blur the proxy relations in a chain of multi-hop re-encryptions in a more complex setting.

Relations to Existing Works. Finally, we summarize the results already known in the non-fine-grained setting or in the single-hop fine-grained setting, and the results that are novel in our work.

The weaker security notions $\mathsf{IND}, \mathsf{wKP}, \mathsf{SH}$ were originally defined by Fuchsbauer et al. [9] for (non-fine-grained) PRE. Fuchsbauer et al. [9] also established two theorems showing adaptive CPA security based on IND and wKP and showing adaptive HRA security based on IND, wKP and SH, respectively, for (non-fine-grained) PRE, building upon the framework of Jafargholi et al. [12].

The notion of single-hop FPRE and its CUL security were recently introduced by Zhou et al. [21], where they also formally proved the relation that CPA implies UNID for single-hop FPRE.

In our work, we propose the concept of multi-hop FPRE to support multi-hop fine-grained re-encryptions, and formalize a set of security notions $\mathsf{CPA}, \mathsf{HRA}, \mathsf{IND}, \mathsf{wKP}, \mathsf{SH}, \mathsf{UNID}, \mathsf{CUL}$ in the multi-hop fine-grained setting. Moreover, we establish several useful relations between these security notions for multi-hop FPRE, by adapting the two theorems in [9] and the relation in [21] to our multi-hop FPRE. Besides, we show the relation that $\mathsf{SH} + \mathsf{CPA} \Rightarrow \mathsf{CUL}$ holds for our multi-hop FPRE, which is for the first time established for PRE (no matter in which setting). Furthermore, we construct two multi-hop FPRE schemes from LWE, and prove their $\mathsf{IND}, \mathsf{wKP}$ and SH securities based on the LWE assumption in the standard model, which are novel in our work. According to the relations we established (i.e., Theorem 1 and Theorem 2), the two multi-hop FPRE schemes achieves adaptive CPA and adaptive HRA securities, respectively.

2 Preliminaries

Notations. Let $\lambda \in \mathbb{N}$ denote the security parameter throughout the paper, and all algorithms, distributions, functions and adversaries take 1^λ as an implicit input. If x is defined by y or the value of y is assigned to x, we write $x := y$. For $i, j \in \mathbb{N}$ with $i < j$, define $[i, j] := \{i, i+1, ..., j\}$ and $[j] := \{1, 2, ..., j\}$. For a set $\mathcal{X}$, denote by $x \leftarrow\!\!\$\ \mathcal{X}$ the procedure of sampling x from $\mathcal{X}$ uniformly at random. If $\mathcal{D}$ is distribution, $x \leftarrow\!\!\$\ \mathcal{D}$ means that x is sampled according to $\mathcal{D}$. All our algorithms are probabilistic unless stated otherwise. We use $y \leftarrow\!\!\$\ \mathcal{A}(x)$ to define the random variable y obtained by executing algorithm $\mathcal{A}$ on input x. If $\mathcal{A}$ is deterministic we write $y \leftarrow \mathcal{A}(x)$. "PPT" abbreviates probabilistic polynomial-time. Denote by negl some negligible function. By $\Pr_i[\cdot]$ we denote the probability of a particular event occurring in game G_i.

For random variables X and Y, the min-entropy of X is defined as $\mathbf{H}_\infty(X) := -\log(\max_x \Pr[X = x])$, and the statistical distance between X and Y is defined as $\Delta(X, Y) := \frac{1}{2} \cdot \sum_x |\Pr[X = x] - \Pr[Y = x]|$. If $\Delta(X, Y) = \mathsf{negl}(\lambda)$, we say that X and Y are statistically indistinguishable (close), and denote it by $X \approx_s Y$.

Let $n, m, m', q \in \mathbb{N}$, and let $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$, $\mathbf{v} \in \mathbb{Z}_q^n$, $\mathbf{B} \in \mathbb{Z}_q^{m' \times n}$. Define the lattice $\Lambda(\mathbf{A}) := \{\mathbf{Ax} \mid \mathbf{x} \in \mathbb{Z}^n\}$, the q-ary lattice $\Lambda_q(\mathbf{A}) := \{\mathbf{Ax} \mid \mathbf{x} \in \mathbb{Z}_q^n\} + q\mathbb{Z}^m$, its "orthogonal" lattice $\Lambda_q^\perp(\mathbf{A}) := \{\mathbf{x} \in \mathbb{Z}^m \mid \mathbf{x}^\top \mathbf{A} = \mathbf{0} \mod q\}$, and the "shifted" lattice $\Lambda_q^\mathbf{v}(\mathbf{A}) := \{\mathbf{r} \in \mathbb{Z}^m \mid \mathbf{r}^\top \mathbf{A} = \mathbf{v}^\top \mod q\}$, which can be further extended to $\Lambda_q^\mathbf{B}(\mathbf{A}) := \{\mathbf{R} \in \mathbb{Z}^{m' \times m} \mid \mathbf{RA} = \mathbf{B} \mod q\}$. Let $\|\mathbf{v}\|$ (resp., $\|\mathbf{v}\|_\infty$) denote its ℓ_2 (resp., infinity) norm. For a matrix $\mathbf{A}$, we define $\|\mathbf{A}\|$ (resp., $\|\mathbf{A}\|_\infty$) as the largest ℓ_2 (resp., infinity) norm of $\mathbf{A}$'s rows. A distribution χ is B-bounded if its support is limited to $[-B, B]$. Let $\mathbb{Z}_q$ be the ring of integers modulo q, and its elements are represented by the integers in $(-q/2, q/2]$.

Due to space limitations, we present lattice backgrounds in the full version [20], where we recall the definitions of discrete Gaussian, LWE assumption, and the $\mathsf{TrapGen}$, Invert, $\mathsf{SamplePre}$ algorithms introduced in [1,11,16].

3 Multi-hop Fine-Grained PRE

In this section, we formalize a new primitive called *Multi-Hop Fine-Grained PRE* (mFPRE), by extending the concept of single-hop FPRE proposed in [21] to support multi-hop of re-encryptions. Compared with (traditional) PRE, FPRE allows fine-grained delegations, by associating re-encryption key $\mathsf{rk}_{i \to j}^f$ with a function f to support the conversion of user i's ciphertext $ct^{(i)}$ encrypting message m to user j's ciphertext $ct^{(j)}$ encrypting the function value $f(m)$. Moreover, in contrast to single-hop FPRE, our multi-hop FPRE supports multiple re-encryptions, namely, user j's re-encrypted ciphertext $ct^{(j)}$ encrypting $f(m)$ can be further re-encrypted to user k's ciphertext $ct^{(k)}$ encrypting $f'(f(m))$ with the help of another $\mathsf{rk}_{j \to k}^{f'}$, and as forth. These multiple re-encryptions can be correctly decrypted to the corresponding function values, as long as the number of re-encryption hops does not exceed the maximum level.

As for security, we formalize the CPA and HRA security for multi-hop FPRE. To achieve both security, we adapt the framework proposed in [9,12] to fine-grained setting and establish two theorems reducing CPA and HRA to a set of weaker security notions, including indistinguishablity (IND), weak key-privacy (wKP) and source-hiding (SH), for multi-hop FPRE. Furthermore, we introduce some other security properties including unidirectionality (UNID) and ciphertext unlinkability (CUL) for multi-hop FPRE. See Fig. 1 in introduction for an overview of the relations between these security notions.

3.1 Syntax of Multi-hop FPRE and Its CPA and HRA Security

Definition 1 (Multi-Hop Fine-Grained PRE). *Let $\mathcal{F}$ be a family of functions from $\mathcal{M}$ to $\mathcal{M}$, where $\mathcal{M}$ is a message space. A multi-hop fine-grained proxy re-encryption (multi-hop FPRE) scheme for function family $\mathcal{F}$ is associated with a maximum level $L \in \mathbb{N}$ and defined with a tuple of PPT algorithms* $\mathsf{mFPRE} = (\mathsf{KGen}, \mathsf{FReKGen}, \mathsf{Enc}, \mathsf{FReEnc}, \mathsf{Dec})$.

- $(pk, sk) \leftarrow_{\$} \mathsf{KGen}$*: The key generation algorithm outputs a pair of public key and secret key (pk, sk).*
- $\mathsf{rk}^{f}_{i\to j} \leftarrow_{\$} \mathsf{FReKGen}(pk^{(i)}, sk^{(i)}, pk^{(j)}, f)$*: Taking as input a public-secret key pair $(pk^{(i)}, sk^{(i)})$, another public key $pk^{(j)}$ and a function $f \in \mathcal{F}$, the fine-grained re-encryption key generation algorithm outputs a fine-grained re-encryption key $\mathsf{rk}^{f}_{i\to j}$ that allows re-encrypting ciphertexts intended to i into ciphertexts encrypted for j.*
- $ct_v \leftarrow_{\$} \mathsf{Enc}(pk, m, v)$*: Taking as input pk, a message $m \in \mathcal{M}$ and a level $v \in [0, L]$, the encryption algorithm outputs a v-level ciphertext ct_v.*
- $ct^{(j)}_{v+1} \leftarrow_{\$} \mathsf{FReEnc}(\mathsf{rk}^{f}_{i\to j}, ct^{(i)}_{v}, v)$*: Taking as input a re-encryption key $\mathsf{rk}^{f}_{i\to j}$ and a ciphertext $ct^{(i)}_{v}$ intended for i and its level $v \in [0, L-1]$, the fine-grained re-encryption algorithm outputs a $(v+1)$-level ciphertext $ct^{(j)}_{v+1}$ re-encrypted for j. We denote it by $ct^{(i)}_{v} \xrightarrow{\mathsf{rk}^{f}_{i\to j}} ct^{(j)}_{v+1}$.*
- $m \leftarrow \mathsf{Dec}(sk, ct)$*: Taking as input a secret key sk and a ciphertext ct, the deterministic decryption algorithm outputs a message m.*

Correctness. *For all $m \in \mathcal{M}, v \in [0, L]$, $(pk, sk) \leftarrow_{\$} \mathsf{KGen}$, $ct_v \leftarrow_{\$} \mathsf{Enc}(pk, m, v)$, it holds that $\mathsf{Dec}(sk, ct_v) = m$.*

Fine-Grained L-Hop Correctness. *For all $m \in \mathcal{M}$, user indices $i_0, i_1, \cdots, i_L$, functions $f_1, \cdots, f_L \in \mathcal{F}$, $(pk^{(i_j)}, sk^{(i_j)}) \leftarrow_{\$} \mathsf{KGen}$ with $j \in [0, L]$, 0-level ciphertext $ct^{(i_0)}_0 \leftarrow_{\$} \mathsf{Enc}(pk^{(i_0)}, m, 0)$ and re-encryption hops $ct^{(i_0)}_0 \xrightarrow{\mathsf{rk}^{f_1}_{i_0\to i_1}} ct^{(i_1)}_1 \xrightarrow{\mathsf{rk}^{f_2}_{i_1\to i_2}} \cdots \xrightarrow{\mathsf{rk}^{f_L}_{i_{L-1}\to i_L}} ct^{(i_L)}_L$, where each $\mathsf{rk}^{f_j}_{i_{j-1}\to i_j} \leftarrow_{\$} \mathsf{FReKGen}(pk^{(i_{j-1})}, sk^{(i_{j-1})}, pk^{(i_j)}, f_j)$ and each $ct^{(i_j)}_j \leftarrow_{\$} \mathsf{FReEnc}(\mathsf{rk}^{f_j}_{i_{j-1}\to i_j}, ct^{(i_{j-1})}_{j-1}, j-1)$, it holds that for all $j \in [L]$,*

$$\mathsf{Dec}(sk^{(i_j)}, ct^{(i_j)}_j) = f_j(f_{j-1}(\ldots f_1(m))).$$

CPA Security. Below we formalize the indistinguishability of ciphertexts under chosen-plaintext attacks (CPA) for multi-hop FPRE.

Definition 2 (CPA Security for Multi-hop FPRE). *A multi-hop FPRE scheme* mFPRE *is* CPA *secure, if for any PPT adversary* $\mathcal{A}$ *and any polynomial* $\mathfrak{n}$*, it holds that* $\mathsf{Adv}^{\mathsf{CPA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}(\lambda) := \left|\Pr[\mathsf{Exp}^{\mathsf{CPA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}} \Rightarrow 1] - \frac{1}{2}\right| \leq \mathsf{negl}(\lambda)$*, where the experiment* $\mathsf{Exp}^{\mathsf{CPA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ *is defined in Fig. 2.*

$\underline{\mathsf{Exp}^{\mathsf{CPA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}}$:
For $i \in [\mathfrak{n}]$: $(pk^{(i)}, sk^{(i)}) \leftarrow\!\!\$\ \mathsf{KGen}$
$\mathcal{Q}_{rk} := \emptyset$ //record re-encryption key queries
$\mathcal{Q}_c := \emptyset$ //record corruption queries
$i^* := \perp$ //record challenge user
$(i^*, m_0, m_1, v, st) \leftarrow\!\!\$\ \mathcal{A}^{\mathcal{O}_{\mathrm{ReKey}}(\cdot,\cdot,\cdot),\mathcal{O}_{\mathrm{Cor}}(\cdot)}(\{pk^{(i)}\}_{i\in[\mathfrak{n}]})$
If $(i^* \in \mathcal{Q}_c)$ or $\mathsf{CheckTA}(i^*, \mathcal{Q}_{rk}, \mathcal{Q}_c) = 1$:
  Return $b \leftarrow\!\!\$\ \{0,1\}$ //avoid **TA1**, **TA2**
$\beta \leftarrow\!\!\$\ \{0,1\}$
$ct_v^* \leftarrow\!\!\$\ \mathsf{Enc}(pk^{(i^*)}, m_\beta, v)$
$\beta' \leftarrow\!\!\$\ \mathcal{A}^{\mathcal{O}_{\mathrm{ReKey}}(\cdot,\cdot,\cdot),\mathcal{O}_{\mathrm{Cor}}(\cdot)}(st, ct_v^*)$
If $\beta' = \beta$: Return 1; Else: Return 0

$\underline{\mathcal{O}_{\mathrm{ReKey}}(i,j,f)}$: //re-encryption key queries
If $\mathsf{CheckTA}(i^*, \mathcal{Q}_{rk} \cup \{(i,j)\}, \mathcal{Q}_c) = 1$:
  Return $\perp$ //avoid **TA2**
$\mathcal{Q}_{rk} := \mathcal{Q}_{rk} \cup \{(i,j)\}$
$\mathsf{rk}^f_{i\to j} \leftarrow\!\!\$\ \mathsf{FReKGen}(pk^{(i)}, sk^{(i)}, pk^{(j)}, f)$
Return $\mathsf{rk}^f_{i\to j}$

$\underline{\mathcal{O}_{\mathrm{Cor}}(i)}$: //corruption queries
If $i = i^*$: Return $\perp$ //avoid **TA1**
If $\mathsf{CheckTA}(i^*, \mathcal{Q}_{rk}, \mathcal{Q}_c \cup \{i\}) = 1$:
  Return $\perp$ //avoid **TA2**
$\mathcal{Q}_c := \mathcal{Q}_c \cup \{i\}$
Return $sk^{(i)}$

$\underline{\mathsf{CheckTA}(i^*, \mathcal{Q}_{rk}, \mathcal{Q}_c)}$: //check **TA2**
If $\exists\ (i^*, j_1), (j_1, j_2), \ldots, (j_{t-1}, j_t) \in \mathcal{Q}_{rk}$
  s.t. $j_t \in \mathcal{Q}_c$ for some $t \geq 1$:
  Return 1
Else: Return 0

Fig. 2. The CPA security experiment $\mathsf{Exp}^{\mathsf{CPA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ for mFPRE. Here CheckTA is a sub-procedure used to check the trivial attacks.

Remark 1 (On the formalization of CPA *security and discussion on trivial attacks).* We formalize the CPA security by defining the experiment $\mathsf{Exp}^{\mathsf{CPA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ in Fig. 2. More precisely, we consider a multi-user setting, and the adversary $\mathcal{A}$ is allowed to make two kinds of oracle queries *adaptively*:

- through $\mathcal{O}_{\mathrm{ReKey}}(i,j,f)$ query, $\mathcal{A}$ can get re-encryption keys $\mathsf{rk}^f_{i\to j}$, and
- through $\mathcal{O}_{\mathrm{Cor}}(i)$ query, $\mathcal{A}$ can corrupt user i and obtain its secret key $sk^{(i)}$.

We stress that the adversary can issue multiple $\mathcal{O}_{\mathrm{ReKey}}(i,j,f)$ queries, even for the same delegator i and same delegatee j, thus achieving *multiple delegations*. At some point, $\mathcal{A}$ outputs a challenge user i^*, a pair of messages (m_0, m_1) as well as a level v, and receives a challenge ciphertext ct_v^* which encrypts m_β under $pk^{(i^*)}$ at level v, where β is the challenge bit that $\mathcal{A}$ aims to guess.

To prevent trivial attacks from $\mathcal{A}$, we keep track of two sets: $\mathcal{Q}_c$ records the corrupted users, and $\mathcal{Q}_{rk}$ records the tuples (i,j) that $\mathcal{A}$ obtains a re-encryption key $\mathsf{rk}^f_{i\to j}$. Based on that, there are two kinds of trivial attacks **TA1**-**TA2** to obtain information about the plaintext underlying the challenge ciphertext ct_v^*.

TA1: $i^* \in \mathcal{Q}_c$, i.e., $\mathcal{A}$ corrupts user i^* and obtains its secret key $sk^{(i^*)}$. In this case, $\mathcal{A}$ can decrypt ct_v^* directly via $\mathsf{Dec}(sk^{(i^*)}, ct_v^*)$ and recover m_β.

TA2: $\exists (i^*, j_1), (j_1, j_2), \ldots, (j_{t-1}, j_t) \in \mathcal{Q}_{rk}$ s.t. $j_t \in \mathcal{Q}_c$ for some $t \geq 1$, i.e., $\mathcal{A}$ gets a chain of re-encryption keys $\mathsf{rk}_{i^* \to j_1}^{f_1}, \mathsf{rk}_{j_1 \to j_2}^{f_2}, \ldots, \mathsf{rk}_{j_{t-1} \to j_t}^{f_t}$ starting from the challenge user i^* and ending at some corrupted user j_t for whom $\mathcal{A}$ ever obtains its secret key $sk^{(j_t)}$. In this case, $\mathcal{A}$ can re-encrypt ct_v^* via $ct_v^* \xrightarrow{\mathsf{rk}_{i^* \to j_1}^{f_1}} ct_{v+1}^{(j_1)} \xrightarrow{\mathsf{rk}_{j_1 \to j_2}^{f_2}} \cdots \xrightarrow{\mathsf{rk}_{j_{t-1} \to j_t}^{f_t}} ct_{v+t}^{(j_t)}$, then simply decrypt $ct_{v+t}^{(j_t)}$ with $sk^{(j_t)}$ to obtain a function of m_β. This kind of trivial attacks is checked by the algorithm $\mathsf{CheckTA}$ defined in Fig. 2 throughout the experiment.

As such, we exclude the above trivial attacks in the CPA experiment.

We note that in contrast to the CPA security for PRE defined in [9], our CPA security does not provide a re-encryption oracle for re-encrypting ciphertexts from the challenge user i^* to uncorrupted users $j \notin \mathcal{Q}_c$. This is because in our CPA experiment, $\mathcal{A}$ can obtain re-encryption keys from i^* to $j \notin \mathcal{Q}_c$ through the $\mathcal{O}_{\mathrm{ReKey}}$ oracle and do re-encryption itself for such ciphertexts.

HRA Security. Next we formalize the indistinguishability of ciphertexts under honest-re-encryption attacks (HRA) for multi-hop FPRE. Originally, HRA was first introduced by Cohen [6] as a stronger and more reasonable security notion than CPA for PRE. Below we adapt HRA security to the fine-grained setting for mFPRE. Compared with the CPA security, HRA also allows the adversary to have access to a re-encryption oracle $\mathcal{O}_{\mathrm{ReEnc}}$, through which the adversary can learn re-encryptions of ciphertexts from the challenge user i^* to *corrupted* users $j \in \mathcal{Q}_c$, as long as the queried ciphertexts are honestly generated and different from (all derivatives of) the challenge ciphertext ct_v^*.

Definition 3 (HRA Security for Multi-Hop FPRE). *A multi-hop FPRE scheme* mFPRE *is* HRA *secure, if for any PPT adversary* $\mathcal{A}$ *and any polynomial* $\mathfrak{n}$*, it holds that* $\mathsf{Adv}^{\mathsf{HRA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}(\lambda) := \left| \Pr[\mathsf{Exp}^{\mathsf{HRA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}} \Rightarrow 1] - \frac{1}{2} \right| \leq \mathsf{negl}(\lambda)$*, where the experiment* $\mathsf{Exp}^{\mathsf{HRA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ *is defined in Fig. 3.*

Remark 2 (On the formalization of HRA *security and discussion on trivial attacks).* We formalize the HRA security by defining the experiment $\mathsf{Exp}^{\mathsf{HRA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ in Fig. 3. More precisely, we consider a multi-user setting, and the adversary $\mathcal{A}$ is allowed to make four kinds of oracle queries *adaptively*:

- through $\mathcal{O}_{\mathrm{ReKey}}(i, j, f)$ query, $\mathcal{A}$ can get re-encryption keys $\mathsf{rk}_{i \to j}^{f}$;
- through $\mathcal{O}_{\mathrm{Cor}}(i)$ query, $\mathcal{A}$ can corrupt user i and obtain its secret key $sk^{(i)}$;
- through $\mathcal{O}_{\mathrm{Enc}}(i, m, v)$ query, $\mathcal{A}$ can obtain honestly generated ciphertexts, which are indexed by counters ctr and can be further re-encrypted through $\mathcal{O}_{\mathrm{ReEnc}}$ query;
- through $\mathcal{O}_{\mathrm{ReEnc}}(i, j, f, k)$ query, $\mathcal{A}$ can obtain re-encryptions of honestly generated ciphertexts (including the challenge ciphertext ct_v^* to be defined later, as well as the re-encrypted ciphertexts output by $\mathcal{O}_{\mathrm{ReEnc}}$ previously), where k is the index of the honestly generated ciphertext to be re-encrypted and i, j, f specify the re-encryption key $\mathsf{rk}_{i \to j}^{f}$ to be used.

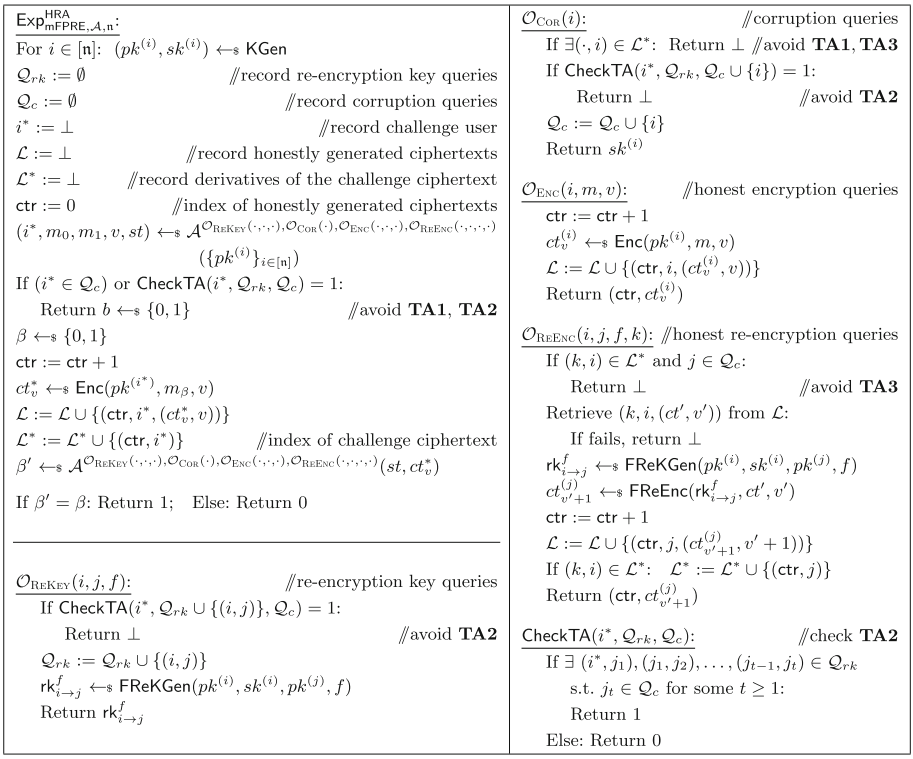

Fig. 3. The HRA security experiment $\mathsf{Exp}^{\mathsf{HRA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ for mFPRE. Here the oracles $\mathcal{O}_{\mathrm{ReKey}}$, $\mathcal{O}_{\mathrm{Cor}}$ and the sub-procedure CheckTA are the same as those in Fig. 2.

At some point, $\mathcal{A}$ outputs a challenge user i^*, a pair of messages (m_0, m_1) as well as a level v, and receives a challenge ciphertext ct_v^* which encrypts m_β under $pk^{(i^*)}$ at level v, where β is the challenge bit that $\mathcal{A}$ aims to guess.

Similar to the CPA security, we also exclude the two trivial attacks **TA1**-**TA2** as defined in Remark 1, from which $\mathcal{A}$ can trivially obtain information about the plaintext m_β underlying the challenge ciphertext ct_v^*. Moreover, there is an additional trivial attack **TA3** to obtain information about m_β.

TA3: Via $\mathcal{O}_{\mathrm{ReEnc}}$ queries, $\mathcal{A}$ obtains a chain of re-encryptions $ct_v^* \xrightarrow{\mathcal{O}_{\mathrm{ReEnc}}} ct_{v+1}^{(j_1)} \xrightarrow{\mathcal{O}_{\mathrm{ReEnc}}} \cdots \xrightarrow{\mathcal{O}_{\mathrm{ReEnc}}} ct_{v+t}^{(j_t)}$ starting from the challenge ciphertext ct_v^* and ending at ciphertext $ct_{v+t}^{(j_t)}$ of some corrupted user $j_t \in \mathcal{Q}_c$ from whom $\mathcal{A}$ ever obtains its secret key $sk^{(j_t)}$. In this case, $\mathcal{A}$ can use $sk^{(j_t)}$ to decrypt $ct_{v+t}^{(j_t)}$ to trivially obtain a function of m_β.

To exclude this additional trivial attack, we keep track of a set $\mathcal{L}^*$ to record (index of) the challenge ciphertext ct_v^* as well as all honestly generated re-encryptions of ct_v^* output by $\mathcal{O}_{\mathrm{ReEnc}}$.

3.2 Achieving CPA and HRA Security for Multi-hop FPRE from Weaker Security Notions: IND, wKP and SH

Our CPA and HRA security for multi-hop FPRE formalized in the previous subsection are defined in an *adaptive* manner, where the adversary $\mathcal{A}$ can designate the challenge user i^* and make all oracle queries adaptively, including corruption queries $\mathcal{O}_{\text{COR}}$, re-encryption key queries $\mathcal{O}_{\text{REKEY}}$, and honest encryption queries $\mathcal{O}_{\text{ENC}}$ and honest re-encryption queries $\mathcal{O}_{\text{REENC}}$ in the case of HRA. Accordingly, the tuples (i, j) for which $\mathcal{A}$ obtains a re-encryption key $\mathsf{rk}^f_{i\to j}$ (i.e., the set $\mathcal{Q}_{rk}$ in Fig. 2 and Fig. 3) are adaptively determined by $\mathcal{A}$ and form a complex directed graph. In the case of HRA, the tuples (i, j) for which $\mathcal{A}$ makes a re-encryption query $\mathcal{O}_{\text{REENC}}(i, j, \cdot, \cdot)$ form another complex directed graph.

One possible way to achieve adaptive CPA/HRA security is first proving a selective version of CPA/HRA security, and then reducing the adaptive security to the selective counterpart via a guessing strategy. The selective CPA/HRA security means that $\mathcal{A}$ has to declare the graphs for re-encryption keys/re-encryptions at the beginning of the experiment, and thus it is relatively easy to prove selective security in general. However, the price is a considerably large security loss $O(2^{n^2})$ incurred by the guessing of the graphs.

To reduce the security loss of adaptive security, Jafargholi et al. [12] proposed a generic framework for upgrading selective security to adaptive security with a more fine-grained analysis. Later, Fuchsbauer et al. [9] applied the framework of [12] to the CPA/HRA security of (traditional) PRE.

In this subsection, we will extend the framework of Jafargholi et al. [12] further to the CPA and HRA security of our multi-hop fine-grained PRE, by adapting the techniques of Fuchsbauer et al. [9] to the fine-grained setting. More precisely, we will first defined three weaker security notions, including indistinguishability (IND), weak key-privacy (wKP) and source-hiding (SH), to our multi-hop FPRE, and then establish two theorems showing $\mathsf{CPA}, \mathsf{HRA}$ security of our multi-hop FPRE based on these weaker security notions. The formalization of the weaker security notions and the proofs of the theorems are mainly adapted from [9,12].

Now we present the formal definitions of IND, wKP, SH for multi-hop FPRE.

Indistinguishability. The IND security of multi-hop FPRE considers the indistinguishability of ciphertexts in a single-user and multi-challenge setting, where the adversary is given no re-encryption keys compared with the CPA security.

Definition 4 (IND Security). *A multi-hop FPRE scheme* mFPRE *is* IND *secure, if for any PPT adversary* $\mathcal{A}$*, it holds that* $\mathsf{Adv}^{\mathsf{IND}}_{\mathsf{mFPRE},\mathcal{A}}(\lambda) := |\Pr[\mathsf{Exp}^{\mathsf{IND}}_{\mathsf{mFPRE},\mathcal{A}} \Rightarrow 1] - \frac{1}{2}| \leq \mathsf{negl}(\lambda)$*, where the experiment* $\mathsf{Exp}^{\mathsf{IND}}_{\mathsf{mFPRE},\mathcal{A}}$ *is defined in Fig. 4.*

Weak Key-Privacy. The original key-privacy for PREs was introduced in [2]. In [9], weak key-privacy was introduced and it requires the indistinguishability between the re-encryption key $\mathsf{rk}_{0\to j}$ from user 0 to user j and the re-encryption key $\mathsf{rk}_{1\to j}$ from user 1 to user j. Below we adapt it to our multi-hop FPRE, by

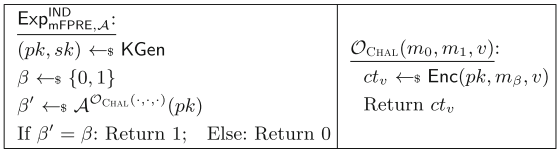

Fig. 4. The indistinguishability experiment $\mathsf{Exp}^{\mathsf{IND}}_{\mathsf{mFPRE},\mathcal{A}}$ for mFPRE.

requiring the existence of a PPT algorithm $\mathsf{FReKGen}^*$ which can simulate the generation of re-encryption keys $\mathsf{rk}^f_{0\to j}$ without the secret key of source user 0.

Definition 5 (wKP Security). *A multi-hop FPRE scheme* mFPRE *has weak key privacy (*wKP *security), if there exists a PPT simulation algorithm* $\mathsf{FReKGen}^*$*, s.t. for any PPT adversary* $\mathcal{A}$ *and any polynomial* $\mathfrak{n}$*, it holds that* $\mathsf{Adv}^{\mathsf{wKP}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}(\lambda) := |\Pr[\mathsf{Exp}^{\mathsf{wKP}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}} \Rightarrow 1] - \frac{1}{2}| \leq \mathsf{negl}(\lambda)$*, where* $\mathsf{Exp}^{\mathsf{wKP}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ *is defined in Fig. 5.*

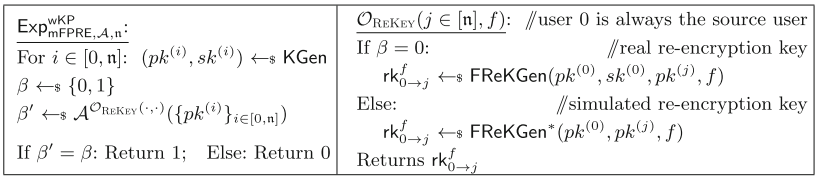

Fig. 5. The weak key-privacy experiment $\mathsf{Exp}^{\mathsf{wKP}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ for mFPRE.

Source-Hiding. Roughly speaking, source-hiding (SH) requires the indistinguishability between freshly-encrypted ciphertexts (via Enc) and re-encrypted ciphertexts (via FReEnc), even if the adversary has all secret keys and re-encryption keys. SH security can help us upgrade CPA security to HRA security for FPRE.

Definition 6 (SH Security). *A multi-hop FPRE scheme* mFPRE *has the property of source-hiding (*SH *security), if for any (unbounded) adversary* $\mathcal{A}$*, it holds that* $\mathsf{Adv}^{\mathsf{SH}}_{\mathsf{mFPRE},\mathcal{A}}(\lambda) := |\Pr[\mathsf{Exp}^{\mathsf{SH}}_{\mathsf{mFPRE},\mathcal{A}} \Rightarrow 1] - \frac{1}{2}| \leq \mathsf{negl}(\lambda)$*, where experiment* $\mathsf{Exp}^{\mathsf{SH}}_{\mathsf{mFPRE},\mathcal{A}}$ *is defined in Fig. 6.*

Achieving CPA and HRA Security for Multi-Hop FPRE. Now we are ready to present two theorems showing (adaptive) CPA and HRA of multi-hop FPRE assuming the weak security notions IND, wKP and SH. The theorems are essentially applications of the framework of Jafargholi et al. [12] and adaptions of the techniques of Fuchsbauer et al. [9] to multi-hop FPRE. We refer to the full version [20] for their proofs, as they almost verbatim follow [9,12].

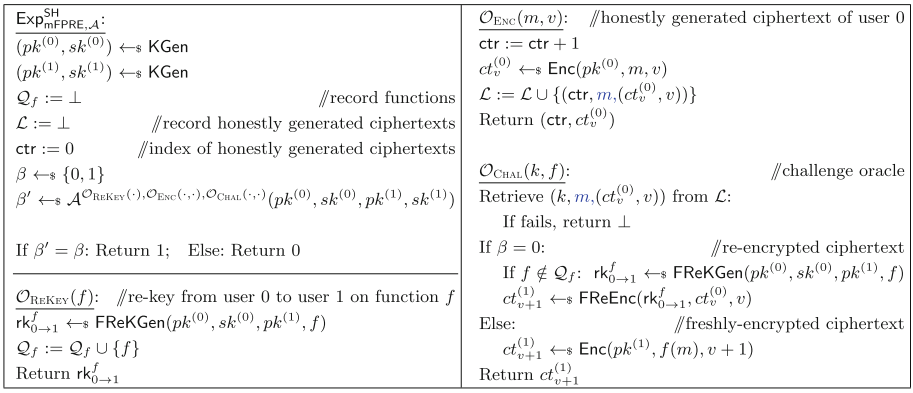

Fig. 6. The source-hiding experiment $\mathsf{Exp}^{\mathsf{SH}}_{\mathsf{mFPRE},\mathcal{A}}$ for mFPRE.

To state the theorems precisely, we consider an adversary $\mathcal{A}$ in the CPA/HRA security experiment, and define some notations. If we view users $[\mathfrak{n}]$ as vertices and re-encryption keys $\mathsf{rk}^f_{i\to j}$ that $\mathcal{A}$ obtains through $\mathcal{O}_{\mathrm{ReKey}}$ queries as an edge from i to j, then it forms a directed graph. We define the subgraph that is reachable from the challenge user i^* as *the challenge graph* of $\mathcal{A}$, denoted by G. For the challenge graph G, if we denote by δ the outdegree (i.e., the maximum outdegree over all vertices) and d the depth, then the challenge graph is in the graph class $\mathcal{G}(\mathfrak{n}, \delta, d)$ of all graphs with $\mathfrak{n}$ vertices, outdegree δ and depth d.

In the full version [20], we further define the pebbling time complexity τ and space complexity σ for the class $\mathcal{G}(\mathfrak{n}, \delta, d)$, respectively, according to [9,12].

Theorem 1 (IND + wKP ⇒ CPA for Multi-Hop FPRE). *If a multi-hop FPRE scheme* mFPRE *has both* IND *and* wKP *security, then it is* CPA *secure.*

More precisely, for any PPT adversary $\mathcal{A}$ against the CPA *security with challenge graph G in $\mathcal{G}(\mathfrak{n}, \delta, d)$ whose pebbling time complexity is τ and space complexity is σ, there exist PPT algorithms $\mathcal{B}$ and $\mathcal{B}'$ s.t.* $\mathsf{Adv}^{\mathsf{CPA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}(\lambda) \le (2 \cdot \mathsf{Adv}^{\mathsf{IND}}_{\mathsf{mFPRE},\mathcal{B}} + 2\tau \cdot \mathsf{Adv}^{\mathsf{wKP}}_{\mathsf{mFPRE},\mathcal{B}',\delta}) \cdot \mathfrak{n}^{\sigma+\delta+1}$.

Theorem 2 (IND + wKP + SH ⇒ HRA for Multi-Hop FPRE). *If a multi-hop FPRE scheme* mFPRE *has* IND, wKP *and* SH *security simultaneously, then it is* HRA *secure.*

More precisely, for any PPT adversary $\mathcal{A}$ against the HRA *security with challenge graph G in $\mathcal{G}(\mathfrak{n}, \delta, d)$ whose pebbling time complexity is τ and space complexity is σ, there exist PPT algorithms $\mathcal{B}, \mathcal{B}'$ and $\mathcal{B}''$ s.t.* $\mathsf{Adv}^{\mathsf{HRA}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}(\lambda) \le (2 \cdot \mathsf{Adv}^{\mathsf{IND}}_{\mathsf{mFPRE},\mathcal{B}} + 2\tau \cdot \mathsf{Adv}^{\mathsf{wKP}}_{\mathsf{mFPRE},\mathcal{B}',\delta}) \cdot \mathfrak{n}^{\sigma+\delta+1} + 2\mathfrak{n}(\mathfrak{n}-1)L \cdot \mathsf{Adv}^{\mathsf{SH}}_{\mathsf{mFPRE},\mathcal{B}''}$, *where L is the maximum level supported by* mFPRE.[5]

[5] We note that Theorem 2 has slightly different parameters than the corresponding theorem (i.e., Theorem 6) in [9]. This is because we use slightly different proof strategy than [9] when reducing to SH, in order to change all re-encrypted ciphertexts to freshly generated ciphertexts. We refer to the full version [20] for more details.

Note that the security loss of Theorem 1 and Theorem 2 is dominating by $2\tau \cdot \mathfrak{n}^{\sigma+\delta+1}$ and $2\mathfrak{n}(\mathfrak{n}-1)L$.

- For an arbitrary adversary $\mathcal{A}$ with an arbitrary challenge graph G, according to the bounds given in [9], we have the pebbling time complexity $\tau \leq (2\delta)^d$, the space complexity $\sigma \leq \mathfrak{n}$, the outdegree $\delta \leq \mathfrak{n}$ and the depth $d \leq \mathfrak{n}$. Moreover, L is (at most) a polynomial in $\mathfrak{n}$. Consequently, the security loss for arbitrary adversary $\mathcal{A}$ is $\mathfrak{n}^{O(\mathfrak{n})}$.
- In many realistic scenarios like key rotation for encrypted cloud storage or forwarding of encrypted mail, as demonstrated in [9], the proxy relations are in fact *trees, chains or low-depth graphs*, so does the challenge graph G. In these situations, according to the bounds given in [9], we have the pebbling time complexity $\tau = O(1)^{\log \mathfrak{n}}$, the space complexity $\sigma = O(\log \mathfrak{n})$ and the outdegree $\delta =$ constant, and consequently, the security loss is only quasi-polynomial $\mathfrak{n}^{O(\log \mathfrak{n})}$.

3.3 Other Security Notions for Multi-hop FPRE: UNID and CUL

In this subsection, we formalize two additional security notions for multi-hop FPRE, namely unidirectionality (UNID) and ciphertext unlinkability (CUL), by adapting the formalization in [21] defined for single-hop FPRE.

Unidirectionality. Intuitively, unidirectionality (UNID) means that the proxy ability in one direction does not imply the proxy ability in the other direction.

$\underline{\mathsf{Exp}^{\mathsf{UNID}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}}$:
For $i \in [\mathfrak{n}]$: $(pk^{(i)}, sk^{(i)}) \leftarrow\!\!\$\ \mathsf{KGen}$
$\mathcal{Q}_{rk} := \emptyset$ //record re-encryption key queries
$\mathcal{Q}_c := \emptyset$ //record corruption queries
$i^* := \perp, j^* := \perp$ //record challenge users
$(i^*, j^*, f, st) \leftarrow\!\!\$\ \mathcal{A}^{\mathcal{O}_{\mathrm{ReKey}}(\cdot,\cdot,\cdot),\mathcal{O}_{\mathrm{Cor}}(\cdot)}(\{pk^{(i)}\}_{i\in[\mathfrak{n}]})$
If $(i^* = j^*)$ or $(i^* \in \mathcal{Q}_c)$ or $\mathsf{CheckTA}(i^*, j^*, \mathcal{Q}_{rk}, \mathcal{Q}_c) = 1$:
  Return 0 //avoid **TA1′**, **TA2′**, **TA3′**, **TA4′**
$\mathsf{rk}^f_{j^*\to i^*} \leftarrow\!\!\$\ \mathsf{FReKGen}(pk^{(j^*)}, sk^{(j^*)}, pk^{(i^*)}, f)$
$\mathcal{Q}_{rk} := \mathcal{Q}_{rk} \cup \{(j^*, i^*)\}$
$(f', \mathsf{rk}^{f'}_{i^*\to j^*}) \leftarrow\!\!\$\ \mathcal{A}^{\mathcal{O}_{\mathrm{ReKey}}(\cdot,\cdot,\cdot),\mathcal{O}_{\mathrm{Cor}}(\cdot)}(st, \mathsf{rk}^f_{j^*\to i^*})$
If f' does not have output diversity:
  Return $\perp$ //avoid **TA5′**
//check the functionality of $\mathsf{rk}^{f'}_{i^*\to j^*}$ in the following way
$m \leftarrow\!\!\$\ \mathcal{M}$, $ct_0^{(i^*)} \leftarrow\!\!\$\ \mathsf{Enc}(pk^{(i^*)}, m, 0)$
$ct_1^{(j^*)} \leftarrow\!\!\$\ \mathsf{FReEnc}(\mathsf{rk}^{f'}_{i^*\to j^*}, ct_0^{(i^*)}, 0)$
If $\mathsf{Dec}(sk^{(j^*)}, ct_1^{(j^*)}) = f'(m)$:
  Return 1
Else: Return 0

$\underline{\mathcal{O}_{\mathrm{ReKey}}(i, j, f)}$:
  If $\mathsf{CheckTA}(i^*, j^*, \mathcal{Q}_{rk} \cup \{(i,j)\}, \mathcal{Q}_c) = 1$:
    Return $\perp$ //avoid **TA3′**, **TA4′**
  $\mathcal{Q}_{rk} := \mathcal{Q}_{rk} \cup \{(i,j)\}$
  $\mathsf{rk}^f_{i\to j} \leftarrow\!\!\$\ \mathsf{FReKGen}(pk^{(i)}, sk^{(i)}, pk^{(j)}, f)$
  Return $\mathsf{rk}^f_{i\to j}$

$\underline{\mathcal{O}_{\mathrm{Cor}}(i)}$:
  If $i = i^*$: Return $\perp$ //avoid **TA2′**
  If $\mathsf{CheckTA}(i^*, j^*, \mathcal{Q}_{rk}, \mathcal{Q}_c \cup \{i\}) = 1$:
    Return $\perp$ //avoid **TA3′**, **TA4′**
  $\mathcal{Q}_c := \mathcal{Q}_c \cup \{i\}$
  Return $sk^{(i)}$

$\underline{\mathsf{CheckTA}(i^*, j^*, \mathcal{Q}_{rk}, \mathcal{Q}_c)}$: //avoid **TA3′**, **TA4′**
  If $\exists\ (i^*, j_1), (j_1, j_2), \ldots, (j_{t-1}, j_t) \in \mathcal{Q}_{rk}$
    s.t. $(j_t \in \mathcal{Q}_c)$ or $(j_t = j^*)$ for some $t \geq 1$:
    Return 1
  Else: Return 0

Fig. 7. The Unidirectionality security experiment $\mathsf{Exp}^{\mathsf{UNID}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ for mFPRE, where "output diversity" is defined as $\Pr[m_0, m_1 \leftarrow\!\!\$\ \mathcal{M} : f'(m_0) \neq f'(m_1)] \geq 1/\mathsf{poly}(\lambda)$ (see the full version [20] for more details).

Definition 7 (Unidirectionality for Multi-Hop FPRE). *A multi-hop FPRE scheme* mFPRE *is unidirectional (*UNID *secure), if for any PPT adversary* $\mathcal{A}$ *and any polynomial* $\mathfrak{n}$*, it holds that* $\mathsf{Adv}^{\mathsf{UNID}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}(\lambda) := \Pr[\mathsf{Exp}^{\mathsf{UNID}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}} \Rightarrow 1] \leq \mathsf{negl}(\lambda)$*, where the experiment* $\mathsf{Exp}^{\mathsf{UNID}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ *is defined in Fig. 7.*

In the full version [20], we give some explanations of the UNID security definition and discuss the trivial attacks **TA1′**-**TA5′**, and then show that the UNID security is implied by the CPA security for multi-hop FPRE.

Ciphertext Unlinkability. In real scenarios, re-encryption relations between ciphertexts often imply the proxy connections between users. Therefore, it is desirable to hide the relations/connections, which is captured by the property ciphertext unlinkability (CUL). We formalize CUL for multi-hop FPRE by requiring the indistinguishability between a chain of re-encrypted ciphertexts $ct_0^{(i_0)} \xrightarrow{\mathsf{rk}^{f_1}_{i_0 \to i_1}} ct_1^{(i_1)} \xrightarrow{\mathsf{rk}^{f_2}_{i_1 \to i_2}} \cdots \xrightarrow{\mathsf{rk}^{f_L}_{i_{L-1} \to i_L}} ct_L^{(i_L)}$ generated by FReEnc and a set of freshly and independently encrypted ciphertexts $(ct_0^{(i_0)}, ct_1^{(i_1)}, \ldots, ct_L^{(i_L)})$ generated by Enc.

Definition 8 (Ciphertext Unlinkability for Multi-Hop PRE). *A multi-hop FPRE scheme* mFPRE *has ciphertext unlinkability (*CUL*), if for any PPT adversary* $\mathcal{A}$ *and any polynomial* $\mathfrak{n}$*, it holds that* $\mathsf{Adv}^{\mathsf{CUL}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}(\lambda) := \left|\Pr[\mathsf{Exp}^{\mathsf{CUL}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}} \Rightarrow 1] - \frac{1}{2}\right| \leq \mathsf{negl}(\lambda)$*, where the experiment* $\mathsf{Exp}^{\mathsf{CUL}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ *is defined in Fig. 8.*

In the full version [20], we give some explanations of the CUL security definition and discuss the trivial attacks **TA1″**-**TA2″**. We note that CUL security is similar to the SH security (cf. Definition 6) as they both capture the indistinguishability of re-encrypted ciphertexts and freshly generated ciphertexts. However, CUL security is defined in a much more realistic setting compared with the SH security: CUL considers a setting of multiple users while SH deals with only two users, and moreover, CUL protects the unlinkability of a chain of L re-encrypted ciphertexts with L the maximum level of mFPRE, while SH considers only chains of two ciphertexts. Nevertheless, in the full version [20], we show that the CUL security is implied by the SH + CPA security.

Remark 3 (Post-compromise Security). In [7], Davidson et al. proposed post-compromise security (PCS) for PRE, which considers the scenario where PRE serves for key rotation and guarantees that security still exists after the compromise of past secret keys. More concretely, suppose that Alice has stored some encrypted data and wants to update her public key from pk to pk'. To this end, she can generate an update token (i.e., a re-encryption key from pk to pk'), and re-encrypts the encrypted data using the token. In such scenario, PCS ensures that an adversary cannot distinguish which of two adversarially-chosen ciphertexts a re-encryption was created from, even when given the old secret key (i.e., the sk corresponding to pk) and the update token. Davidson et al. [7] also discussed the relations between PCS and other security notions of PRE, and proved that HRA together with SH imply PCS for (non-fine-grained) PRE.

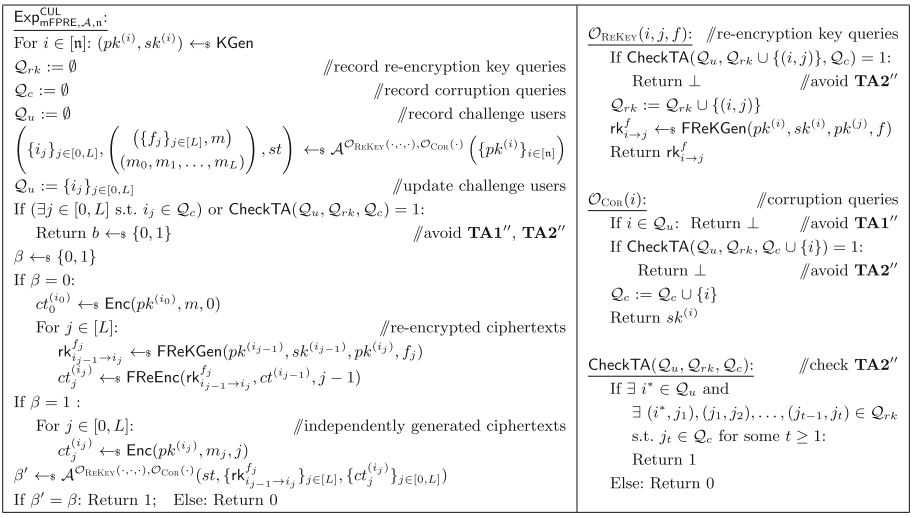

Fig. 8. The Ciphertext Unlinkability security experiment $\mathsf{Exp}^{\mathsf{CUL}}_{\mathsf{mFPRE},\mathcal{A},\mathfrak{n}}$ for mFPRE.

Following [7], we can extend PCS for our multi-hop FPRE, by requiring the indistinguishability between fine-grained re-encryptions of two adversarially chosen ciphertexts, even if the adversary can obtain the old secret key and the fine-grained re-encryption key used to perform the re-encryption. Moreover, similar to [7], we can also show that $\mathsf{HRA} + \mathsf{SH} \Rightarrow$ PCS holds for our multi-hop FPRE. The formalization of PCS and the proof of $\mathsf{HRA} + \mathsf{SH} \Rightarrow$ PCS for multi-hop FPRE are straightforward based on [7], and we will not elaborate on them.

4 Constructions of Multi-hop Fine-Grained PRE Scheme

In this section, we present two constructions of multi-hop fine-grained PRE (mFPRE) schemes, including a CPA secure scheme mFPRE_1 and an HRA secure scheme mFPRE_2, from the LWE assumptions.

4.1 The CPA Secure Multi-hop FPRE Scheme mFPRE_1

Parameters. Let $\mathsf{pp}_{\mathsf{LWE}} = (p, q, n, N, L, \ell, \gamma, \Delta, \chi)$ be LWE-related parameters that meet the following conditions:

- $p, q, n, N, L, \ell, \gamma, \Delta \in \mathbb{N}$, where $q := p^2$, $\gamma \geq O(\sqrt{n \log q}) \cdot \omega(\sqrt{\log n})$;
- χ is a B-bounded distribution, where B satisfies $\gamma \cdot \omega(\log n) \leq B < \min\{p/2, q/(10N)\}$ and $(nB + NB + \ell\Delta)^L B < \min\{p/2, q/(10N)\}$.

More precisely, we describe two settings of parameter in Table 2, one for constant hops ($L = c$) and under polynomial modulus q, while the other for sub-linear

Table 2. Concrete parameters setting, where λ denotes the security parameter and c denotes an arbitrary constant.

Parameters	p	q	n	N	L	ℓ	γ	Δ	B
Settings (L = constant)	λ^{2c+1}	λ^{4c+2}	λ	λ	c	λ	$\sqrt{\lambda}(\log\lambda)^2$	λ	$\sqrt{\lambda}(\log\lambda)^4$
Settings (L = sub-linear)	$2^{\sqrt{\lambda}}$	$2^{2\sqrt{\lambda}}$	λ	λ	$c\cdot\sqrt[3]{\lambda}$	λ	$\sqrt{\lambda}(\log\lambda)^2$	λ	$\sqrt{\lambda}(\log\lambda)^4$

hops ($L = c\cdot\sqrt[3]{\lambda}$) under sub-exponential modulus q. For simplicity, we assume that all algorithms of our scheme mFPRE_1 take $\mathsf{pp}_{\mathsf{LWE}}$ as an implicit input.

Bounded Linear Function Family. The message space is $\mathcal{M} := \mathbb{Z}_p^\ell$. Define the family of bounded linear functions $\mathcal{F}_{\mathrm{lin}}$ from $\mathcal{M}$ to $\mathcal{M}$ over $\mathbb{Z}_p$ as follows:

$$\mathcal{F}_{\mathrm{lin}} = \left\{ \begin{array}{l} f_{\mathbf{M}} : \mathbb{Z}_p^\ell \to \mathbb{Z}_p^\ell \\ \qquad \mathbf{m} \mapsto \mathbf{M}\cdot\mathbf{m} \bmod p \end{array} \;\middle|\; \mathbf{M}\in\mathbb{Z}_p^{\ell\times\ell}, \|\mathbf{M}\|_\infty \le \Delta \right\}. \tag{9}$$

LWE-Based Multi-hop FPRE Scheme mFPRE_1. Let $\mathsf{TrapGen}$, $\mathsf{SamplePre}, \mathsf{Invert}$ be the PPT algorithms introduced in [1,11,16]. Our LWE-based multi-hop FPRE scheme $\mathsf{mFPRE}_1 = (\mathsf{KGen}, \mathsf{FReKGen}, \mathsf{Enc}, \mathsf{FReEnc}, \mathsf{Dec})$ for the bounded linear function family $\mathcal{F}_{\mathrm{lin}}$ defined in (9) is shown in Fig. 9.

$\underline{(pk, sk) \leftarrow_{\$} \mathsf{KGen}:}$
$(\overline{\mathbf{A}} \in \mathbb{Z}_q^{N\times n}, \mathbf{T}) \leftarrow \mathsf{TrapGen}(1^n, 1^N)$
$\underline{\mathbf{A}} \leftarrow_{\$} \mathbb{Z}_q^{\ell\times n}$
$pk := \mathbf{A} = \binom{\overline{\mathbf{A}}}{\underline{\mathbf{A}}} \in \mathbb{Z}_q^{(N+\ell)\times n}$
$sk := \mathbf{T}$
Return (pk, sk)

$\underline{\mathsf{rk}_{i\to j}^{f_{\mathbf{M}}} \leftarrow_{\$} \mathsf{FReKGen}(pk^{(i)} = \mathbf{A}^{(i)}, sk^{(i)} = \mathbf{T}^{(i)}, pk^{(j)} = \mathbf{A}^{(j)}, f_{\mathbf{M}} \in \mathcal{F}_{\mathrm{lin}}):}$
$\mathbf{S} \leftarrow_{\$} \chi^{n\times n}, \mathbf{E} \leftarrow_{\$} \chi^{(N+\ell)\times n}$
Parse $\mathbf{A}^{(i)} = \binom{\overline{\mathbf{A}}^{(i)}}{\underline{\mathbf{A}}^{(i)}}$
$\mathbf{R} \in \mathbb{Z}^{(N+\ell)\times N} \leftarrow_{\$} \mathsf{SamplePre}\left(\mathbf{T}^{(i)}, \overline{\mathbf{A}}^{(i)}, \mathbf{A}^{(j)}\mathbf{S} + \mathbf{E} - \binom{\mathbf{0}}{\mathbf{M}}\underline{\mathbf{A}}^{(i)}, \gamma\right)$
$\mathsf{rk}_{i\to j}^{f_{\mathbf{M}}} := \left(\mathbf{R} \;\middle|\; \begin{matrix}\mathbf{0}\\ \mathbf{M}\end{matrix}\right) \in \mathbb{Z}_p^{(N+\ell)\times(N+\ell)}$ // $\mathbf{M}$ is the description of $f_{\mathbf{M}}$
Return $\mathsf{rk}_{i\to j}^{f_{\mathbf{M}}}$

$\underline{ct_v \leftarrow_{\$} \mathsf{Enc}(pk = \mathbf{A}, \mathbf{m} \in \mathcal{M}, v \in [0, L]):}$
$\mathbf{s} \leftarrow_{\$} \chi^n, \mathbf{e} \leftarrow_{\$} \chi^{N+\ell}$
$ct_v := \mathbf{A}\mathbf{s} + \mathbf{e} + \binom{\mathbf{0}}{p\mathbf{m}} \in \mathbb{Z}_q^{N+\ell}$

$\underline{ct_{v+1}^{(j)} \leftarrow \mathsf{FReEnc}(\mathsf{rk}_{i\to j}^{f_{\mathbf{M}}} \in \mathbb{Z}_p^{(N+\ell)\times(N+\ell)}, ct_v^{(i)} \in \mathbb{Z}_q^{N+\ell}, v \in [0, L-1]):}$
$ct_{v+1}^{(j)} := \mathsf{rk}_{i\to j}^{f_{\mathbf{M}}} \cdot ct_v^{(i)} \in \mathbb{Z}_q^{N+\ell}$
Return $ct_{v+1}^{(j)}$

$\underline{\mathbf{m} \leftarrow \mathsf{Dec}(sk = \mathbf{T}, ct \in \mathbb{Z}_q^{N+\ell}):}$
Parse $ct = \binom{\overline{ct} \in \mathbb{Z}_q^N}{\underline{ct} \in \mathbb{Z}_q^\ell}$
$(\mathbf{s}, \overline{\mathbf{e}}) \leftarrow_{\$} \mathsf{Invert}(\mathbf{T}, \overline{ct})$
$\tilde{\mathbf{m}} = (\tilde{m}_1, \ldots, \tilde{m}_\ell) := \underline{ct} - \underline{\mathbf{A}}\mathbf{s}$
For $i \in [\ell]$: $m_i := \lceil \tilde{m}_i / p \rfloor$
Return $\mathbf{m} = (m_1, m_2, \ldots, m_\ell)$

Fig. 9. The LWE-based Multi-Hop FPRE scheme mFPRE_1 for $\mathcal{F}_{\mathrm{lin}}$.

Correctness. Let $pk = \mathbf{A}$ and $sk = \mathbf{T}$. For a v-level ciphertext ct_v generated by $\mathsf{Enc}(pk, \mathbf{m}, v)$, we have $ct_v = \binom{\overline{ct_v}}{\underline{ct_v}} = \binom{\overline{\mathbf{A}}\mathbf{s}+\overline{\mathbf{e}}}{\underline{\mathbf{A}}\mathbf{s}+\underline{\mathbf{e}}+p\mathbf{m}}$, where $\mathbf{e} = \binom{\overline{\mathbf{e}}}{\underline{\mathbf{e}}} \leftarrow_{\$} \chi^{N+\ell}$ and the upper part is an LWE instance of $\overline{\mathbf{A}}$. Since $\overline{\mathbf{e}}$ is B-bounded with $B < q/(10N)$, $\|\overline{\mathbf{e}}\| \le \sqrt{N}\,\|\overline{\mathbf{e}}\|_\infty \le \sqrt{N}B < q/(10\cdot\sqrt{N})$. Then by the property of Invert (cf. the full version [20]), $(\mathbf{s}, \overline{\mathbf{e}})$ can be correctly recovered via $(\mathbf{s}, \overline{\mathbf{e}}) \leftarrow \mathsf{Invert}(\mathbf{T}, \overline{ct_v})$. Thus according to the decryption algorithm $\mathsf{Dec}(sk, ct_v)$, we get $\tilde{\mathbf{m}} = \underline{ct_v} - \underline{\mathbf{A}}\mathbf{s} = \underline{\mathbf{e}} + p\mathbf{m}$, and by parsing $\underline{\mathbf{e}} = (e_1, \ldots, e_\ell)^\top$, we have that $\tilde{m}_i = e_i + pm_i$ for all

$i \in [\ell]$. Moreover, since $\underline{\mathbf{e}}$ is B-bounded with $B < p/2$, each $|e_i| \leq B < p/2$. Consequently, $\lceil \tilde{m}_i/p \rfloor = m_i$ and Dec can recover $\mathbf{m}$ correctly from ct_v.

Fine-Grained L-Hop Correctness. For $ct_0^{(i)} \xrightarrow{\mathsf{rk}_{i\to j}^{f_{\mathbf{M}_1}}} ct_1^{(j)}$, where $ct_0^{(i)} \leftarrow_{\$} \mathsf{Enc}(pk^{(i)}, \mathbf{m}, 0)$, $\mathsf{rk}_{i\to j}^{f_{\mathbf{M}_1}} \leftarrow_{\$} \mathsf{FReKGen}(pk^{(i)}, sk^{(i)}, pk^{(j)}, f_{\mathbf{M}_1})$ and $ct_1^{(j)} \leftarrow_{\$} \mathsf{FReEnc}(\mathsf{rk}_{i\to j}^{f_{\mathbf{M}_1}}, ct_0^{(i)}, 0)$, we will show that the decryption of $ct_1^{(j)}$ results in $f_{\mathbf{M}_1}(\mathbf{m}) = \mathbf{M}_1\mathbf{m}$. More precisely, let $\mathsf{rk}_{i\to j}^{\mathbf{M}_1} := \left(\mathbf{R}_1 \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}_1\end{smallmatrix}\right)$, we have

$$\begin{aligned} ct_1^{(j)} &:= \left(\mathbf{R}_1 \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}_1\end{smallmatrix}\right) \cdot ct_0^{(i)} = \left(\mathbf{R}_1 \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}_1\end{smallmatrix}\right) \cdot \left(\begin{pmatrix}\overline{\mathbf{A}}^{(i)}\\ \underline{\mathbf{A}}^{(i)}\end{pmatrix}\mathbf{s}_0 + \begin{pmatrix}\overline{\mathbf{e}_0}\\ \underline{\mathbf{e}_0}\end{pmatrix} + \begin{pmatrix}\mathbf{0}\\ p\mathbf{m}\end{pmatrix}\right) \\ &= \left(\mathbf{R}_1\overline{\mathbf{A}}^{(i)} + \begin{pmatrix}\mathbf{0}\\ \mathbf{M}_1\end{pmatrix}\underline{\mathbf{A}}^{(i)}\right)\cdot \mathbf{s}_0 + \mathbf{R}_1\overline{\mathbf{e}_0} + \begin{pmatrix}\mathbf{0}\\ \mathbf{M}_1\underline{\mathbf{e}_0}\end{pmatrix} + \begin{pmatrix}\mathbf{0}\\ p\cdot\mathbf{M}_1\mathbf{m}\end{pmatrix} \\ &= \left(\mathbf{A}^{(j)}\mathbf{S} + \mathbf{E}\right)\cdot\mathbf{s}_0 + \mathbf{R}_1\overline{\mathbf{e}_0} + \begin{pmatrix}\mathbf{0}\\ \mathbf{M}_1\underline{\mathbf{e}_0}\end{pmatrix} + \begin{pmatrix}\mathbf{0}\\ p\cdot\mathbf{M}_1\mathbf{m}\end{pmatrix} \\ &= \mathbf{A}^{(j)}\underbrace{\mathbf{S}\mathbf{s}_0}_{:=\mathbf{s}_1} + \underbrace{\mathbf{E}\mathbf{s}_0 + \mathbf{R}_1\overline{\mathbf{e}_0} + \begin{pmatrix}\mathbf{0}\\ \mathbf{M}_1\underline{\mathbf{e}_0}\end{pmatrix}}_{:=\mathbf{e}_1} + \begin{pmatrix}\mathbf{0}\\ p\cdot \underbrace{\mathbf{M}_1\mathbf{m}}_{=f_{\mathbf{M}_1}(\mathbf{m})}\end{pmatrix}, \end{aligned} \tag{10}$$

where $\mathbf{s}_0 \leftarrow_{\$} \chi^n$, $\mathbf{e}_0 = \begin{pmatrix}\overline{\mathbf{e}_0}\\ \underline{\mathbf{e}_0}\end{pmatrix} \leftarrow_{\$} \chi^{N+\ell}$, $\mathbf{S} \leftarrow_{\$} \chi^{n\times n}$, $\mathbf{E} \leftarrow_{\$} \chi^{(N+\ell)\times n}$. Here the second last equality follows from the fact that $\mathbf{R}_1$ generated by $\mathbf{R}_1 \leftarrow_{\$} \mathsf{SamplePre}(\mathbf{T}^{(i)}, \overline{\mathbf{A}}^{(i)}, \mathbf{A}^{(j)}\mathbf{S}+\mathbf{E}-\begin{pmatrix}\mathbf{0}\\ \mathbf{M}_1\end{pmatrix}\underline{\mathbf{A}}^{(i)}, \gamma)$ satisfies $\mathbf{R}_1\overline{\mathbf{A}}^{(i)} = \mathbf{A}^{(j)}\mathbf{S}+\mathbf{E}-\begin{pmatrix}\mathbf{0}\\ \mathbf{M}_1\end{pmatrix}\underline{\mathbf{A}}^{(i)}$ and $\|\mathbf{R}_1\|_\infty \leq \gamma\cdot\omega(\log n)$ according to the property of $\mathsf{SamplePre}$ (cf. the full version [20]). Besides, $\|\mathbf{R}_1\|_\infty \leq \gamma\cdot\omega(\log n)$ implies that $\|\mathbf{R}_1\|_\infty \leq B$ due to $\gamma\cdot\omega(\log n) \leq B$. Now that $\mathbf{S}, \mathbf{E}, \mathbf{R}_1, \mathbf{s}_0, \mathbf{e}_0$ are all B-bounded and $\mathbf{M}_1$ is Δ-bounded, so we have $\|\mathbf{s}_1\|_\infty \leq nB^2$ and $\|\mathbf{e}_1\|_\infty \leq (nB + NB + \ell\Delta)B < \min\{p/2, q/(10N)\}$. Then by a similar argument as that for correctness, since $\|\mathbf{e}_1\|_\infty < q/(10N)$ and $\|\mathbf{e}_1\|_\infty < p/2$, Dec recovers $f_{\mathbf{M}_1}(\mathbf{m}) = \mathbf{M}_1\mathbf{m}$ from $ct_1^{(j)}$.

Next suppose that $ct_1^{(j)}$ is further re-encrypted to $ct_2^{(k)}$, i.e., $ct_1^{(j)} \xrightarrow{\mathsf{rk}_{j\to k}^{f_{\mathbf{M}_2}}} ct_2^{(k)}$, where $\mathsf{rk}_{j\to k}^{f_{\mathbf{M}_2}} \leftarrow_{\$} \mathsf{FReKGen}(pk^{(j)}, sk^{(j)}, pk^{(k)}, f_{\mathbf{M}_2})$ and $ct_2^{(k)} \leftarrow_{\$} \mathsf{FReEnc}(\mathsf{rk}_{j\to k}^{f_{\mathbf{M}_2}}, ct_1^{(j)}, 1)$, we will show that the decryption of $ct_2^{(k)}$ results in $f_{\mathbf{M}_2}(f_{\mathbf{M}_1}(\mathbf{m})) = \mathbf{M}_2\cdot\mathbf{M}_1\cdot\mathbf{m}$. By a similar analysis as above, let $\mathsf{rk}_{j\to k}^{f_{\mathbf{M}_2}} := \left(\mathbf{R}_2 \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}_2\end{smallmatrix}\right)$, we have

$$\begin{aligned} ct_2^{(k)} &:= \left(\mathbf{R}_2 \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}_2\end{smallmatrix}\right)\cdot ct_1^{(j)} = \left(\mathbf{R}_2 \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}_2\end{smallmatrix}\right)\cdot\left(\begin{pmatrix}\overline{\mathbf{A}}^{(j)}\\ \underline{\mathbf{A}}^{(j)}\end{pmatrix}\mathbf{s}_1 + \begin{pmatrix}\overline{\mathbf{e}_1}\\ \underline{\mathbf{e}_1}\end{pmatrix} + \begin{pmatrix}\mathbf{0}\\ p\mathbf{M}_1\mathbf{m}\end{pmatrix}\right) \\ &= \mathbf{A}^{(k)}\underbrace{\mathbf{S}\mathbf{s}_1}_{:=\mathbf{s}_2} + \underbrace{\mathbf{E}\mathbf{s}_1 + \mathbf{R}_2\overline{\mathbf{e}_1} + \begin{pmatrix}\mathbf{0}\\ \mathbf{M}_2\underline{\mathbf{e}_1}\end{pmatrix}}_{:=\mathbf{e}_2} + \begin{pmatrix}\mathbf{0}\\ p\cdot \underbrace{\mathbf{M}_2\mathbf{M}_1\mathbf{m}}_{=f_{\mathbf{M}_2}(f_{\mathbf{M}_1}(\mathbf{m}))}\end{pmatrix}, \end{aligned}$$

where $\mathbf{S} \leftarrow_{\$} \chi^{n\times n}$ and $\mathbf{E} \leftarrow_{\$} \chi^{(N+\ell)\times n}$. Similarly, we know that $\mathbf{S}, \mathbf{E}, \mathbf{R}_2$ are B-bounded and $\mathbf{M}_2$ is Δ-bounded. Together with the fact that $\|\mathbf{s}_1\|_\infty \leq nB^2 \leq (nB + NB + \ell\Delta)B$ and $\|\mathbf{e}_1\|_\infty \leq (nB + NB + \ell\Delta)B$, it follows that $\|\mathbf{s}_2\|_\infty \leq (nB + NB + \ell\Delta)nB^2$ and $\|\mathbf{e}_2\|_\infty \leq (nB + NB + \ell\Delta)^2 B < \min\{p/2, q/(10N)\}$.

Again, with a similar argument as that for correctness, the decryption algorithm Dec recovers $f_{\mathbf{M}_2}(f_{\mathbf{M}_1}(\mathbf{m})) = \mathbf{M}_2\mathbf{M}_1\mathbf{m}$ from $ct_2^{(k)}$.

As the re-encryption proceeds, after L hops of re-encryption under $f_{\mathbf{M}_1}, f_{\mathbf{M}_2}, \cdots, f_{\mathbf{M}_L}$, we get an L-level ciphertext $ct_L^{(\eta)}$ and it satisfies

$$ct_L^{(\eta)} = \mathbf{A}^{(\eta)}\mathbf{s}_L + \mathbf{e}_L + \begin{pmatrix} \mathbf{0} \\ p \cdot \underbrace{\mathbf{M}_L \cdots \mathbf{M}_2\mathbf{M}_1\mathbf{m}}_{=f_{\mathbf{M}_L}(\cdots f_{\mathbf{M}_2}(f_{\mathbf{M}_1}(\mathbf{m})))} \end{pmatrix},$$

where $\|\mathbf{s}_L\|_\infty \leq (nB + NB + \ell\Delta)^{L-1}nB^2$ and $\|\mathbf{e}_L\|_\infty \leq (nB + NB + \ell\Delta)^L B < \min\{p/2, q/(10N)\}$. Consequently, the function value $f_{\mathbf{M}_L}(\cdots f_{\mathbf{M}_2}(f_{\mathbf{M}_1}(\mathbf{m}))) = \mathbf{M}_L \cdots \mathbf{M}_2\mathbf{M}_1\mathbf{m}$ can be recovered from $ct_L^{(\eta)}$ by the decryption algorithm Dec.

Below we show the IND security and wKP security of our scheme mFPRE_1 via the following two theorems. Then together with Theorem 1 ($\mathsf{IND} + \mathsf{wKP} \Rightarrow \mathsf{CPA}$) in Subsect. 3.2, it yields the CPA security of our scheme mFPRE_1.

Theorem 3 (IND Security of mFPRE_1). *Assume that the $\mathsf{LWE}_{n,q,\chi,N+\ell}$-assumption holds, then the scheme mFPRE_1 proposed in Fig. 9 has IND security. More precisely, for any PPT adversary $\mathcal{A}$ that make at most Q_{chal} queries to $\mathcal{O}_{\mathrm{CHAL}}$, there exists a PPT algorithm $\mathcal{B}$ against the LWE assumption s.t. $\mathsf{Adv}^{\mathsf{IND}}_{\mathsf{mFPRE}_1,\mathcal{A}}(\lambda) \leq Q_{chal} \cdot \mathsf{Adv}^{\mathsf{LWE}}_{[n,q,\chi,N+\ell],\mathcal{B}}(\lambda)$.*

Proof of Theorem 3. We prove the theorem via two games G_0 and G_1.

Game G_0: This is the IND experiment (cf. Fig. 4). Let Win denote the event that $\beta' = \beta$. By definition, $\mathsf{Adv}^{\mathsf{IND}}_{\mathsf{mFPRE}_1,\mathcal{A}}(\lambda) = |\Pr_0[\mathsf{Win}] - \frac{1}{2}|$.

Let $(pk = \mathbf{A}, sk = \mathbf{T})$. In this game, the challenger chooses a random bit $\beta \leftarrow_\$ \{0,1\}$ and answers $\mathcal{A}$'s $\mathcal{O}_{\mathrm{CHAL}}$ queries $(\mathbf{m}_0, \mathbf{m}_1, v)$ with $ct_v \leftarrow_\$ \mathsf{Enc}(pk, \mathbf{m}_\beta, v)$, i.e., $ct_v := \mathbf{A}\mathbf{s} + \mathbf{e} + \binom{\mathbf{0}}{p\mathbf{m}_\beta}$ for $\mathbf{s} \leftarrow_\$ \chi^n, \mathbf{e} \leftarrow_\$ \chi^{N+\ell}$.

Game G_1: It is the same as G_0, except that, when answering $\mathcal{O}_{\mathrm{CHAL}}(\mathbf{m}_0, \mathbf{m}_1, v)$ queries, the challenger returns a uniformly sampled $ct_v \leftarrow_\$ \mathbb{Z}_q^{N+\ell}$ to $\mathcal{A}$. Clearly, now the challenge bit β is completely hidden to $\mathcal{A}$, thus $\Pr_0[\mathsf{Win}] = \frac{1}{2}$.

It is not hard to see that the $ct_v \leftarrow_\$ \mathsf{Enc}(pk, \mathbf{m}_\beta, v)$ in G_1 is indistinguishable from the $ct_v \leftarrow_\$ \mathbb{Z}_q^{N+\ell}$ in G_1 based on the LWE assumption. Formally, we have the following claim with proof appeared in the full version [20].

Claim 1. $\left|\Pr_0[\mathsf{Win}] - \Pr_1[\mathsf{Win}]\right| \leq Q_{chal} \cdot \mathsf{Adv}^{\mathsf{LWE}}_{[n,q,\chi,N+\ell],\mathcal{B}}(\lambda)$.

Finally, taking all things together, Theorem 3 follows. □

Theorem 4. (wKP Security of mFPRE_1). *Assume that the $\mathsf{LWE}_{n,q,\chi,N+\ell}$-assumption holds, then the scheme mFPRE_1 proposed in Fig. 9 has wKP security. More precisely, for any PPT adversary $\mathcal{A}$ that makes at most Q_{rk} queries to $\mathcal{O}_{\mathrm{REKEY}}$ and for any polynomial $\mathfrak{n}$, there exists a PPT algorithm $\mathcal{B}$ against the LWE assumption s.t. $\mathsf{Adv}^{\mathsf{wKP}}_{\mathsf{mFPRE}_1,\mathcal{A},\mathfrak{n}}(\lambda) \leq \mathfrak{n} \cdot nQ_{rk} \cdot \mathsf{Adv}^{\mathsf{LWE}}_{[n,q,\chi,N+\ell],\mathcal{B}}(\lambda) + \mathsf{negl}(\lambda)$.*

Proof of Theorem 4. We prove the theorem via a sequence of games G_0-G_2, where G_0 is the wKP experiment, and in G_2, $\mathcal{A}$ has a negligible advantage.

Game G_0: This is the wKP experiment (cf. Fig. 5). Let Win denote the event that $\beta' = \beta$. By definition, $\mathsf{Adv}^{\mathsf{wKP}}_{\mathsf{mFPRE}_1,\mathcal{A},\mathfrak{n}}(\lambda) = |\Pr_0[\mathsf{Win}] - \frac{1}{2}|$.

Let $pk^{(i)} = \mathbf{A}^{(i)}, sk^{(i)} = \mathbf{T}^{(i)}$ denote the public key and secret key of user $i \in [0, \mathfrak{n}]$. In this game, the challenger chooses a random bit $\beta \leftarrow_\$ \{0,1\}$ and answers $\mathcal{A}$'s $\mathcal{O}_{\text{ReKey}}$ queries ($j \in [\mathfrak{n}], f_{\mathbf{M}} \in \mathcal{F}_{\text{lin}}$) as follows:

- If $\beta = 0$, the challenger returns $\mathsf{rk}^{f_{\mathbf{M}}}_{0\to j} \leftarrow_\$ \mathsf{FReKGen}(\mathbf{A}^{(0)}, \mathbf{T}^{(0)}, \mathbf{A}^{(j)}, f_{\mathbf{M}})$. More precisely, it samples $\mathbf{S} \leftarrow_\$ \chi^{n\times n}, \mathbf{E} \leftarrow_\$ \chi^{(N+\ell)\times n}$, $\mathbf{R} \leftarrow_\$ \mathsf{SamplePre}(\mathbf{T}^{(0)}, \overline{\mathbf{A}}^{(0)}, \mathbf{A}^{(j)}\mathbf{S} + \mathbf{E} - \binom{\mathbf{0}}{\mathbf{M}}\underline{\mathbf{A}}^{(0)}, \gamma)$, and returns $\mathsf{rk}^{f_{\mathbf{M}}}_{0\to j} := \left(\mathbf{R} \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}\end{smallmatrix}\right)$ to $\mathcal{A}$.
- If $\beta = 1$, the challenger invokes $\mathsf{rk}^{f_{\mathbf{M}}}_{0\to j} \leftarrow_\$ \mathsf{FReKGen}^*(\mathbf{A}^{(0)}, \mathbf{A}^{(j)}, f_{\mathbf{M}})$ which is defined as $\mathsf{FReKGen}^* : \mathbf{R} \leftarrow_\$ D_{\mathbb{Z}^{(N+\ell)\times N},\gamma}$ and $\mathsf{rk}^{f_{\mathbf{M}}}_{0\to j} := \left(\mathbf{R} \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}\end{smallmatrix}\right)$. Then the challenger returns $\mathsf{rk}^{f_{\mathbf{M}}}_{0\to j}$ to $\mathcal{A}$.

Game $\mathsf{G}_{0.t}, t \in [0, \mathfrak{n}]$: It is the same as G_0, except for the reply to $\mathcal{A}$'s $\mathcal{O}_{\text{ReKey}}(j, f_{\mathbf{M}})$ query when $\beta = 0$:

- For $j \le t$, the challenger uniformly samples $\mathbf{U} \leftarrow_\$ \mathbb{Z}_q^{(N+\ell)\times n}$ and invokes $\mathbf{R} \leftarrow_\$ \mathsf{SamplePre}(\mathbf{T}^{(0)}, \overline{\mathbf{A}}^{(0)}, \mathbf{U}, \gamma)$ to get $\mathsf{rk}^{f_{\mathbf{M}}}_{0\to j} := \left(\mathbf{R} \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}\end{smallmatrix}\right)$.
- For $j > t$, the challenger answers the query just like G_0, that is, $\mathbf{R} \leftarrow_\$ \mathsf{SamplePre}\left(\mathbf{T}^{(0)}, \overline{\mathbf{A}}^{(0)}, \mathbf{A}^{(j)}\mathbf{S} + \mathbf{E} - \binom{\mathbf{0}}{\mathbf{M}}\underline{\mathbf{A}}^{(0)}, \gamma\right)$ with $\mathbf{S} \leftarrow_\$ \chi^{n\times n}, \mathbf{E} \leftarrow_\$ \chi^{(N+\ell)\times n}$.

Clearly, $\mathsf{G}_{0.0}$ is identical to G_0. Thus, we have $\Pr_0[\mathsf{Win}] = \Pr_{0.0}[\mathsf{Win}]$.

Below we show the computational indistinguishability between $\mathsf{G}_{0.t-1}$ and $\mathsf{G}_{0.t}$ based on the LWE assumption.

Claim 2. For all $t \in [\mathfrak{n}]$, $|\Pr_{0.t-1}[\mathsf{Win}] - \Pr_{0.t}[\mathsf{Win}]| \le nQ_{rk} \cdot \mathsf{Adv}^{\mathsf{LWE}}_{[n,q,\chi,N+\ell],\mathcal{B}}(\lambda)$.

Proof. Firstly, we construct a PPT adversary $\mathcal{B}'$ against the nQ_{rk}-$\mathsf{LWE}_{n,q,\chi,N+\ell}$-assumption, such that $|\Pr_{0.t-1}[\mathsf{Win}] - \Pr_{0.t}[\mathsf{Win}]| \le \mathsf{Adv}^{nQ_{rk}\text{-}\mathsf{LWE}}_{[n,q,\chi,N+\ell],\mathcal{B}'}(\lambda)$. Then by a standard hybrid argument, we have $\mathsf{Adv}^{nQ_{rk}\text{-}\mathsf{LWE}}_{[n,q,\chi,N+\ell],\mathcal{B}'}(\lambda) \le nQ_{rk} \cdot \mathsf{Adv}^{\mathsf{LWE}}_{[n,q,\chi,N+\ell],\mathcal{B}}(\lambda)$ and the claim follows.

Algorithm $\mathcal{B}'$. Given a challenge $(\mathbf{A}, \mathbf{Z})$, $\mathcal{B}'$ wants to distinguish $\mathbf{Z} = \mathbf{AS} + \mathbf{E}$ from $\mathbf{Z} \leftarrow_\$ \mathbb{Z}_q^{(N+\ell)\times nQ_{rk}}$, where $\mathbf{A} \leftarrow_\$ \mathbb{Z}_q^{(N+\ell)\times n}$, $\mathbf{S} \leftarrow_\$ \chi^{n\times nQ_{rk}}$, $\mathbf{E} \leftarrow_\$ \chi^{(N+\ell)\times nQ_{rk}}$.

$\mathcal{B}'$ is constructed by simulating $\mathsf{G}_{0.t-1}/\mathsf{G}_{0.t}$ for $\mathcal{A}$ as follows. Firstly, $\mathcal{B}'$ sets $pk^{(t)} := \mathbf{A}^{(t)} := \mathbf{A}$ directly for the user t, and invokes KGen honestly to generate $(pk^{(i)}, sk^{(i)})$ for all other users $i \in [0, \mathfrak{n}] \setminus \{t\}$. In particular, $\mathcal{B}'$ owns $sk^{(0)} = \mathbf{T}^{(0)}$. $\mathcal{B}'$ sends $\{pk^{(i)}\}_{i\in[0,\mathfrak{n}]}$ to $\mathcal{A}$. Then $\mathcal{B}'$ chooses a random bit $\beta \leftarrow_\$ \{0,1\}$ and parses $\mathbf{Z} = (\mathbf{Z}_1 \mid \cdots \mid \mathbf{Z}_{Q_{rk}}) \in \mathbb{Z}_q^{(N+\ell)\times nQ_{rk}}$ with each $\mathbf{Z}_k \in \mathbb{Z}_q^{(N+\ell)\times n}$ for $k \in [Q_{chal}]$. On receiving an $\mathcal{O}_{\text{ReKey}}(j \in [\mathfrak{n}], f_{\mathbf{M}})$ query from $\mathcal{A}$, if $\beta = 1$, $\mathcal{B}'$ invokes $\mathsf{FReKGen}^*$ to get $\mathsf{rk}^{f_{\mathbf{M}}}_{0\to j}$ and returns it to $\mathcal{A}$, the same as $\mathsf{G}_{0.t-1}$ and $\mathsf{G}_{0.t}$. Otherwise, i.e., $\beta = 0$, $\mathcal{B}'$ answers the $\mathcal{O}_{\text{ReKey}}(j \in [\mathfrak{n}], f_{\mathbf{M}})$ query in the following way:

- For $j \leq t-1$, $\mathcal{B}'$ samples $\mathbf{U} \leftarrow_{\$} \mathbb{Z}_q^{(N+\ell)\times n}$ and invokes $\mathbf{R} \leftarrow_{\$} \mathsf{SamplePre}(\mathbf{T}^{(0)}, \overline{\mathbf{A}}^{(0)}, \mathbf{U}, \gamma)$ to get $\mathsf{rk}_{0\to j}^{f_{\mathbf{M}}} := \left(\mathbf{R} \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}\end{smallmatrix}\right)$, the same as $\mathsf{G}_{0.t-1}$ and $\mathsf{G}_{0.t}$.
- For $j = t$, suppose that this is the k-th $\mathcal{O}_{\mathrm{ReKey}}$ query with $k \in [Q_{rk}]$, $\mathcal{B}'$ makes use of $\mathbf{Z}_k$ to invoke $\mathbf{R} \leftarrow_{\$} \mathsf{SamplePre}(\mathbf{T}^{(0)}, \overline{\mathbf{A}}^{(0)}, \mathbf{Z}_k - \binom{\mathbf{0}}{\mathbf{M}}\underline{\mathbf{A}}^{(0)}, \gamma)$ to get $\mathsf{rk}_{0\to t}^{f_{\mathbf{M}}} := \left(\mathbf{R} \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}\end{smallmatrix}\right)$.
  In the case of $\mathbf{Z} = \mathbf{AS} + \mathbf{E}$, by parsing $\mathbf{S} = (\mathbf{S}_1 \mid \cdots \mid \mathbf{S}_{Q_{rk}}) \in \mathbb{Z}_q^{n\times nQ_{rk}}$ with each $\mathbf{S}_k \in \mathbb{Z}_q^{n\times n}$ and parsing $\mathbf{E} = (\mathbf{E}_1 \mid \cdots \mid \mathbf{E}_{Q_{rk}}) \in \mathbb{Z}_q^{(N+\ell)\times nQ_{rk}}$ with each $\mathbf{E}_k \in \mathbb{Z}_q^{(N+\ell)\times n}$, we have $\mathbf{Z}_k = \mathbf{AS}_k + \mathbf{E}_k = \mathbf{A}^{(t)}\mathbf{S}_k + \mathbf{E}_k$ for $\mathbf{S}_k \leftarrow_{\$} \chi^{n\times n}$ and $\mathbf{E}_k \leftarrow_{\$} \chi^{(N+\ell)\times n}$, and consequently, $\mathcal{B}'$'s simulation is identical to $\mathsf{G}_{0.t-1}$.
  In the case of $\mathbf{Z} \leftarrow_{\$} \mathbb{Z}_q^{(N+\ell)\times nQ_{rk}}$, we have that $\mathbf{Z}_k$ is uniformly distributed over $\mathbb{Z}_q^{(N+\ell)\times n}$, so $\mathcal{B}'$'s simulation is identical to $\mathsf{G}_{0.t}$.
- For $j > t$, $\mathcal{B}'$ samples $\tilde{\mathbf{S}} \leftarrow_{\$} \chi^{n\times n}, \tilde{\mathbf{E}} \leftarrow_{\$} \chi^{(N+\ell)\times n}$, $\mathbf{R} \leftarrow_{\$} \mathsf{SamplePre}(\mathbf{T}^{(0)}, \overline{\mathbf{A}}^{(0)}, \mathbf{A}^{(j)}\tilde{\mathbf{S}} + \tilde{\mathbf{E}} - \binom{\mathbf{0}}{\mathbf{M}}\underline{\mathbf{A}}^{(0)}, \gamma)$ to get $\mathsf{rk}_{0\to j}^{f_{\mathbf{M}}} := \left(\mathbf{R} \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}\end{smallmatrix}\right)$, the same as $\mathsf{G}_{0.t-1}$ and $\mathsf{G}_{0.t}$.

Finally, $\mathcal{B}'$ receives β' from $\mathcal{A}$ and outputs 1 to its own challenger if and only if $\beta' = \beta$.

Now we analyze the advantage of $\mathcal{B}'$. Overall, $\mathcal{B}'$ simulates $\mathsf{G}_{0.t-1}$ for $\mathcal{A}$ in the case $\mathbf{Z} = \mathbf{AS} + \mathbf{E}$ while simulates $\mathsf{G}_{0.t}$ for $\mathcal{A}$ in the case $\mathbf{Z} \leftarrow_{\$} \mathbb{Z}_q^{(N+\ell)\times nQ_{rk}}$. Thus $\mathcal{B}'$ successfully distinguishes $\mathbf{Z} = \mathbf{AS} + \mathbf{E}$ from $\mathbf{Z} \leftarrow_{\$} \mathbb{Z}_q^{(N+\ell)\times nQ_{rk}}$ as long as the probability that $\beta' = \beta$ in $\mathsf{G}_{0.t-1}$ differs non-negligibly from that in $\mathsf{G}_{0.t}$. Consequently, we have $\mathsf{Adv}^{nQ_{rk}\text{-}\mathsf{LWE}}_{[n,q,\chi,N+\ell],\mathcal{B}'}(\lambda) \geq \left|\Pr_{0.t-1}[\mathsf{Win}] - \Pr_{0.t}[\mathsf{Win}]\right|$. ▌

Game G_1: It's the same as G_0, except for the reply to $\mathcal{A}$'s $\mathcal{O}_{\mathrm{ReKey}}(j, f_{\mathbf{M}})$ query when $\beta = 0$:

- For all $j \in [\mathfrak{n}]$, the challenger samples $\mathbf{U} \leftarrow_{\$} \mathbb{Z}_q^{(N+\ell)\times n}$ and uses $\mathbf{U}$ to invoke $\mathbf{R} \leftarrow_{\$} \mathsf{SamplePre}(\mathbf{T}^{(0)}, \overline{\mathbf{A}}^{(0)}, \mathbf{U}, \gamma)$, and returns $\mathsf{rk}_{0\to j}^{f_{\mathbf{M}}} := \left(\mathbf{R} \;\middle|\; \begin{smallmatrix}\mathbf{0}\\ \mathbf{M}\end{smallmatrix}\right)$ to $\mathcal{A}$.

Clearly, $\mathsf{G}_1 = \mathsf{G}_{0.\mathfrak{n}}$ and $\Pr_1[\mathsf{Win}] = \Pr_{0.\mathfrak{n}}[\mathsf{Win}]$.

Game G_2: It's the same as G_1, except for the reply to $\mathcal{A}$'s $\mathcal{O}_{\mathrm{ReKey}}(j, f_{\mathbf{M}})$ query when $\beta = 0$. The challenger samples $\mathbf{R}$ by $\mathbf{R} \leftarrow_{\$} D_{\mathbb{Z}^{(N+\ell)\times N},\gamma}$, instead of invoking $\mathbf{R} \leftarrow_{\$} \mathsf{SamplePre}\left(\mathbf{T}^{(0)}, \overline{\mathbf{A}}^{(0)}, \mathbf{U} \leftarrow_{\$} \mathbb{Z}_q^{(N+\ell)\times n}, \gamma\right)$ as in G_1.

Since $\gamma \geq O(\sqrt{n \log q}) \cdot \omega(\sqrt{\log n})$, according to the indistinguishability of preimage-sampling $\mathsf{SamplePre}$ (as recalled in the full version [20]), G_2 is statistically close to G_1. Thus we have $\left|\Pr_1[\mathsf{Win}] - \Pr_2[\mathsf{Win}]\right| \leq \mathsf{negl}(\lambda)$.

Finally, note that in G_2, the challenger's reply to $\mathcal{A}$'s $\mathcal{O}_{\mathrm{ReKey}}$ query in the case $\beta = 0$ is identical to that in the case $\beta = 1$. Thus the challenge bit β is completely hidden to $\mathcal{A}$, and we have $\Pr_2[\mathsf{Win}] = \frac{1}{2}$.

Taking all things together, Theorem 4 follows. □

By plugging Theorem 3 (IND security) and Theorem 4 (wKP security) into Theorem 1 (IND + wKP ⇒ CPA) in Subsect. 3.2, we have the following corollary showing the CPA security of mFPRE_1 based on the LWE assumption.

Corollary 1 (CPA Security of mFPRE_1). *Assume that the $\mathsf{LWE}_{n,q,\chi,N+\ell}$-assumption holds, then the scheme mFPRE_1 proposed in Fig. 9 is CPA secure. More precisely, for any PPT adversary $\mathcal{A}$ that makes at most Q_{rk} queries to $\mathcal{O}_{\mathrm{ReKey}}$ and forms a challenge graph G (i.e., subgraph reachable from the vertex of challenge user) in $\mathcal{G}(\mathfrak{n}, \delta, d)$, for any polynomial $\mathfrak{n}$, there exists a PPT algorithm $\mathcal{B}$ against the LWE assumption s.t.*

$$\mathsf{Adv}^{\mathsf{CPA}}_{\mathsf{mFPRE}_1,\mathcal{A},\mathfrak{n}} \leq (2\tau \cdot n\mathfrak{n}Q_{rk} + 2) \cdot \mathfrak{n}^{\sigma+\delta+1} \cdot \mathsf{Adv}^{\mathsf{LWE}}_{[n,q,\chi,N+\ell],\mathcal{B}}(\lambda) + \mathsf{negl}(\lambda),$$

where δ denotes the outdegree, d the depth, τ the pebbling time complexity and σ space complexity for the class $\mathcal{G}(\mathfrak{n}, \delta, d)$, respectively.

4.2 The HRA Secure Multi-hop FPRE Scheme mFPRE_2

Parameters. Let $\mathsf{pp}_{\mathsf{LWE}} = (p, q, n, N, L, \ell, \gamma, \Delta, \chi, \{\chi_v\}_{v\in[0,L]})$ be LWE-related parameters that meet the following conditions:

- $p, q, n, N, L, \ell, \gamma, \Delta \in \mathbb{N}$, where $q := p^2$, $\gamma \geq O(\sqrt{n \log q}) \cdot \omega(\sqrt{\log n})$;
- χ is a B-bounded distribution, where B satisfies $\gamma \cdot \omega(\log n) \leq B$.
- For each $v \in [0, L]$, χ_v is the uniform distribution over $[-B_v, B_v]$, where B_v satisfies $B_v \geq 2^{\sqrt[3]{\lambda}} \cdot (nB + NB + \ell\Delta)B_{v-1}$ for $v \geq 1$ and $B_L \leq \min\{p/4, q/(20N)\}$.

More precisely, we describe two settings of parameter in Table 3, one for constant hops ($L = c$) and the other for sub-linear hops ($L = c \cdot \sqrt[3]{\lambda}$), both under sub-exponential modulus q. For simplicity, we assume that all algorithms of our scheme mFPRE_2 take $\mathsf{pp}_{\mathsf{LWE}}$ as an implicit input.

Table 3. Concrete parameters setting, where λ denotes the security parameter and c denotes an arbitrary constant.

Parameters	p	q	n	N	L	ℓ	γ	Δ	B	B_v $(v \in [0, L])$
Settings (L = constant)	$2^{\sqrt{\lambda}}$	$2^{2\sqrt{\lambda}}$	λ	λ	c	λ	$\sqrt{\lambda}(\log\lambda)^2$	λ	$\sqrt{\lambda}(\log\lambda)^4$	$(\lambda^2 \cdot 2^{\sqrt[3]{\lambda}+1})^{v+1}$
Settings (L = sub-linear)	$2^{\lambda^{3/4}}$	$2^{2\lambda^{3/4}}$	λ	λ	$c \cdot \sqrt[3]{\lambda}$	λ	$\sqrt{\lambda}(\log\lambda)^2$	λ	$\sqrt{\lambda}(\log\lambda)^4$	$(\lambda^2 \cdot 2^{\sqrt[3]{\lambda}+1})^{v+1}$

LWE-Based Multi-hop FPRE Scheme mFPRE_2. Our LWE-based FPRE scheme $\mathsf{mFPRE}_2 = (\mathsf{KGen}, \mathsf{FReKGen}, \mathsf{Enc}, \mathsf{FReEnc}, \mathsf{Dec})$ is also for the bounded linear function family $\mathcal{F}_{\mathrm{lin}}$ defined in (9) in Subsect. 4.1, and is shown in Fig. 10.

The analysis for the correctness and fine-grained L-hop correctness of mFPRE_2 are similar to those for mFPRE_1. Due to space limitations, we postpone the formal analysis to the full version [20].

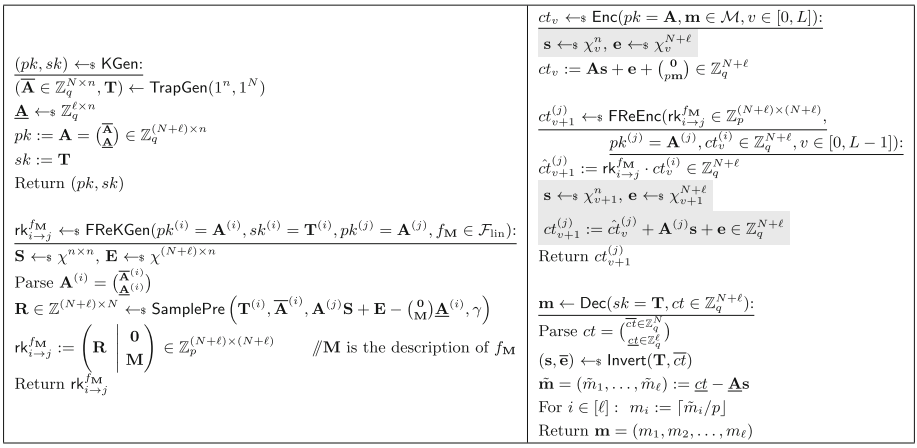

Fig. 10. The LWE-based Multi-Hop FPRE scheme mFPRE_2 for $\mathcal{F}_{\mathrm{lin}}$. For ease of reading, we emphasize different parts with the CPA secure scheme mFPRE_1 in gray boxes.

Next, we show the IND security, wKP security and SH security of mFPRE_2 via the following three theorems. Then together with Theorem 2 (IND + wKP + SH ⇒ HRA) in Subsect. 3.2, it yields the HRA security of our scheme mFPRE_2.

Theorem 5 (IND Security of mFPRE_2). *Assume that the* $\mathsf{LWE}_{n,q,\chi_i,N+\ell}$*-assumption holds for all* $i \in [0, L]$*, then the scheme* mFPRE_2 *proposed in Fig. 10 has* IND *security. More precisely, for any PPT adversary* $\mathcal{A}$ *that make at most* Q_{chal} *queries to* $\mathcal{O}_{\mathrm{CHAL}}$*, there exist PPT algorithms* $\mathcal{B}_0, \ldots, \mathcal{B}_L$ *against the LWE assumptions such that* $\mathsf{Adv}^{\mathsf{IND}}_{\mathsf{mFPRE}_2,\mathcal{A}}(\lambda) \le Q_{chal} \cdot \sum_{i=0}^{L} \mathsf{Adv}^{\mathsf{LWE}}_{[n,q,\chi_i,N+\ell],\mathcal{B}_i}(\lambda)$.

We refer to the full version [20] for the proof of Theorem 5.

Theorem 6 (wKP Security of mFPRE_2). *Assume that the* $\mathsf{LWE}_{n,q,\chi,N+\ell}$*-assumption holds, then the scheme* mFPRE_2 *proposed in Fig. 10 has* wKP *security. More precisely, for any PPT adversary* $\mathcal{A}$ *that makes at most* Q_{rk} *queries to* $\mathcal{O}_{\mathrm{REKEY}}$ *and for any polynomial* $\mathfrak{n}$*, there exists a PPT algorithm* $\mathcal{B}$ *against the LWE assumption s.t.* $\mathsf{Adv}^{\mathsf{wKP}}_{\mathsf{mFPRE}_2,\mathcal{A},\mathfrak{n}}(\lambda) \le \mathfrak{n} \cdot nQ_{rk} \cdot \mathsf{Adv}^{\mathsf{LWE}}_{[n,q,\chi,N+\ell],\mathcal{B}}(\lambda) + \mathsf{negl}(\lambda)$.

Note that the KGen and FReKGen algorithms of scheme mFPRE_2 are the same as those of mFPRE_1 in Subsect. 4.1, so does the wKP security. Consequently, the proof of Theorem 6 is identical to that for Theorem 4 and we omit it.

Theorem 7 (SH Security of mFPRE_2). *The scheme* mFPRE_2 *proposed in Fig. 10 has* SH *security. More precisely, for any (unbounded) adversary* $\mathcal{A}$*, we have* $\mathsf{Adv}^{\mathsf{SH}}_{\mathsf{mFPRE}_2,\mathcal{A}}(\lambda) \le \mathsf{negl}(\lambda)$.

We refer to the full version [20] for the proof of Theorem 7.

By plugging Theorem 5 (IND security), Theorem 6 (wKP security) and Theorem 7 (SH security) into Theorem 2 (IND + wKP + SH ⇒ HRA) in

Subsect. 3.2, we have the following corollary showing the HRA security of our scheme mFPRE_2 based on the LWE assumption.

Corollary 2 (HRA Security of mFPRE_2). *Assume that the $\mathsf{LWE}_{n,q,\chi,N+\ell}$-assumption and the $\mathsf{LWE}_{n,q,\chi_i,N+\ell}$-assumption hold for all $i \in [0, L]$, then the scheme mFPRE_2 proposed in Fig. 10 is HRA secure. More precisely, for any PPT adversary $\mathcal{A}$ that makes at most Q_{rk} queries to $\mathcal{O}_{\mathrm{ReKey}}$ and forms a challenge graph G (i.e., subgraph reachable from the vertex of challenge user) in $\mathcal{G}(\mathfrak{n}, \delta, d)$, for any polynomial $\mathfrak{n}$, there exists PPT algorithms $\mathcal{B}_0, \ldots, \mathcal{B}_L$ and $\mathcal{B}$ against the LWE assumption s.t.*

$$\mathsf{Adv}^{\mathsf{HRA}}_{\mathsf{mFPRE}_2,\mathcal{A},\mathfrak{n}} \leq \left(2\sum_{i=0}^{L} \mathsf{Adv}^{\mathsf{LWE}}_{[n,q,\chi_i,N+\ell],\mathcal{B}_i}(\lambda) + 2\tau \cdot n\mathfrak{n}Q_{rk} \cdot \mathsf{Adv}^{\mathsf{LWE}}_{[n,q,\chi,N+\ell],\mathcal{B}}(\lambda)\right) \cdot \mathfrak{n}^{\sigma+\delta+1} + \mathsf{negl}(\lambda),$$

where δ denotes the outdegree, d the depth, τ the pebbling time complexity and σ space complexity for the class $\mathcal{G}(\mathfrak{n}, \delta, d)$, respectively.

Acknowledgments. We would like to thank the reviewers for their valuable comments and helpful suggestions. Yunxiao Zhou, Shengli Liu and Shuai Han were partially supported by the National Key R&D Program of China under Grant 2022YFB2701500, National Natural Science Foundation of China (Grant Nos. 61925207, 62372292), Guangdong Major Project of Basic and Applied Basic Research (2019B030302008), and Young Elite Scientists Sponsorship Program by China Association for Science and Technology (YESS20200185).

References

1. Ajtai, M.: Generating hard instances of lattice problems (extended abstract). In: 28th ACM STOC, pp. 99–108. ACM Press, May 1996
2. Ateniese, G., Benson, K., Hohenberger, S.: Key-private proxy re-encryption. In: Fischlin, M. (ed.) CT-RSA 2009. LNCS, vol. 5473, pp. 279–294. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-00862-7_19
3. Ateniese, G., Fu, K., Green, M., Hohenberger, S.: Improved proxy re-encryption schemes with applications to secure distributed storage. In: NDSS 2005. The Internet Society, February 2005
4. Blaze, M., Bleumer, G., Strauss, M.: Divertible protocols and atomic proxy cryptography. In: Nyberg, K. (ed.) EUROCRYPT 1998. LNCS, vol. 1403, pp. 127–144. Springer, Heidelberg (1998). https://doi.org/10.1007/BFb0054122
5. Chandran, N., Chase, M., Liu, F.-H., Nishimaki, R., Xagawa, K.: Re-encryption, functional re-encryption, and multi-hop re-encryption: a framework for achieving obfuscation-based security and instantiations from lattices. In: Krawczyk, H. (ed.) PKC 2014. LNCS, vol. 8383, pp. 95–112. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-54631-0_6
6. Cohen, A.: What about Bob? The inadequacy of CPA security for proxy reencryption. In: Lin, D., Sako, K. (eds.) PKC 2019. LNCS, vol. 11443, pp. 287–316. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-17259-6_10
7. Davidson, A., Deo, A., Lee, E., Martin, K.: Strong post-compromise secure proxy re-encryption. In: Jang-Jaccard, J., Guo, F. (eds.) ACISP 2019. LNCS, vol. 11547, pp. 58–77. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21548-4_4

8. Fan, X., Liu, F.-H.: Proxy re-encryption and re-signatures from lattices. In: Deng, R.H., Gauthier-Umaña, V., Ochoa, M., Yung, M. (eds.) ACNS 2019. LNCS, vol. 11464, pp. 363–382. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21568-2_18
9. Fuchsbauer, G., Kamath, C., Klein, K., Pietrzak, K.: Adaptively secure proxy re-encryption. In: Lin, D., Sako, K. (eds.) PKC 2019. LNCS, vol. 11443, pp. 317–346. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-17259-6_11
10. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: Mitzenmacher, M. (ed.) 41st ACM STOC, pp. 169–178. ACM Press, May/June 2009
11. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Ladner, R.E., Dwork, C. (eds.) 40th ACM STOC, pp. 197–206. ACM Press, May 2008
12. Jafargholi, Z., Kamath, C., Klein, K., Komargodski, I., Pietrzak, K., Wichs, D.: Be adaptive, avoid overcommitting. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017. LNCS, vol. 10401, pp. 133–163. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63688-7_5
13. Lai, J., Huang, Z., Au, M.H., Mao, X.: Constant-size CCA-secure multi-hop unidirectional proxy re-encryption from indistinguishability obfuscation. In: Susilo, W., Yang, G. (eds.) ACISP 2018. LNCS, vol. 10946, pp. 805–812. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93638-3_49
14. Lai, J., Huang, Z., Au, M.H., Mao, X.: Constant-size CCA-secure multi-hop unidirectional proxy re-encryption from indistinguishability obfuscation. Theor. Comput. Sci. **847**, 1–16 (2020). https://www.sciencedirect.com/science/article/pii/S0304397520305302
15. Miao, P., Patranabis, S., Watson, G.J.: Unidirectional updatable encryption and proxy re-encryption from DDH. In: Boldyreva, A., Kolesnikov, V. (eds.) PKC 2023, Part II. LNCS, vol. 13941, pp. 368–398. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-31371-4_13
16. Micciancio, D., Peikert, C.: Trapdoors for lattices: simpler, tighter, faster, smaller. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 700–718. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29011-4_41
17. Phong, L.T., Wang, L., Aono, Y., Nguyen, M.H., Boyen, X.: Proxy re-encryption schemes with key privacy from LWE. Cryptology ePrint Archive, Report 2016/327 (2016). https://eprint.iacr.org/2016/327
18. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Gabow, H.N., Fagin, R. (eds.) 37th ACM STOC, pp. 84–93. ACM Press, May 2005
19. Smith, T.: DVD Jon: Buy DRM-less tracks from apple iTunes (2005). https://www.theregister.com/2005/03/18/itunes_pymusique/
20. Zhou, Y., Liu, S., Han, S.: Multi-hop fine-grained proxy re-encryption. Cryptology ePrint Archive, 2024/055 (2024). https://eprint.iacr.org/2024/055
21. Zhou, Y., Liu, S., Han, S., Zhang, H.: Fine-grained proxy re-encryption: definitions & constructions from LWE. In: Guo, J., Steinfeld, R. (eds.) ASIACRYPT 2023, Part VI. LNCS, vol. 14443, pp. 199–231. Springer, Cham (2023). https://doi.org/10.1007/978-981-99-8736-8_7