Property Preserving Symmetric Encryption Revisited

Sanjit Chatterjee[1] and M. Prem Laxman Das[2]

[1] Department of Computer Science and Automation, Indian Institute of Science
sanjit@csa.iisc.ernet.in
[2] Society for Electronic Transactions and Security, Chennai, India
prem.lax@gmail.com

Abstract. At EUROCRYPT 2012 Pandey and Rouselakis introduced the notion of property preserving symmetric encryption which enables checking for a property on plaintexts by running a public test on the corresponding ciphertexts. Their primary contributions are: (i) a separation between 'find-then-guess' and 'left-or-right' security notions; (ii) a concrete construction for left-or-right secure orthogonality testing in composite order bilinear groups.

This work undertakes a comprehensive (crypt)analysis of property preserving symmetric encryption on both these fronts. We observe that the quadratic residue based property used in their separation result is a special case of testing equality of one-bit messages, suggest a very simple and efficient deterministic encryption scheme for testing equality and show that the two security notions, find-then-guess and left-or-right, are tightly equivalent in this setting. On the other hand, the separation result easily generalizes for the equality property. So contextualized, we posit that the question of separation between security notions is property specific and subtler than what the authors envisaged; mandating further critical investigation. Next, we show that given a find-then-guess secure orthogonality preserving encryption of vectors of length 2n, there exists left-or-right secure orthogonality preserving encryption of vectors of length n, giving further evidence that find-then-guess is indeed a meaningful notion of security for property preserving encryption. Finally, we cryptanalyze the scheme for testing orthogonality. A simple distinguishing attack establishes that it is not even the weakest selective find-then-guess secure. Our main attack extracts out the subgroup elements used to mask the message vector and indicates greater vulnerabilities in the construction beyond indistinguishability. Overall, our work underlines the importance of cryptanalysis in provable security.

Keywords: bilinear pairings, property preserving encryption, predicate private encryption, symmetric key

1 Introduction

The question of constructing practical cryptographic schemes for securing data in the cloud has attracted a lot of research during the last decade. Notions like order

preserving encryption [8, 10], attribute-based encryption [26, 24, 21], functional encryption [16, 1, 15, 14, 6, 25] and format preserving encryption [7] are useful for this purpose. The notions of IBE [12, 19, 11] and public key encryption with keyword search [17, 33, 13, 34] deal with testing of equality. Homomorphic encryption too [22, 35, 23] plays an important role in cloud security. These schemes aim to achieve data privacy, user privacy, secure computation on encrypted data, etc., on the cloud.

At EUROCRYPT 2012 Pandey and Rouselakis [29] defined the notion of *property preserving symmetric encryption* (PPEnc) which can be used for data clustering [27]. This notion, the authors claim, is most useful in the symmetric key setting. A PPEnc scheme is a collection of four algorithms, namely, Setup, Encrypt, Decrypt and Test where Test is used to check whether the underlying messages satisfy a particular property or not. The authors claim that it is sufficient to consider a simpler notion called *property preserving tag* (PPTag), obtained by dropping the decryption algorithm. The standard approach is to use a semantic secure symmetric key encryption scheme to encrypt the "payload" message while the encryption algorithm of PPTag is used to create a "tag" that is used as one of the inputs to Test to publicly check whether the message satisfies the property or not. In fact a similar approach was taken in [28, 32]. Following the Bellare et al. approach for standard encryption [4, 5], they define several security notions for property preserving encryption such as find-then-guess (FtG) and left-or-right (LoR) security. However, unlike Bellare et al. [4] who showed FtG implies LoR in the ordinary symmetric key setting, [29] claims that there is a separation between FtG and LoR notions and a hierarchy among the FtG classes that does not collapse. While the notion of property preserving encryption and its security are defined in the abstract setting of a general k-ary property, the separation results are conditioned on the assumed existence of a PPEnc for a concrete binary property based on quadratic residuosity, called P_{qr}. Finally, the paper proposes a scheme for achieving orthogonality, which is claimed to be LoR secure in the generic bilinear group model.

Property preserving encryption has a direct connection with predicate private encryption [32]. In such a scheme, given a token one can check whether a ciphertext satisfies a certain predicate or not. A PPTag scheme may be easily constructed from a predicate private encryption scheme by concatenating ciphertext and token for a given message. If one starts from a full secure predicate-private scheme, one obtains an LoR secure PPTag scheme [29, 1]. In [29], the authors also claim that property preserving encryption is a generalization of order preserving encryption of Boldyreva et al. [8, 10, 9].

Our Motivation. Property preserving symmetric encryption is an interesting new concept, with a potential practical application for outsourcing computation and it is related to several other primitives like order preserving encryption and predicate encryption. Hence it is imperative that this notion be critically evaluated from the definitional perspective. Because of the separation, designers working on the problem of constructing property preserving encryption for various concrete properties may tend to disregard the FtG notion and only aim

at the strongest LoR notion, which is likely to take considerably more resources, see, for example, [1]. Thus it is natural to ask whether the separation indicates any real gap between the two notions and generalizes to any concrete property of interest or is it an artifact related to the peculiarities of the property considered in [29]. The importance of cryptanalyzing the proposed provably secure construction requires no further emphasis.

Our Contributions. In Sect. 3, we revisit the separation results of [29]. As no concrete construction of FtG-secure scheme for P_{qr} was suggested to validate the separation results, we first attempt to build such a scheme. The first observation is that the quadratic residuosity property used in the separation results of [29], can be generalized to a property preserving test of equality. Hence we focus on equality property and show that one-time pad is sufficient to achieve FtG security for equality preserving encryption of one-bit messages. Furthermore, the two notions of FtG and LoR security in fact collapse in such a *deterministic* setting. This result is further generalized for equality testing of n-bit messages where we show a pseudo-random permutation is sufficient to achieve the strongest LoR security. Thus, on one hand we can easily generalize the separation results of [29] for the equality property, on the other we show that in concrete terms the two notions of FtG and LoR effectively collapse for this property. This points to the inherent ambiguity with respect to the actual implication of the separation results for concrete properties of interest. Thus contextualized, we note that the question of whether the separation results of [29] actually indicate any real world difference between the two notions of FtG and LoR security for property preserving encryption still remains open.

In Sect. 4, we look at the relation of FtG and LoR in the context of orthogonality property. We show that given an FtG secure orthogonality preserving encryption of vectors of length $2n$, there exists LoR secure orthogonality preserving encryption of vectors of length n. This result gives further credence to our already established evidence that FtG is indeed a meaningful notion of security for property preserving encryption. We also show that in the property preserving scenario orthogonality implies equality.

In Sect. 5, we cryptanalyze the scheme for testing orthogonality from [29]. We show that the PPEnc scheme given in [29, Sect. 5] is not even weakest selective find-then-guess secure, which falsifies the claim [29, Theorem 5.1] that it is LoR secure. Going beyond indistinguishability, we show that if an adversary is allowed just one query and then given a ciphertext for some unknown message vector $x = (x_1, \ldots, x_n)$, s/he can extract significant non-trivial information about x including whether x is orthogonal to any message of adversary's choice. Thus the attack defeats the very purpose of having property preserving encryption in the symmetric key setting and may be of independent interest in understanding the security of cryptographic schemes in the composite order pairing setting.

We draw our conclusion in Sect. 6. Some of the detailed proofs are provided in Appendix A.

2 Definitions

We recall the basic definition of property preserving encryption and notions of its security from [29]. The paper claims that the idea makes most sense in the symmetric key setting – in the public key setting an adversary can gain non-trivial information about a target ciphertext by encrypting messages of her own choice and then testing for the property on the target message.

As in [29], we too model any k-ary property on $\mathcal{M}$ as a Boolean function on $\mathcal{M}^k$. One of the main properties considered is orthogonality, which depends on computing inner products in finite dimensional vector spaces over finite fields. Let $v = (v_1, \ldots, v_n)$ and $w = (w_1, \ldots, w_n)$ be vectors over a finite field $\mathbb{F}_q$. The inner product between them is defined as $v \cdot w = v_1 w_1 + \ldots + v_n w_n \pmod{q}$. These vectors are *orthogonal* if $v \cdot w = 0$.

Definition 1. *A property preserving encryption scheme (*PPEnc*) for the k-ary property P is a collection of four probabilistic polynomial time (PPT) algorithms, which are defined as follows:*

1. $\mathsf{Setup}(1^\lambda)$*: This takes as input the security parameter and outputs the message space ($\mathcal{M}$), public parameters (PP) and the secret key (SK).*
2. $\mathsf{Encrypt}(PP, SK, m)$*: This algorithm outputs the ciphertext CT corresponding to the message m, using the secret key SK and public parameter PP.*
3. $\mathsf{Decrypt}(PP, SK, CT)$*: This algorithm outputs the plaintext message m.*
4. $\mathsf{Test}(CT_1, \ldots, CT_k, PP)$*: This is a public algorithm that takes as inputs ciphertexts $CT_1, \ldots, CT_k$ corresponding to messages $m_1, \ldots, m_k$, respectively and outputs a bit.*

These set of four algorithms must satisfy the standard correctness requirement. In addition, if the Test *algorithm outputs $b \in \{0, 1\}$ then, except with negligible probability, one has $P(m_1, \ldots, m_k) = b$.*

A related notion of PPTag scheme was also defined. Informally, such a scheme does not have any decrypt module.

Definition 2. *A property preserving tag scheme (*PPTag*) for the k-ary property P is a collection of three probabilistic polynomial time (PPT) algorithms, which are defined as follows:*

1. $\mathsf{Setup}(1^\lambda)$*: This takes as input the security parameter and outputs the message space ($\mathcal{M}$), public parameters (PP) and the secret key (SK).*
2. $\mathsf{Encrypt}(PP, SK, m)$*: This algorithm outputs the ciphertext CT corresponding to the message m, using the secret key SK and public parameter PP.*
3. $\mathsf{Test}(CT_1, \ldots, CT_k, PP)$*: This is a public algorithm that takes as inputs ciphertexts $CT_1, \ldots, CT_k$ corresponding to messages $m_1, \ldots, m_k$, respectively and outputs a bit.*

This set of algorithms must satisfy the standard correctness requirement. If the Test *algorithm outputs $b \in \{0, 1\}$ then, except with negligible probability, one has $P(m_1, \ldots, m_k) = b$.*

Remark 1. In [29], the authors suggest the following strategy while designing a secure property preserving encryption scheme. The actual "payload" message is encrypted using an IND-CPA secure symmetric encryption scheme. For testing the property, a tag is constructed for each message using a PPTag scheme.

2.1 Security Notions

Inspired by the study of security notions of symmetric key encryption by Bellare et al. [4], Pandey and Rouselakis [29] propose several notions of security for property preserving symmetric encryption. These notions are defined by taking into account the specific nature of PPEnc. Here we informally describe the two notions of security for such schemes which are most relevant to our work. For more details refer to [29].

Definition 3. *For a k-ary property P, any two sequences $X = (x_1, \ldots, x_n)$ and $Y = (y_1, \ldots, y_n)$ of inputs are said to have the same equality pattern if*

$$P(x_{i_1}, \ldots, x_{i_k}) = P(y_{i_1}, \ldots, y_{i_k}),\ \forall\ (i_1, \ldots, i_k) \in [n]^k.$$

Find-then-Guess Security (FtG). Challenger and adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ plays the following game $\mathsf{Game}^{\mathsf{FtG}}_{\Pi,\mathcal{A},\lambda}(b)$ which is formally defined in [29, Sect. 3]. After the Setup phase, in $\mathcal{A}_1$, the adversary first adaptively queries the encryption oracle for messages $(m_1, \ldots, m_t)$. Then the adversary outputs the challenge messages (m_0^*, m_1^*). In $\mathcal{A}_2$, after the challenger returns the ciphertext of m_b^* for a random $b \in \{0, 1\}$, the adversary again adaptively queries $(m_{t+1}, \ldots, m_q)$. The adversary wins the game if s/he can correctly predict the bit b. Adversarial queries must satisfy the *extra* condition that the equality patterns of $(m_1, \ldots, m_t, m_0^*, m_{t+1}, \ldots, m_q)$ and $(m_1, \ldots, m_t, m_1^*, m_{t+1}, \ldots, m_q)$ are the same. Otherwise $\mathcal{A}$ can trivially win the game.

Definition 4. *Let $\Pi = \mathsf{Setup}, \mathsf{Encrypt}, \mathsf{Decrypt}, \mathsf{Test}$ be a symmetric key property preserving encryption scheme. Then Π is said to be FtG secure if there exists a negligible function $n(\cdot)$ such that for all PPT FtG adversaries $\mathcal{A}$ as above and for all $\lambda \in \mathbb{N}$ sufficiently large, the advantage of $\mathcal{A}$ in the FtG game is negligible:*

$$\mathsf{Adv}^{\mathsf{FtG}}_{\Pi,\mathcal{A},\lambda} = \left| \Pr\left[\mathsf{Game}^{\mathsf{FtG}}_{\Pi,\mathcal{A},\lambda}(1) = 1\right] - \Pr\left[\mathsf{Game}^{\mathsf{FtG}}_{\Pi,\mathcal{A},\lambda}(0) = 1\right] \right| \leq n(\lambda).$$

They [29] further introduce a hierarchy in the FtG notion depending on the number of challenge queries. In particular, any adversary playing the FtG^{η} game, for $\eta \in \mathbb{N}$, is allowed to make η many challenge queries interleaved between encryption oracle queries. A *selective* FtG notion may be defined in the usual way, following [11], where the adversary outputs the challenge messages even before receiving the public parameters.

Left-or-Right Security (LoR). Challenger and adversary $\mathcal{A}$ plays the following game $\mathsf{Game}^{\mathsf{LoR}}_{\Pi,\mathcal{A},\lambda}(b)$. After setup, $\mathcal{A}$ makes q encryption queries, where each query is of the form $(m_0^{(i)}, m_1^{(i)})$. The queries are such that the tuples $(m_0^{(1)}, \ldots, m_0^{(q)})$ and $(m_1^{(1)}, \ldots, m_1^{(q)})$ have the same equality pattern. The challenger returns the

encryption of $m_b^{(i)}$ for each i where the random bit b is chosen at the beginning of game. At the end, the adversary has to output a guess b' of b and wins if $b' = b$. The game is formally defined in [29, Sect. 3]. The definition of adversarial advantage is as follows.

Definition 5. *Let $\Pi = \mathsf{Setup}, \mathsf{Encrypt}, \mathsf{Decrypt}, \mathsf{Test}$ be a symmetric key property preserving encryption scheme. Then Π is said to be* LoR *secure if there exists a negligible function $n(\cdot)$ such that for all PPT* LoR *adversaries $\mathcal{A}$ as above and for all $\lambda \in \mathbb{N}$ sufficiently large, the advantage of $\mathcal{A}$ in the* LoR *game is negligible:*

$$\mathsf{Adv}^{\mathsf{LoR}}_{\Pi,\mathcal{A},\lambda} = \left|\Pr\left[\mathsf{Game}^{\mathsf{LoR}}_{\Pi,\mathcal{A},\lambda}(1) = 1\right] - \Pr\left[\mathsf{Game}^{\mathsf{LoR}}_{\Pi,\mathcal{A},\lambda}(0) = 1\right]\right| \leq n(\lambda).$$

3 Separation Results: A Closer Look

Let $\mathcal{QR}_p$ (resp. $\mathcal{QNR}_p$) be the set of quadratic residues (resp. quadratic non-residues) in $\mathbb{Z}_p^*$ for some prime p. Consider the quadratic residuosity property P_{qr} defined as follows:

$$P_{qr}(m_1, m_2) = \begin{cases} 1 \text{ if } m_1 \cdot m_2 \in \mathcal{QR}_p \\ 0 \text{ if } m_1 \cdot m_2 \in \mathcal{QNR}_p \end{cases} \tag{1}$$

Assuming there *exists* an FtG secure property preserving encryption scheme Π for P_{qr}; Pandey and Rouselakis construct an artificial scheme Π' which is FtG but not LoR secure [29, Theorem 4.1]. In a similar fashion they establish that $\mathsf{FtG}^{\eta} \nrightarrow \mathsf{FtG}^{\eta+1}$ [29, Theorem 4.4]. Note that (i) the separation results are *specific* to property P_{qr} and (ii) *conditioned* on the existence of FtG secure scheme for P_{qr} and no such construction was known or suggested in [29].

Property preserving encryption is a rather broad category and a separation based on the specificity of a particular property does not necessarily provide enough insight about the relationship between different security notions for another concrete property or how two notions are related in general. For example, the separation result for P_{qr} in [29] does not give any clue whether the same will hold for another property, say orthogonality. Another crucial question is whether the separation is real or merely an artifact – is there any 'natural' construction for a 'natural' property that is FtG but not LoR secure.

Clearly, a thorough investigation of these questions requires identifying natural properties that encompass other properties and then analysing the real difference between security notions of property preserving encryption in the context of these natural properties. For example, consider the set of all unary properties. It is suggested [29] that for any unary property P, a PPTag scheme can be trivially obtained by providing $P(m)$ in the clear as part of the ciphertext. We note that in such a scenario, the two notions FtG and LoR actually collapse. The case for binary properties, however, is more subtle as we see next.

3.1 Equivalence Testing via Equality

We demonstrate that certain equivalence relations can be tested via equality property – P_{qr} property used in [29] is one such relation.

Claim 1. *To construct a* PPTag *scheme for P_{qr}; it suffices to construct a* PPTag *scheme for equality where the message space is $\mathcal{M} = \{0, 1\}$.*[1]

Proof. The argument is quite straightforward. A "sign" function S was used by [29] to define P_{qr} where $S(m) = 0$ if $m \in \mathcal{QR}_p$; else $S(m) = 1$. In other words, P_{qr} divides the message space $\mathcal{M} = \mathbb{Z}_p^*$ into 2 equivalence classes. Given any message in $\mathbb{Z}_p^*$ one can efficiently determine $S(m)$ and then use the PPTag scheme for equality over the message space $\{0, 1\}$ to encrypt $S(m)$. Product of two messages x and y belongs to $\mathcal{QR}_p$ if and only if both x and y belong to same equivalence class. Thus testing whether the product of x and y is a quadratic residue or not is now reduced to the task of testing whether $S(x)$ and $S(y)$ are equal or not. □

The property P_{qr} used in [29] is a particular instance of a larger class of property $\mathcal{P}$. In particular, the property $\mathcal{P}$ induces an equivalence relation on a set $\mathcal{M}$ such that there exists an efficient algorithm to determine the class in which a given element lies. Another example of such property is to test, given two integers m and n, whether their difference is divisible by a fixed prime p. It is easy to see that a PPTag scheme for such a property $\mathcal{P}$ can be realized by any PPTag scheme for equality. Note, however, that there do exist equivalence relations for which the question of membership testing is not known to be easy.

3.2 Natural LoR Secure Equality Testing

We describe a property preserving encryption scheme for testing equality over message space $\{0, 1\}$.

1. $\mathsf{Setup}(1^\lambda)$: Set $SK = t$, where $t \in_R \{0, 1\}$.
2. $\mathsf{Encrypt}(SK, m)$: $CT = t \oplus m$.
3. $\mathsf{Decrypt}(SK, CT)$: $m' = CT \oplus t$.
4. $\mathsf{Test}(CT_1, CT_2)$: Return 1 if and only if $CT_1 = CT_2$.

It is well-known that as a symmetric key encryption scheme the above construction (or any deterministic encryption scheme) is not FtG secure in the sense of [4] but it is as a PPEnc as the following claim shows.

Claim 2. *The above construction is an* FtG *secure* PPEnc *for one-bit messages.*

[1] Here and afterwards we often focus on PPTag schemes as the problem of constructing a PPEnc is essentially reduced to the problem of constructing a PPTag scheme (see Remark 1.

Proof. The key idea is that an FtG adversary $\mathcal{A}$ is restricted by the equality pattern. If $\mathcal{A}$ makes the challenge query as $(0,1)$ or $(1,0)$ then s/he cannot make any encryption oracle query. Hence, the one-time pad ensures the challenge bit is information theoretically hidden from $\mathcal{A}$. On the other hand, if the challenge query is of the form $(0,0)$ or $(1,1)$ then there is no non-trivial information for $\mathcal{A}$ to learn either from the encryption queries or from the challenge. □

The above result further leads us to the following interesting consequence. Let $E : \mathcal{K} \times \{0,1\} \longrightarrow \{C_0, C_1\}$ be a deterministic encryption scheme.

Claim 3. *If E is* FtG *secure* PPEnc *scheme for equality then it is* LoR *secure.*

Proof. Let $\mathcal{A}$ be a valid LoR adversary for E. We will construct a valid FtG adversary $\mathcal{B}$ for E, which is playing the FtG game with its own challenger $\mathcal{C}$ by internally running $\mathcal{A}$.

Observe that $\mathcal{A}$ has to respect the equality pattern and hence can only make queries from the following disjoint sets: $S_1 = \{(0,0),(1,1)\}$ and $S_2 = \{(0,1),(1,0)\}$. If $\mathcal{A}$ makes queries from the set S_1, then FtG $\longrightarrow$ LoR holds trivially.

Now let us analyze the case when $\mathcal{A}$ makes queries from $S_2 = \{(0,1),(1,0)\}$. Let us, without loss of generality, assume that $\mathcal{A}$'s first query is $(0,1)$. $\mathcal{B}$ sets the same message $(0,1)$ as its own FtG challenge query, forwards it to $\mathcal{C}$. In response $\mathcal{C}$ provides a challenge ciphertext C_b to $\mathcal{B}$, $b \in \{0,1\}$ by encrypting $\beta \in_R \{0,1\}$ using the encryption function E as per the rule of the FtG game. $\mathcal{B}$ forwards the same C_b to $\mathcal{A}$. Note that by the definition of FtG security, $\mathcal{B}$ cannot make any other query to $\mathcal{C}$. However, if $\mathcal{A}$ repeats the same query $(0,1)$, then $\mathcal{B}$ simply forwards the same ciphertext C_b. If $\mathcal{A}$ queries the other valid message pair $(1,0)$, then $\mathcal{B}$ returns ciphertext C_{1-b}. When $\mathcal{A}$ outputs a bit as its guess and halts, then $\mathcal{B}$ outputs the same bit to $\mathcal{C}$ and halts.

The simulation of $\mathcal{A}$'s environment by $\mathcal{B}$ is perfect. In fact, after the first query, $\mathcal{A}$ can on its own generate the response for all other queries it is going to make. Now the FtG security of E ensures that the encryption of 1 is indistinguishable from the encryption of 0. Hence, the advantage of $\mathcal{B}$ is same as that of $\mathcal{A}$ and the two notions actually collapse. □

As a consequence we note that the one-time pad construction of PPEnc achieves LoR security. However, it is well-known that the same is not even FtG secure as standard symmetric key encryption scheme. Thus there exists binary property preserving encryption scheme secure in the strong LoR sense of property preserving encryption but does not even achieve FtG security as a standard symmetric key encryption scheme.

Based on our previous observations we suggest the following direct construction of LoR secure PPEnc for equality testing on $\mathcal{M} = \{0,1\}^n$. A PPTag can be obtained by dropping the Decrypt algorithm from the description.[2]

[2] Similar construction for testing equality in the context of authenticated encryption and searchable encryption schemes was suggested earlier by Rogaway-Shrimpton [31] and Amanatidis et al. [2]. Their constructions used deterministic MAC which is modeled as a PRF.

Property Preserving Encryption for Equality. We describe a scheme Π to test for equality of strings of length n.[3] Let $\{\mathcal{F}\}_n$ be a pseudo-random permutation (PRP) family and an element $F \in \{\mathcal{F}\}_n$ is defined as $F : \{0,1\}^n \times \{0,1\}^n \longrightarrow \{0,1\}^n$.

1. $\mathsf{Setup}(1^\lambda)$: Set a random n-bit binary string K as the secret key SK.
2. $\mathsf{Encrypt}(SK, m)$: $CT = F_K(m)$.
3. $\mathsf{Decrypt}(SK, CT)$: Return $F_K^{-1}(CT)$.
4. $\mathsf{Test}(CT_1, CT_2)$: Return 1 if and only if $CT_1 = CT_2$.

Claim 4. *If the underlying PRP family is secure, then Π is* LoR *secure.*

Proof. (Sketch) The claim is established through a simple hybrid argument. Let the adversary $\mathcal{A}$ for the LoR game set $(m^*_{0,1}, m^*_{1,1}), \ldots, (m^*_{0,t}, m^*_{1,t})$ as challenges. We claim that the games $\mathsf{Game}_0 : m^*_{0,1}, \ldots, m^*_{0,t}$ and $\mathsf{Game}_1 : m^*_{1,1}, \ldots, m^*_{1,t}$ are indistinguishable. We note that, by the security of the PRP, the Game_0 is indistinguishable from a game where the challenger computes the response from a random permutation. Similarly, challenges output in Game_1 will be indistinguishable from the output of a random permutation. □

3.3 Separation Between FtG and LoR Notions for Equality

After establishing the existence of natural PPEnc/PPTag scheme for equality testing satisfying LoR security (and, hence, FtG security), we now generalize the result of [29, Theorem 4.1] to show that the separation holds for the equality property and need not necessarily be restricted to small number of equivalence classes. Let $\mathcal{M}$ be the message space. Suppose $z = \lceil \log_2 |\mathcal{M}| \rceil$ so that every element $m \in \mathcal{M}$ can be represented by a bit string of length z. Note that z (and not $|\mathcal{M}|$) is a polynomial in the security parameter. Let $\Pi = (\mathsf{Setup}, \mathsf{Encrypt}, \mathsf{Test})$ be any FtG secure PPTag scheme for equality. From this scheme we construct another scheme $\Pi' = (\mathsf{Setup}', \mathsf{Encrypt}', \mathsf{Test}')$ for realizing the same property. The construction uses a PRF family $\mathcal{F} : \{0,1\}^\kappa \times \{0,1\}^z \longrightarrow \{0,1\}^z$.[4]

1. $\mathsf{Setup}'(1^\lambda)$: Calls Setup of Π to obtain (PP, SK) and chooses $k \in_R \{0,1\}^\kappa$ (as the key for the PRF). The algorithm outputs PP as the public parameters for Π' and sets the secret key as $SK' = (SK, k)$.
2. $\mathsf{Encrypt}'(PP, SK', m)$: While encrypting $m \in \mathcal{M}$, the encryption algorithm of Π is used to obtain $ct = \mathsf{Encrypt}(PP, SK, m)$. Then choose a bit $b \in_R \{0,1\}$. The ciphertext of Π' is computed as

$$CT = \begin{cases} (ct, b, F_k(m)), \text{ if } b = 0, \\ (ct, b, F_k(m) \oplus m), \text{ otherwise.} \end{cases}$$

[3] For the case of PPTag there is no need to decrypt and hence the construction can be extended to arbitrary length messages by the use of a CRHF H with n-bit digests.

[4] The PRF can be replaced by a set of $|\mathcal{M}|$ random bit strings when $|\mathcal{M}|$ is *small* (i.e., polynomial in the security parameter). On the other hand, for arbitrary length messages one can use a collision resistant hash function (CRHF) H to first map the message to a digest of z-bit and then apply the PRF on the digest.

3. $\mathsf{Test}'(CT_1, CT_2, PP)$: Given $CT_1 = (ct_1, b_1, t_1)$ and $CT_2 = (ct_2, b_2, t_2)$, the algorithm outputs $\mathsf{Test}(ct_1, ct_2, PP)$.

The following two lemma generalize the result of [29] and together establish that the separation result for FtG and LoR holds for equality property. We provide the proofs in Appendix A.

Lemma 1. *If the scheme Π is* FtG *secure and $\mathcal{F}$ is a secure PRF then Π' constructed as above is also* FtG *secure. In particular, $\epsilon_{\Pi'} \leq \epsilon_{\Pi} + 2\epsilon_{\mathcal{F}}$ where ϵ_X denotes the advantage in the corresponding security game for the primitive $X \in \{\Pi, \mathcal{F}, \Pi'\}$.*

Lemma 2. *There is an* LoR *adversary for the scheme Π' with non-negligible advantage.*

Remark 2. We point out an interesting consequence of the above separation result. Shen-Shi-Waters [32] proposed two security notions, the single challenge and full challenge security for predicate private symmetric encryption (see [32] for the definitions of security). The strategy outlined in Lemma 1 and Lemma 2 in the context of PPTag can be adapted to establish a similar separation between single challenge and full challenge security of predicate private encryption. Suppose we are given a single challenge secure predicate private scheme for equality, called Ψ. From that we construct another scheme Ψ' where the only changes are in the Setup and Encrypt as described in the context of Π' above. In particular, the encryption algorithm of Ψ' outputs a ciphertext of Ψ together with either $(b, F_k(m))$ or $(b, F_k(m) \oplus m)$ depending upon whether $b = 0$ or $b = 1$. A similar argument as in the case of PPTag above shows that Ψ' is single challenge secure but not full secure.

Hierarchy Among FtG Classes. We briefly comment on the separation result for the hierarchy among FtG classes given in [29]. The reader may refer to the full version [20] for further details. The equality property over small message space is used to establish the result. We start with a scheme Π which is FtG^{η} secure and derive a scheme Π' which is not $\mathsf{FtG}^{\eta+1}$ secure. For each message m the Setup algorithm of Π' stores a set of random bit strings $\{t_{m,1}, \ldots, t_{m,\eta}\}$ as part of secret key. Encryption algorithm of Π' chooses $b \in_R \{1, \ldots, \eta + 1\}$ and returns

$$\Pi'.\mathsf{Encrypt}(PP, SK, m) = (\Pi.\mathsf{Encrypt}(PP, SK, m), b, val),$$

where

$$val = \begin{cases} t_{m,b}, \text{ if } 1 \leq b \leq \eta \\ t_{m,1} \oplus \ldots \oplus t_{m,\eta)} \oplus m, \text{ if } b = \eta + 1. \end{cases}$$

The derived scheme Π' is not $\mathsf{FtG}^{\eta+1}$ secure, but FtG^{η} secure.

3.4 The Bottom Line

At this point a reader may wonder what could be a plausible conclusion of our analysis. On one hand, a PRP is sufficient to construct LoR secure PPEnc for equality and the two notions of FtG and LoR collapse in such a setting. On the other, for the same property there is a *theoretical* gap between FtG and LoR notions of security which may or may not be the case for other properties of interest. In fact, in the next section we show that for orthogonality any FtG secure PPEnc for vectors of length $2n$ gives an LoR secure PPEnc for vectors of length n.

It seems the only reasonable conclusion is that no conclusive evidence exists indicating any real world difference between the two notions of security for PPEnc in general. This leads us to the following open question: is there a 'natural' construction of a scheme for testing equality or, for that matter, any other 'natural' property, which is FtG secure but not LoR secure. Resolving this question will shed further light into the usefulness of the hierarchy of security notions introduced in [29].

4 Orthogonality: Relation Between FtG and LoR and with Equality

We show that it is possible to construct an LoR secure scheme from FtG secure scheme for orthogonality which provides evidence that FtG is a meaningful notion for property preserving encryption. Next, we show that orthogonality implies equality in the property preserving scenario.

4.1 FtG_{2n} implies LoR_n

Shen, Shi and Waters showed [32, Theorem 2.8] that a single challenge secure symmetric key predicate-only encryption scheme for testing orthogonality of vectors of length $2n$ may be used to construct one achieving full security for n length vectors. Inspired by their technique we derive a similar result for property preserving orthogonality testing.

Let Θ_{2n} be an FtG secure PPTag encryption scheme for testing orthogonality of vectors of length $2n$. We construct a PPTag scheme Θ_n for testing orthogonality of vectors of length n as follows. In the following we assume that the underlying field on which the vectors are defined does not have characteristic 2 (this is a technical requirement in the security argument). For $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$, as usual $x||y := (x_1, \ldots, x_n, y_1, \ldots, y_n)$.

1. $\Theta_n \cdot \mathsf{Setup}(1^\lambda)$: The public parameters and the secret key are the same as the corresponding ones of Θ_{2n}.
2. $\Theta_n \cdot \mathsf{Encrypt}(PP, SK, x)$: The ciphertext is $\Theta_{2n} \cdot \mathsf{Encrypt}(PP, SK, x||x)$.
3. $\Theta_n \cdot \mathsf{Test}(CT_1, CT_2, PP)$: The test is carried out using that of the Θ_{2n} scheme as $\Theta_n \cdot \mathsf{Test}(CT_1, CT_2, PP) = 1$ if and only if $\Theta_{2n} \cdot \mathsf{Test}(CT_1, CT_2, PP) = 1$.

Next, we show that Θ_n is LoR secure. The proof proceeds via a sequence of hybrids. Any adversary who can distinguish two adjacent games can break the FtG security of Θ_{2n}.

Theorem 5. *The scheme Θ_{2n} is* FtG *secure implies the derived scheme Θ_n is* LoR *secure.*

Proof. (Sketch) Recall that we have assumed that the underlying field on which the vectors are defined does not have characteristic 2. We observe that $x \cdot y = 0$ if and only if $(x||x) \cdot (y||y) = 0$. The encoding which maps x to $x||x$ is used for proving LoR security via a hybrid argument.

Let $\mathcal{A}$ be a valid LoR adversary for Θ_n. The adversary $\mathcal{A}$ sets as challenges the pairs $(x_0^{(1)}, x_1^{(1)}), \ldots, (x_0^{(q)}, x_1^{(q)})$ to the challenger $\mathcal{C}$. The challenger fixes a random bit b and returns encryption of $x_b^{(i)},\ 1 \leq i \leq q$. The adversary outputs a bit b' at the end of the game and wins if $b = b'$.

We prove that the distributions of the ciphertexts of the sequence of messages $(x_0^{(1)}, x_0^{(2)}, \ldots, x_0^{(q)})$ and $(x_1^{(1)}, x_1^{(2)}, \ldots, x_1^{(q)})$ are indistinguishable. That is, the adversary $\mathcal{A}$ cannot distinguish the games $\mathcal{G}_0$ and $\mathcal{G}_1$ of Table 1. The proof proceeds via a sequence of hybrid games. We tabulate the sequence of hybrids in Table 1. In $\mathcal{G}_B$, the value α is chosen at random from the underlying field. We mention that a sequence of intermediate games is defined between two consecutive games for proving indistinguishability, where only one ciphertext is changed. One such sequence between $\mathcal{G}_A$ and $\mathcal{G}_B$ is given in Table 1.

Table 1. Left: Sequence of hybrids $\mathcal{G}_0$ through $\mathcal{G}_1$; Right: Intermediate games between $\mathcal{G}_A$ and $\mathcal{G}_B$

$\mathcal{G}_0 : x_0^{(1)}\|\|x_0^{(1)}, \ldots, x_0^{(q)}\|\|x_0^{(q)}$
$\mathcal{G}_A : x_0^{(1)}\|\|0, \ldots, x_0^{(q)}\|\|0$
$\mathcal{G}_B : x_0^{(1)}\|\|\alpha x_1^{(1)}, \ldots, x_0^{(q)}\|\|\alpha x_1^{(q)}$
$\mathcal{G}_C : 0\|\|\alpha x_1^{(1)}, \ldots, 0\|\|\alpha x_1^{(q)}$
$\mathcal{G}_D : x_1^{(1)}\|\|\alpha x_1^{(1)}, \ldots, x_1^{(q)}\|\|\alpha x_1^{(q)}$
$\mathcal{G}_1 : x_1^{(1)}\|\|x_1^{(1)}, \ldots, x_1^{(q)}\|\|x_1^{(q)}$

$\mathcal{G}_A : x_0^{(1)}\|\|0, x_0^{(2)}\|\|0, \ldots, x_0^{(q)}\|\|0$
$\mathcal{G}_{A,1} : x_0^{(1)}\|\|\alpha x_1^{(1)}, x_0^{(2)}\|\|0, \ldots, x_0^{(q)}\|\|0$
$\mathcal{G}_{A,2} : x_0^{(1)}\|\|\alpha x_1^{(1)}, x_0^{(2)}\|\|\alpha x_1^{(2)}, x_0^{(3)}\|\|0, \ldots, x_0^{(q)}\|\|0$
$\vdots \qquad \vdots$
$\mathcal{G}_B : x_0^{(1)}\|\|\alpha x_1^{(1)}, \ldots, x_0^{(q)}\|\|\alpha x_1^{(q)}$

We first argue that $\mathcal{G}_0$ and $\mathcal{G}_A$ are indistinguishable. Consider an intermediate game, called $\mathcal{G}_{0,1}$, defined as $x_0^{(1)}||0, x_0^{(2)}||x_0^{(2)}, \ldots, x_0^{(q)}||x_0^{(q)}$.

Notice that this game differs from $\mathcal{G}_0$ only in the first component. We claim that $\mathcal{G}_0$ and $\mathcal{G}_{0,1}$ are indistinguishable. For, suppose $\mathcal{A}$ can distinguish them. Setting $(x_0^{(1)}||x_0^{(1)}, x_0^{(1)}||0)$ as challenge messages and querying the rest of the elements, $\mathcal{A}$ can be used to construct a valid FtG adversary for Θ_{2n}. We proceed by defining a sequence of games where any two consecutive games vary exactly at one component. Similar argument would show that $\mathcal{G}_B$ and $\mathcal{G}_C$ are indistinguishable. The games $\mathcal{G}_C$ and $\mathcal{G}_D$ too may similarly be shown to be indistinguishable.

Recall that $\mathcal{G}_B$ was defined using a random parameter α. Even though, say for example $(x_0^{(1)}||0) \cdot (x_0^{(2)}||0) \neq 0$ holds, it may so happen that $(x_0^{(1)}||x_1^{(1)}) \cdot (x_0^{(2)}||x_1^{(2)}) = 0$. Thus, a random choice of α ensures that setting as the challenge $(x_0^{(1)}||0, x_0^{(1)}||\alpha x_1^{(1)})$ and the rest of the elements as queries one gets a valid FtG adversary for Θ_{2n}. This argument shows that $\mathcal{G}_A$ and $\mathcal{G}_B$ are indistinguishable. Similar argument shows that $\mathcal{G}_D$ and $\mathcal{G}_1$ are indistinguishable. □

4.2 A Direct Test for Equality from Orthogonality

Katz et al. [28] suggested a simple encoding to test for equality using inner product: create a ciphertext for $\mathcal{I} = (1, I)$ and a token for $\mathcal{J} = (-J, 1)$. Now the inner product of $\mathcal{I}$ and $\mathcal{J}$ is 0 if and only if $I = J$. This encoding does not directly work for property preserving encryption as there is no separate token and the Test is performed only on the ciphertexts. Nevertheless, we show that one can construct a scheme for testing equality property, given a scheme for testing orthogonality of vectors. The new scheme inherits the same security as that of the underlying orthogonality testing scheme. Note that this result is of theoretical interest, but of little practical value as we already have much more efficient scheme for testing equality.

The setting is as follows. Let the message space be $\mathbb{F}_q$, where the finite field is assumed to contain $i = \sqrt{-1}$. Examples of fields which contain i are $\mathbb{F}_{2^n}$; $\mathbb{F}_p$, where $p \equiv 1 \pmod 4$; or extensions of the form $\mathbb{F}_q$ which contain i. The square root of -1 may be given explicitly or may be computed using Tonelli-Shanks algorithm [3, Chapter 7].

We encode any $x \in \mathbb{F}_q$ as a vector in $\mathbb{F}_q^5$, where the encoding is given by $x \mapsto v_x = (x^2+1, ix^2, ix, ix, i)$ (in characteristic 2 fields $m \mapsto v_m = (m+1, m, 1)$). The mapping $m \mapsto v_m$ is one-to-one. Observe that, elements x and y are equal if and only if $v_x \cdot v_y = 0$. We now describe a scheme Π' for testing equality, given a scheme Π for testing orthogonality of vectors of length 5 over $\mathbb{F}_q$.

1. $\mathsf{Setup}(1^\lambda)$: The public parameters and secret key for Π' are those of Π.
2. $\mathsf{Encrypt}(PP, SK, m)$: While encrypting $m \in \mathbb{F}_q$, the encryption algorithm first computes the encoding v_m corresponding to m. Then the ciphertext corresponding to m is $CT = \Pi.\mathsf{Encrypt}(PP, SK, v_m)$.
3. $\mathsf{Test}(CT_1, CT_2, PP)$: Same as that of Π.

Lemma 3. *If Π is* FtG *(respectively* LoR*) secure then so is Π', correspondingly.*

Proof. We describe the FtG case as the LoR case may be similarly handled. Suppose Π' is not FtG secure, with $\mathcal{A}_{\Pi'}$ a valid adversary. We construct $\mathcal{A}_\Pi$, an FtG adversary for scheme Π, which internally runs $\mathcal{A}_{\Pi'}$. Whenever $\mathcal{A}_{\Pi'}$ makes an encryption query m, the adversary $\mathcal{A}_\Pi$ forwards v_m to the challenger $\mathcal{B}_\Pi$. On receiving the ciphertext, it forwards it to $\mathcal{A}_{\Pi'}$. When $\mathcal{A}_{\Pi'}$ sets (m_0^*, m_1^*) as challenge, the adversary $\mathcal{A}_\Pi$ forwards $(v_{m_0^*}, v_{m_1^*})$ to the challenger. On receiving the encryption of one of the two vectors $\mathcal{A}_\Pi$ forwards it to $\mathcal{A}_{\Pi'}$. The other queries made by $\mathcal{A}_{\Pi'}$ may be handled similarly. When $\mathcal{A}_{\Pi'}$ outputs a bit b' and halts,

so does $\mathcal{A}_{\Pi}$. This is a perfect simulation and $\mathcal{A}_{\Pi}$ wins with the same advantage as that of $\mathcal{A}_{\Pi'}$. □

5 Cryptanalysis of Pandey and Rouselakis Construction

The only construction proposed in [29] is a PPTag scheme for testing orthogonality of two vectors over a finite field. The proposed scheme works in the composite order bilinear pairing setting. It is claimed without proof in [29, Theorem 5.1] that the scheme achieves LoR security in the generic group model with a precise bound on the adversarial advantage.

We identify an inherent symmetry in the construction that is required for the public Test algorithm. The same symmetry allows the adversary to construct 'pseudo-ciphertext' for many messages from a valid ciphertext of a known message. Suitably manipulated pseudo-ciphertext can be exploited by the adversary to win the indistinguishability game with overwhelming probability. Thus the scheme is not even selective FtG secure. However, the properties of pseudo-ciphertexts allow an adversary to go even further. We show that, after making a single query, an adversary can gain non-trivial information about the underlying message vector given any valid ciphertext. In particular, the adversary can choose any vector and then check whether the unknown message is orthogonal to it or not. This effectively negates the main motivation of using the symmetric key setting for property preserving encryption.

5.1 Pandey and Rouselakis Construction

We recall the scheme of [29] for testing orthogonality of two vectors defined over a prime field $\mathbb{F}_p$, referred to as PR scheme hereafter.

1. $\mathsf{Setup}(1^\lambda, n)$: Pick two distinct primes p and q uniformly at random in the range $(2^{\lambda-1}, 2^\lambda)$ where λ is the security parameter. Let $\mathbb{G}$ and $\mathbb{G}_T$ be two groups of order $N = pq$ such that there is an efficiently computable bilinear map $e : \mathbb{G} \times \mathbb{G} \longrightarrow \mathbb{G}_T$. Select a vector $(\gamma_1, \ldots, \gamma_n) \in \mathbb{Z}_q$ such that $\sum_{i=1}^n \gamma_i^2 = \delta^2 \pmod q$. Let g_0 (resp. g_1) be a generator of a subgroup of order p (resp. q) of $\mathbb{G}$. Set the message space as $\mathcal{M} = (\mathbb{Z}_N^* \bigcup \{0\})^n$. Set
$$PP = \langle n, N, \mathbb{G}, \mathbb{G}_T, e \rangle, \quad SK = \langle g_0, g_1, \{\gamma_i\}_{i=1}^n, \delta \rangle.$$
2. $\mathsf{Encrypt}(PP, SK, M)$: On input a message $M = (m_1, \ldots, m_n)$, select two random elements ϕ and ψ from $\mathbb{Z}_N$. The ciphertext is computed as
$$CT = (ct_0, \{ct_i\}_{i=1}^n) = \left(g_1^{\psi\delta}, \{g_0^{\phi m_i} \cdot g_1^{\psi\gamma_i}\}_{i=1}^n \right).$$
3. $\mathsf{Test}(CT^{(1)}, CT^{(2)}, PP)$: When two ciphertexts $CT^{(1)} = (ct_0^{(1)}, \{ct_i^{(1)}\}_{i=1}^n)$ and $CT^{(2)} = (ct_0^{(2)}, \{ct_i^{(2)}\}_{i=1}^n)$ are input, the algorithm outputs 1 if and only if:
$$\prod_{i=1}^n e(ct_i^{(1)}, ct_i^{(2)}) = e(ct_0^{(1)}, ct_0^{(2)}).$$

Correctness ensures that Test outputs 1 only when the underlying messages are orthogonal, except with a negligible probability.

5.2 A Valid FtG Adversary

Notice that the construction ensures that the quadratic form relation $\gamma_1^2 + \gamma_2^2 + \ldots + \gamma_n^2 = \delta^2 \pmod q$ is formed in the exponent for one subgroup element of $\mathbb{G}_T$ while the inner product of the two message vectors is computed in the exponent of the other. However, the above equality implies that $\gamma_1(\gamma_1+\gamma_2)+\gamma_2(\gamma_2-\gamma_1)+\gamma_3^2+\ldots+\gamma_n^2 = \delta^2 \bmod q$ also holds.

Given a ciphertext for some message $x = (x_1, \ldots, x_n)$, say $(c_0, c_1, c_2, \ldots, c_n)$, the tuple $W = (c_0, c_1 \cdot c_2, c_2/c_1, c_3, \ldots, c_n)$ may be computed. We can hence easily see that the tuple W may be used in the Test algorithm in place of a valid ciphertext of $x' = (x_1 + x_2, x_2 - x_1, x_3, \ldots, x_n)$. The advantage is that, even though the adversary is forbidden to query x' in the security game, s/he may still obtain a ciphertext of x if it is a valid query, and then, compute and use W for testing for orthogonality to x'.

Many such relations among the secret key tuple $(\gamma_1, \ldots, \gamma_n)$ exist that are equal to δ^2. We give more such examples in Lemma 4. But, this observation motivates us to define the notion of *pseudo-ciphertext*.

Definition 6. *A pseudo-ciphertext for PR scheme, associated with a valid message z, is an element $W_z \in \mathbb{G}^{n+1}$ such that $\mathsf{Test}(CT_x, W_z, PP) = 1$ if and only if $\mathsf{Test}(CT_x, CT_z, PP) = 1$, except with negligible probability, where CT_x and CT_z are properly formed ciphertexts for x and z respectively.*

Next, we prove that [29] scheme is not FtG secure.

Proposition 1. *The* PPTag *scheme proposed in [29] for testing orthogonality is not even* ***selective FtG*** *secure.*

Proof. One can construct a valid selective FtG adversary for the $n = 2$ case as follows. The adversary sets $(0, 1)$ and $(1, 0)$ as challenges. Then s/he queries $(1, 1)$ and forms a pseudo-ciphertext for $(2, 0)$. Using that pseudo-ciphertext adversary can trivially win the indistinguishability game.

Now consider the case where $n \geq 3$. The claim is established in terms of the following attack game between the adversary ($\mathcal{A}$) and the challenger ($\mathcal{S}$).
(**i**) $\mathcal{A}$ outputs a pair of n-dimensional vectors (μ_0^*, μ_1^*) as the challenge messages where $n \ll N$. The challenges are of the form $\mu_0^* = (m_1, m_0, 1, \ldots, 1)$ and $\mu_1^* = (m_1, m_1, 1, \ldots, 1)$, where $m_1 \neq m_0$ are from $\mathbb{Z}_N^*$.
(**ii**) $\mathcal{A}$ receives the public parameter PP from challenger.
(**iii**) $\mathcal{A}$ queries $Q = ((m_1+m_0)/2, (m_0-m_1)/2, 1, \ldots, 1, -(n-3))$. Observe that Q is not orthogonal to either of the challenge messages μ_0^* and μ_1^* and hence, is a valid query. $\mathcal{S}$ responds with CT_Q, which is equal to

$$\left(g_1^{\psi\delta}, g_0^{\phi(m_1+m_0)/2} g_1^{\psi\gamma_1}, g_0^{\phi(m_0-m_1)/2} g_1^{\psi\gamma_2}, g_0^{\phi} g_1^{\psi\gamma_3}, \ldots, g_0^{\phi} g_1^{\psi\gamma_{n-1}}, g_0^{-(n-3)\phi} g_1^{\psi\gamma_n}\right)$$

for some $\psi, \phi \in_R \mathbb{Z}_N$. Given CT_Q, $\mathcal{A}$ takes the product and ratio of the third and second components of the ciphertext to obtain respectively $g_0^{m_0\phi} g_1^{\psi(\gamma_1+\gamma_2)}$ and $g_0^{-m_1\phi} g_1^{\psi(\gamma_2-\gamma_1)}$. $\mathcal{A}$ now computes the *pseudo-ciphertext* (Definition 6) $W_{Q'}$ for $Q' = (m_0, -m_1, 1, \ldots, 1, -(n-3))$ as

$$(g_1^{\psi\delta}, g_0^{m_0\phi} g_1^{\psi(\gamma_1+\gamma_2)}, g_0^{-m_1\phi} g_1^{\psi(\gamma_2-\gamma_1)}, g_0^{\phi} g_1^{\psi\gamma_3}, \ldots, g_0^{\phi} g_1^{\psi\gamma_{n-1}}, g_0^{-(n-3)\phi} g_1^{\psi\gamma_n}).$$

Note that the message vector Q' is orthogonal to μ_0^* but not to μ_1^*.
(**iv**) $\mathcal{A}$ now asks for the challenge ciphertext. Suppose that $\mathcal{S}$ responds with an encryption for μ_b^*

$$CT_b = \left(g_1^{\tilde{\psi}\delta}, g_0^{m_1\tilde{\phi}} g_1^{\gamma_1\tilde{\psi}}, g_0^{m_b\tilde{\phi}} g_1^{\gamma_2\tilde{\psi}}, g_0^{\tilde{\phi}} g_1^{\gamma_3\tilde{\psi}}, \cdots, g_0^{\tilde{\phi}} g_1^{\gamma_n\tilde{\psi}}\right),$$

where $b \in_R \{0,1\}$ and $\tilde{\phi}, \tilde{\psi} \in_R \mathbb{Z}_N$ are as chosen by $\mathcal{S}$.
(**v**) $\mathcal{A}$ runs the Test algorithm on $(CT_b, W_{Q'}, PP)$. This amounts to computing the following quantities:

$$\begin{aligned} A &= e(g_1^{\psi\delta}, g_1^{\tilde{\psi}\delta}) \quad \text{and} \\ B &= e(g_0^{m_0\phi} g_1^{\psi(\gamma_1+\gamma_2)}, g_0^{m_1\tilde{\phi}} g_1^{\gamma_1\tilde{\psi}}) \cdot e(g_0^{-m_1\phi} g_1^{\psi(\gamma_2-\gamma_1)}, g_0^{m_b\tilde{\phi}} g_1^{\gamma_2\tilde{\psi}}) \cdot \\ &\quad \prod_{i=3}^{n-1} e(g_0^{\phi} g_1^{\psi\gamma_i}, g_0^{\tilde{\phi}} g_1^{\gamma_i\tilde{\psi}}) \cdot e(g_0^{-(n-3)\phi} g_1^{\psi\gamma_n}, g_0^{\tilde{\phi}} g_1^{\gamma_n\tilde{\psi}}). \end{aligned}$$

If $A = B$ then $\mathcal{A}$ outputs $b' = 0$, otherwise $\mathcal{A}$ outputs $b' = 1$.

We see that $A = B$ implies $b = 0$, except with negligible probability. Hence, the adversary wins the selective FtG game with overwhelming probability of success. □

Remark 3. We give yet another attack on the scheme for even n. Let $x = (x_1, \ldots, x_n)$ be any valid message. Observe that both

$$\begin{aligned} \delta^2 &= \gamma_1(\gamma_1+\gamma_2) + \gamma_2(\gamma_2-\gamma_1) + \ldots + \gamma_{n-1}(\gamma_{n-1}+\gamma_n) + \gamma_n(\gamma_n-\gamma_{n-1}), \\ \delta^2 &= \gamma_1(\gamma_1-\gamma_2) + \gamma_2(\gamma_2+\gamma_1) + \ldots + \gamma_{n-1}(\gamma_{n-1}-\gamma_n) + \gamma_n(\gamma_n+\gamma_{n-1}) \end{aligned}$$

hold modulo q. Hence, from the ciphertext for x, pseudo-ciphertexts for both

$$\begin{aligned} \xi_1 &= (x_1+x_2, x_2-x_1, \ldots, x_{n-1}+x_n, x_n-x_{n-1}) \text{ and} \\ \xi_2 &= (x_1-x_2, x_2+x_1, \ldots, x_{n-1}-x_n, x_n+x_{n-1}) \end{aligned}$$

can be formed. Note that neither ξ_1 nor ξ_2 is orthogonal to x, while ξ_1 is orthogonal to ξ_2. Thus, for example, after setting (ξ_1, x) as the challenge pair, querying x and computing pseudo-ciphertext for ξ_2, the adversary can win the FtG game. A similar attack may also be worked out for odd n.

Remark 4. It would have been illustrating to see where exactly the proof of [29, Theorem 5.1] fails. Unfortunately no such proof is provided by the authors.

5.3 Insecurity Beyond Indistinguishability

Recall that in the ciphertext of PR scheme described in Sect. 5.1, the message components reside in the exponent and even the party who possesses the secret key does not have the ability to decrypt. Thus it is not reasonable to expect that one can attack the scheme in the sense of message recovery for high min-entropy messages. Our next attack demonstrates that an adversary is still capable of extracting significant amount of information. This will lead to a total break of the scheme when the messages come from a smaller domain, which could be the case in applications dealing with, for example, certain types of streaming data as envisaged in [29].

We assume that the adversary is allowed to make just one query and is given a valid ciphertext as response. We show how the adversary can process the given ciphertext and then utilize pairing to unmask the subgroup elements containing the message vector of any ciphertext, by working in the target group.

Attack for $n = 2$ case. Suppose the adversary makes a query $(1/2, 1/2)$ and gets the ciphertext $(c_0, c_1, c_2) = (g_1^{\psi\delta}, g_0^{\phi/2} g_1^{\psi\gamma_1}, g_0^{\phi/2} g_1^{\psi\gamma_2})$. Observe that

$$(c_0, c_1 \cdot c_2, c_2/c_1) = (g_1^{\psi\delta}, g_0^{\phi} g_1^{\psi(\gamma_1+\gamma_2)}, g_1^{\psi(\gamma_2-\gamma_1)})$$
$$(c_0, c_1/c_2, c_1 \cdot c_2) = (g_1^{\psi\delta}, g_1^{\psi(\gamma_1-\gamma_2)}, g_0^{\phi} g_1^{\psi(\gamma_1+\gamma_2)})$$

are pseudo-ciphertexts (see Definition 6) for $(1, 0)$ and $(0, 1)$, respectively, which can be computed by the adversary. We represent the formation of the two pseudo-ciphertexts, respectively, via the following two matrices with the obvious interpretation:

$$M_1 = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \quad \text{and} \quad M_2 = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}.$$

Suppose now the adversary gets a ciphertext for some unknown message $x = (x_1, x_2)$ as $(C_0, C_1, C_2) = (g_1^{\tilde{\psi}\delta}, g_0^{\tilde{\phi}x_1} g_1^{\tilde{\psi}\gamma_1}, g_0^{\tilde{\phi}x_2} g_1^{\tilde{\psi}\gamma_2})$. With the pseudo-ciphertext for $(1, 0)$, the adversary computes

$$\frac{e(C_1, c_1 \cdot c_2) e(C_2, c_2/c_1)}{e(C_0, c_0)} = \frac{e(g_0^{\tilde{\phi}x_1} g_1^{\tilde{\psi}\gamma_1}, g_0^{\phi} g_1^{\psi(\gamma_1+\gamma_2)}) \cdot e(g_0^{\tilde{\phi}x_2} g_1^{\tilde{\psi}\gamma_2}, g_1^{\psi(\gamma_2-\gamma_1)})}{e(g_1^{\tilde{\psi}\delta}, g_1^{\psi\delta})}$$
$$= e(g_0, g_0)^{\phi\tilde{\phi}x_1}.$$

Thus the adversary now possesses $(e(g_0, g_0)^{\phi\tilde{\phi}x_1}, e(g_0, g_0)^{\phi\tilde{\phi}x_2})$, after processing the pseudo-ciphertext for $(0, 1)$ similarly.

This trivially breaks the FtG security of PR scheme. Moreover, the adversary can test if x is orthogonal to any $y = (y_1, y_2)$ of his choice by checking whether

$$\left(e(g_0, g_0)^{\phi\tilde{\phi}x_1}\right)^{y_1} \cdot \left(e(g_0, g_0)^{\phi\tilde{\phi}x_2}\right)^{y_2} = 1.$$

The adversary may also test for relations among the message coordinates, like whether $x_1 = \alpha x_2$ for some α in a testable range. If x comes from a small domain

then one can exhaustively try for all candidate y to check whether x and y are orthogonal and thereby recover x with non-negligible probability.

Attack for General n. Before describing the attack, we show that many a pseudo-ciphertexts can be formed from a valid ciphertext.

Lemma 4. *For $1 \leq i \leq n$, let $M_i = ((m_{st}^{(i)}))$ be an $n \times n$ matrix defined as follows. Define $m_{it}^{(i)} = 1,\ 1 \leq t \leq n$. For $1 \leq s \leq n$, but $s \neq i$*

$$m_{st}^{(i)} = \begin{cases} 1, & t = s \\ -1, & t = i \\ 0, & \text{otherwise.} \end{cases}$$

Let $CT = (c_0, c_1, \ldots, c_n)$ be a valid ciphertext for $x = (x_1, \ldots, x_n)$. Define $\xi_i = M_i x^T$. Define $W_i = (d_0^{(i)}, d_1^{(i)}, \ldots, d_n^{(i)})$ as follows. For all j, define

$$d_j^{(i)} = \begin{cases} c_0, & \text{if } j = 0 \\ \prod_{k=1}^{n} c_k^{m_{jk}^{(i)}}, & \text{otherwise.} \end{cases}$$

Then W_i is a pseudo-ciphertext for ξ_i.

Proof. We provide details for $i = 1$ – the general case is similar. Observe that by applying M_1 to x^T one obtains $\xi_1 = (\sum_{l=1}^{n} x_l, x_2 - x_1, \ldots, x_n - x_1)$. We also note that $M_1(\gamma_1, \ldots, \gamma_n)^T = (\sum_{l=1}^{n} \gamma_l, \gamma_2 - \gamma_1, \ldots, \gamma_n - \gamma_1)$. By an easy computation:

$$\gamma_1 \sum \gamma_l + \gamma_2(\gamma_2 - \gamma_1) + \ldots + \gamma_n(\gamma_n - \gamma_1) = \delta^2 \pmod{q}.$$

Let $(g_1^{\psi\delta}, g_0^{\phi x_1} g_1^{\psi\gamma_1}, \ldots, g_0^{\phi x_n} g_1^{\psi\gamma_n})$ be a valid ciphertext for x. From this, we compute a pseudo-ciphertext for ξ_1 as

$$W_1 = (g_1^{\psi\delta}, g_0^{\phi \sum x_l} g_1^{\psi \sum \gamma_l}, g_0^{\phi(x_2 - x_1)} g_1^{\psi(\gamma_2 - \gamma_1)}, \ldots, g_0^{\phi(x_n - x_1)} g_1^{\psi(\gamma_n - \gamma_1)}).$$

Let a ciphertext for $y = (y_1, \ldots, y_n)$ be given as

$$CT_y = (c_0, c_1, \ldots, c_n) = \left(g_1^{\tilde{\psi}\delta}, g_0^{\tilde{\phi} y_1} g_1^{\tilde{\psi}\gamma_1}, \ldots, g_0^{\tilde{\phi} y_n} g_1^{\tilde{\psi}\gamma_n} \right).$$

Suppose we run Test with CT_y and W_1. It is easy to see that:

$$\frac{e(c_0, g_0^{\phi \sum x_l} g_1^{\psi \sum \gamma_l}) \prod_{l=2}^{n} e(c_l, g_0^{\phi(x_l - x_1)} g_1^{\psi(\gamma_l - \gamma_1)})}{e(g_1^{\tilde{\psi}\delta}, g_1^{\psi\delta})} = e(g_0, g_0)^{\phi\tilde{\phi}(y \cdot \xi_1)}$$

$$= 1$$

if and only if y is orthogonal to ξ_1, except with negligible probability. □

Corollary 1. *By querying the vector $x = (1/n, \ldots, 1/n)$, one can obtain the pseudo-ciphertexts for each of the unit vectors $e_i = (0, \ldots, 0, 1, 0 \ldots, 0)$ (1 in the ith place), $1 \leq i \leq n$.*

In the following theorem we describe the attack for general n.

Theorem 6. *Suppose in the proposed PR scheme of [29] the adversary is allowed to make one query for any message of its choice. Then, given a valid ciphertext for any unknown message $(x_1, \ldots, x_n)$, the adversary can extract the tuple of elements $(\eta; \eta^{\phi' x_1}, \ldots, \eta^{\phi' x_n})$ for some η belonging to the order-p subgroup of $\mathbb{G}_T$ and $\phi' \in \mathbb{Z}_N$.*

Proof. Let $(d_0, d_1, \ldots, d_n) = (g_1^{\psi\delta}, g_0^{\phi/n} g_1^{\psi\gamma_1}, \ldots, g_0^{\phi/n} g_1^{\psi\gamma_n})$ be the ciphertext for the queried message $(1/n, \ldots, 1/n)$. A ciphertext CT_x for some unknown $x = (x_1, \ldots, x_n)$ is given to the adversary, where $CT_x = (c_0, c_1, \ldots, c_n) = \left(g_1^{\tilde{\psi}\delta}, g_0^{\tilde{\phi} x_1} g_1^{\tilde{\psi}\gamma_1}, \ldots, g_0^{\tilde{\phi} x_n} g_1^{\tilde{\psi}\gamma_n}\right)$.

Notice that the unit vector e_i can be written as $e_i = M_i(1/n, \ldots, 1/n)^T$. From Lemma 4, the adversary can compute $W_i = (w_0^{(i)}, w_1^{(i)}, \ldots, w_n^{(i)})$, a pseudo-ciphertext for e_i as

$$W_i = \left(g_1^{\psi\delta}, g_1^{\psi(\gamma_1 - \gamma_i)}, \ldots, g_1^{\psi(\gamma_{i-1} - \gamma_i)}, g_0^{\phi} g_1^{\psi(\sum \gamma_j)}, g_1^{\psi(\gamma_{i+1} - \gamma_i)}, \ldots, g_1^{\psi(\gamma_n - \gamma_i)}\right).$$

The adversary further computes $\left(\prod_{l=1}^{n} e(c_l, w_l^{(i)})\right) / e(c_0, w_0^{(i)}) = e(g_0, g_0)^{\phi\tilde{\phi} x_i}$. In a similar fashion, the adversary obtains a tuple over the order-p subgroup of the target group $\mathbb{G}_T$ as $\Omega = \left(e(g_0, g_0)^{\phi\tilde{\phi} x_1}, \ldots, e(g_0, g_0)^{\phi\tilde{\phi} x_n}\right)$. The adversary now computes $\eta := \left(\prod_{i=1}^{n} e(d_i, d_i)\right) / e(d_0, d_0) = e(g_0, g_0)^{\phi^2/n}$. Rewriting Ω as powers of η, s/he gets $\Omega = (\eta^{\phi' x_1}, \ldots, \eta^{\phi' x_n})$. Hence the result. □

As already pointed out for the $n = 2$ case, the above argument shows that the adversary is capable of extracting a lot of information from the ciphertext of any unknown message vector x. Recall that the fundamental reason for having PPTag in symmetric setting is to prevent the adversary from being able to test whether a ciphertext of some unknown message satisfies a certain property and thereby learn some non-trivial information about the message. Given Ω the adversary can precisely do that and thus the scheme in [29] defeats the very purpose of symmetric key property preserving encryption.

6 Concluding Remarks

In this work we perform a comprehensive (crypt)analysis of property preserving symmetric encryption. On the definitional front, we revisit the FtG and LoR separation result in [29]. To do that we show equality property captures property P_{qr} used in the separation results and provide a simple construction for equality property to demonstrate that the separation results are non-vacuous. Based on the security attributes of our construction and its generalization we raised the pertinent question of whether the separation results actually indicate any real world difference between the two notions of security and argue for a

property specific study of the security notions. Continuing further in this direction, we see that an LoR-secure scheme may be constructed from a so-called weaker FtG-secure one for orthogonality. We demonstrate several attacks on the PPTag scheme for testing orthogonality from [29] refuting the claim that the scheme is provably secure. Our main attack successfully unmasks the subgroup elements where the message vector is mapped to and thereby points to greater vulnerability beyond the notion of indistinguishability.

Acknowledgements

The authors wish to thank the anonymous reviewers for their valuable comments. The authors also thank Chethan Kamath, Neal Koblitz, Alfred Menezes, Omkant Pandey, Yannis Rouselakis and Palash Sarkar for their comments on a preliminary version of this work.

Appendix A

We first argue the separation result for polynomial size message space and use it to prove the general case.

A.1 Separation Result for Polynomial Size Message Space

Let $\mathcal{M} = \{\alpha_i \mid 1 \leq i \leq l\}$ be the message space and each α_i can be represented by a z-bit string where $z = \lceil \log_2 l \rceil$. We argue the separation result $\mathsf{FtG} \nrightarrow \mathsf{LoR}$ for equality property in the case where l is polynomial in security parameter. Let Π be an FtG secure PPTag scheme for equality over $\mathcal{M}$. From this scheme we construct another scheme Π' for realizing the same property as follows.

1. $\Pi' \cdot \mathsf{Setup}(1^\lambda)$: The public parameters for Π' are exactly those of Π. The secret key SK' of Π' comprises of that of Π and a set of binary strings $\{t_i \mid 1 \leq i \leq l\}$, where each t_i is of length z and chosen independently and uniformly at random.
2. $\Pi' \cdot \mathsf{Encrypt}(PP, SK, m)$: Suppose $m = \alpha_i$; the algorithm chooses a random bit b and the output is defined as

$$\Pi'.\mathsf{Encrypt}(PP, SK', m) = \begin{cases} (\Pi.\mathsf{Encrypt}(PP, SK, m), b, t_i), & \text{if } b = 0, \\ (\Pi.\mathsf{Encrypt}(PP, SK, m), b, t_i \oplus \alpha_i), & \text{o.w.} \end{cases}$$

3. $\Pi'.\mathsf{Test}(CT_1, CT_2, PP)$: Same as that of Π, where only the relevant parts of the ciphertexts are used.

Lemma 5. *The scheme Π' is not* FtG *secure implies Π is not* FtG *secure.*

Proof. Consider a valid FtG adversary for Π', denoted by $\mathcal{A}$. We describe how an FtG adversary $\mathcal{B}$ for Π, with same advantage as that of $\mathcal{A}$ and which internally uses $\mathcal{A}$, can be constructed.
(**i**). $\mathcal{B}$ forwards to $\mathcal{A}$ whatever is received from its own challenger as public parameters of Π and initializes an empty table T.
(**ii**). Whenever $\mathcal{A}$ makes an encryption query for $m = \alpha_i$, $1 \leq i \leq l$, $\mathcal{B}$ forwards it to the simulator of Π. On receiving ct from the simulator, $\mathcal{B}$ checks whether the same query was made earlier or not. If the query is made for the first time, then it chooses $t \in_R \{0,1\}^z$, sets $t_i = t$ and updates the table T with $\{(i, t_i)\}$. Else, $\mathcal{B}$ reuses corresponding t_i from T. Finally $\mathcal{B}$ chooses a random bit b and forwards to $\mathcal{A}$

$$CT = \begin{cases} (ct, b, t_i), \text{ if } b = 0, \\ (ct, b, t_i \oplus \alpha_i), \text{ if } b = 1. \end{cases}$$

(**iii**). After a certain number of encryption queries $\mathcal{A}$ outputs the challenge (m_0^*, m_1^*). Two cases arise with respect to the challenges, which we describe below.
Case 1: The challenge messages m_0^* and m_1^* are equal.
Case 2: The challenge messages m_0^* and m_1^* are different. In this case, the adversary cannot make encryption query for these two messages.

$\mathcal{B}$ forwards (m_0^*, m_1^*) to the simulator of Π and gets ct^*. If the challenge messages are equal (**Case 1**), then (ct^*, b, val) may be computed by $\mathcal{B}$ in the same way as it responses to the encryption queries. If the challenge messages are different (**Case 2**), then none of m_0^* and m_1^* have been queried previously. $\mathcal{B}$ returns (ct^*, b, t^*), where $b \in_R \{0,1\}$ and $t^* \in_R \{0,1\}^z$. Let $\alpha_j \in \{m_0^*, m_1^*\}$ be the unknown message chosen by the simulator of Π. The strategy adopted by $\mathcal{B}$ gives a perfect simulation. This is because if $b = 0$ then t^* can be set as t_j whereas for $b = 1$, t^* can be set as $t_j \oplus \alpha_j$.
(**iv**). $\mathcal{B}$ follows the same strategy of step (ii) above to answer all the subsequent encryption queries of $\mathcal{A}$.
(**v**). When $\mathcal{A}$ outputs a bit b' and halts, so does $\mathcal{B}$.

Notice that all the ciphertexts which $\mathcal{B}$ computes for forwarding to $\mathcal{A}$ are properly distributed. $\mathcal{B}$ is a polynomial time algorithm and provides a perfect simulation. Hence, advantage of $\mathcal{B}$ is equal to that of $\mathcal{A}$. □

Lemma 6. *There is an* LoR *adversary for the scheme* Π' *with non-negligible advantage.*

Proof. A valid LoR adversary sets as u challenges the same pair of the form (m_0, m_1), with $m_0 \neq m_1$. Equality pattern is clearly preserved between the left and right sequences. If the challenger outputs two ciphertexts for which the b-values are distinct, then the adversary can immediately distinguish the two sequences. The advantage will be $1 - 2^{-u+1}$. □

The strategy outlined in the above proof can be used to prove Lemma 2.

A.2 Proof of Lemma 1

Recall that in the FtG game $\mathcal{A}$ makes a polynomial number of encryption oracle query m_i, $1 \leq i \leq q$, and a single challenge query (m_0^*, m_1^*) maintaining the equality pattern. Two cases arise depending upon whether the challenge messages m_0^* and m_1^* are equal or not. If $m_0^* = m_1^*$ then it is easy to see that any advantage of $\mathcal{A}$ against Π' translates into the same advantage against Π. Hence, we consider the case when $m_0^* \neq m_1^*$. Note that in this case none of the queries to the encryption oracle m_i is equal to m_b^*, for $b \in \{0, 1\}$. Otherwise, the equality pattern of the two sequences will be different allowing $\mathcal{A}$ to trivially distinguish.

Let Game_0 correspond to the queries $(m_1, \ldots, m_i, m_0^*, m_{i+1}, \ldots, m_q)$ while Game_1 to queries $(m_1, \ldots, m_i, m_1^*, m_{i+1}, \ldots, m_q)$ made by the adversary. Suppose $\mathcal{A}$ can distinguish whether it is playing Game_0 or Game_1 with a non-negligible advantage $\epsilon_{\Pi'}$. The proof will proceed through a hybrid argument. Given an adversary $\mathcal{A}$ against Π' we construct a series of four games and then show that if $\mathcal{A}$ can distinguish between any two consecutive games then we can construct either a PRF adversary against $\mathcal{F}$ or an FtG adversary against Π.

Game_0 The challenger runs the Setup algorithm of Π' and gives the PP to $\mathcal{A}$ and keeps the secret key $SK' = (SK, k)$ to itself. The challenger computes the ciphertext corresponding to $(m_1, \ldots, m_i, m_0^*, m_{i+1}, \ldots, m_q)$ using SK' as per the encryption algorithm of Π' and give them to $\mathcal{A}$.

Game_A The challenger runs the Setup algorithm of Π and gives the PP to $\mathcal{A}$ and keeps the secret key SK of Π to itself. Note that the challenger does not generate the PRF key k; instead it will maintain a table $\mathbb{T} = \langle x_i, y_i \rangle$ where x_i and y_i are two z-bit strings. The first entry in each row of $\mathbb{T}$ corresponds to the messages queried by $\mathcal{A}$ while the second entry is a random bit-string. The table is initially empty. Whenever $\mathcal{A}$ makes an encryption query for a message x, the challenger first checks whether there is a corresponding entry in $\mathbb{T}$. If not, it chooses a random z-bit string y and enters (x, y) in the table $\mathbb{T}$ sorted according to the first entry. $\mathcal{A}$ makes encryption queries for $(m_1, \ldots, m_i, m_0^*, m_{i+1}, \ldots, m_q)$. To answer the query of $\mathcal{A}$ for a message, say x, the challenger computes the ciphertext of Π on x and then uses the corresponding random string y from the entry (x, y) in $\mathbb{T}$ to create a ciphertext of Π'. Note that $\mathcal{A}$ makes at most q encryption oracle queries and a single challenge query. So the size of $\mathbb{T}$ is $O(q)$ and hence the challenger can consistently respond to all the queries of $\mathcal{A}$.

Claim 7. *If $\mathcal{A}$ can decide with a non-negligible advantage whether it is playing Game_0 or Game_A then we can construct a PRF distinguisher with the same advantage.*

Recall that in the PRF security game we are provided with an oracle which is either a function from the PRF family or a random function. In the former case the challenger will be playing Game_0 while in the latter case it'll be playing Game_A. Hence, any advantage of $\mathcal{A}$ in distinguishing between the two games translate into the same advantage of the challenger in breaking the PRF security.

Game_1 (resp. Game_B) will be identical to Game_0 (resp. Game_A) except the fact that $\mathcal{A}$ now queries with $(m_1, \ldots, m_i, m_1^*, m_{i+1}, \ldots, m_q)$. An identical argument as in the claim above establishes that any advantage of $\mathcal{A}$ in deciding whether it is playing Game_1 or Game_B translates into the same PRF advantage for the challenger.

Note that the only difference in Game_A and Game_B is in the challenge ciphertext (corresponding to m_0^* and m_1^*). The challenge is computed by calling the encryption algorithm of Π and appending either a random bit string or a one-time encryption of m_b^* (using that random string). Hence, an adversary distinguishing between Game_A and Game_B can be converted into an adversary breaking the FtG security of Π. As there are only polynomial many queries, this case is the same as the one where there are only small (polynomial in λ) number of messages. This case can be easily handled by using random strings. We have already given the analysis in the proof of Lemma 5.

References

1. Shashank Agrawal, Shweta Agrawal, Saikrishna Badrinarayanan, Abishek Kumarasubramanian, Manoj Prabhakaran, and Amit Sahai. Function Private Functional Encryption and Property Preserving Encryption : New Definitions and Positive Results. Cryptology ePrint Archive, Report 2013/744, 2013. http://eprint.iacr.org/.
2. Georgios Amanatidis, Alexandra Boldyreva, and Adam O'Neill. Provably-Secure Schemes for Basic Query Support in Outsourced Databases. In Steve Barker and Gail-Joon Ahn, editors, *Data and Applications Security XXI, 21st Annual IFIP WG 11.3 Working Conference on Data and Applications Security, Redondo Beach, CA, USA, July 8-11, 2007, Proceedings*, volume 4602 of *Lecture Notes in Computer Science*, pages 14–30. Springer, 2007.
3. E. Bach and J.O. Shallit. *Algorithmic Number Theory*. Foundations of Computing. MIT Press, 1996.
4. Mihir Bellare, Anand Desai, E. Jokipii, and Phillip Rogaway. A Concrete Security Treatment of Symmetric Encryption. In *FOCS*, pages 394–403. IEEE Computer Society, 1997.
5. Mihir Bellare, Anand Desai, David Pointcheval, and Phillip Rogaway. Relations Among Notions of Security for Public-Key Encryption Schemes. In Hugo Krawczyk, editor, *CRYPTO*, volume 1462 of *Lecture Notes in Computer Science*, pages 26–45. Springer, 1998.
6. Mihir Bellare and Adam O'Neill. Semantically-Secure Functional Encryption: Possibility Results, Impossibility Results and the Quest for a General Definition. In Michel Abdalla, Cristina Nita-Rotaru, and Ricardo Dahab, editors, *CANS*, volume 8257 of *Lecture Notes in Computer Science*, pages 218–234. Springer, 2013.
7. Mihir Bellare, Thomas Ristenpart, Phillip Rogaway, and Till Stegers. Format-Preserving Encryption. In Michael J. Jacobson Jr., Vincent Rijmen, and Reihaneh Safavi-Naini, editors, *Selected Areas in Cryptography*, volume 5867 of *Lecture Notes in Computer Science*, pages 295–312. Springer, 2009.
8. Alexandra Boldyreva, Nathan Chenette, Younho Lee, and Adam O'Neill. Order-Preserving Symmetric Encryption. In Antoine Joux, editor, *EUROCRYPT*, volume 5479 of *Lecture Notes in Computer Science*, pages 224–241. Springer, 2009.

9. Alexandra Boldyreva, Nathan Chenette, Younho Lee, and Adam ONeill. Order-Preserving Symmetric Encryption. Cryptology ePrint Archive, Report 2012/624, 2012. http://eprint.iacr.org/.
10. Alexandra Boldyreva, Nathan Chenette, and Adam O'Neill. Order-Preserving Encryption Revisited: Improved Security Analysis and Alternative Solutions. In Phillip Rogaway, editor, *CRYPTO*, volume 6841 of *Lecture Notes in Computer Science*, pages 578–595. Springer, 2011.
11. Dan Boneh and Xavier Boyen. Efficient Selective Identity-Based Encryption Without Random Oracles. *J. Cryptology*, 24(4):659–693, 2011.
12. Dan Boneh, Xavier Boyen, and Eu-Jin Goh. Hierarchical Identity Based Encryption with Constant Size Ciphertext. In Ronald Cramer, editor, *EUROCRYPT*, volume 3494 of *Lecture Notes in Computer Science*, pages 440–456. Springer, 2005.
13. Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky, and Giuseppe Persiano. Public Key Encryption with Keyword Search. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT*, volume 3027 of *Lecture Notes in Computer Science*, pages 506–522. Springer, 2004.
14. Dan Boneh, Ananth Raghunathan, and Gil Segev. Function-Private Identity-Based Encryption: Hiding the Function in Functional Encryption. In Canetti and Garay [18], pages 461–478.
15. Dan Boneh, Ananth Raghunathan, and Gil Segev. Function-Private Subspace-Membership Encryption and Its Applications. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT (1)*, volume 8269 of *Lecture Notes in Computer Science*, pages 255–275. Springer, 2013.
16. Dan Boneh, Amit Sahai, and Brent Waters. Functional Encryption: Definitions and Challenges. In Yuval Ishai, editor, *TCC*, volume 6597 of *Lecture Notes in Computer Science*, pages 253–273. Springer, 2011.
17. Dan Boneh and Brent Waters. Conjunctive, Subset, and Range Queries on Encrypted Data. In Salil P. Vadhan, editor, *TCC*, volume 4392 of *Lecture Notes in Computer Science*, pages 535–554. Springer, 2007.
18. Ran Canetti and Juan A. Garay, editors. *Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part II*, volume 8043 of *Lecture Notes in Computer Science*. Springer, 2013.
19. Angelo De Caro, Vincenzo Iovino, and Giuseppe Persiano. Fully Secure Anonymous HIBE and Secret-Key Anonymous IBE with Short Ciphertexts. In Marc Joye, Atsuko Miyaji, and Akira Otsuka, editors, *Pairing*, volume 6487 of *Lecture Notes in Computer Science*, pages 347–366. Springer, 2010.
20. Sanjit Chatterjee and M. Prem Laxman Das. Property Preserving Symmetric Encryption Revisited. Cryptology ePrint Archive, Report 2013/830, 2013. http://eprint.iacr.org/.
21. Sanjam Garg, Craig Gentry, Shai Halevi, Amit Sahai, and Brent Waters. Attribute-Based Encryption for Circuits from Multilinear Maps. In Canetti and Garay [18], pages 479–499.
22. Craig Gentry. Fully Homomorphic Encryption Using Ideal Lattices. In Michael Mitzenmacher, editor, *STOC*, pages 169–178. ACM, 2009.
23. Craig Gentry, Shai Halevi, and Nigel P. Smart. Fully Homomorphic Encryption with Polylog Overhead. In Pointcheval and Johansson [30], pages 465–482.
24. Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based. In Ran Canetti and Juan A. Garay, editors, *CRYPTO (1)*, volume 8042 of *Lecture Notes in Computer Science*, pages 75–92. Springer, 2013.

25. Shafi Goldwasser, S. Dov Gordon, Vipul Goyal, Abhishek Jain, Jonathan Katz, Feng-Hao Liu, Amit Sahai, Elaine Shi, and Hong-Sheng Zhou. Multi-Input Functional Encryption. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT*, volume 8441 of *Lecture Notes in Computer Science*, pages 578–602. Springer, 2014.
26. Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data. In Ari Juels, Rebecca N. Wright, and Sabrina De Capitani di Vimercati, editors, *ACM Conference on Computer and Communications Security*, pages 89–98. ACM, 2006.
27. Sudipto Guha, Adam Meyerson, Nina Mishra, Rajeev Motwani, and Liadan O'Callaghan. Clustering Data Streams: Theory and Practice. *IEEE Trans. Knowl. Data Eng.*, 15(3):515–528, 2003.
28. Jonathan Katz, Amit Sahai, and Brent Waters. Predicate Encryption Supporting Disjunctions, Polynomial Equations, and Inner Products. In Nigel P. Smart, editor, *EUROCRYPT*, volume 4965 of *Lecture Notes in Computer Science*, pages 146–162. Springer, 2008.
29. Omkant Pandey and Yannis Rouselakis. Property Preserving Symmetric Encryption. In Pointcheval and Johansson [30], pages 375–391.
30. David Pointcheval and Thomas Johansson, editors. *Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings*, volume 7237 of *Lecture Notes in Computer Science*. Springer, 2012.
31. Phillip Rogaway and Thomas Shrimpton. A Provable-Security Treatment of the Key-Wrap Problem. In Serge Vaudenay, editor, *Advances in Cryptology - EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28 - June 1, 2006, Proceedings*, volume 4004 of *Lecture Notes in Computer Science*, pages 373–390. Springer, 2006.
32. Emily Shen, Elaine Shi, and Brent Waters. Predicate Privacy in Encryption Systems. In Omer Reingold, editor, *TCC*, volume 5444 of *Lecture Notes in Computer Science*, pages 457–473. Springer, 2009.
33. Elaine Shi, John Bethencourt, Hubert T.-H. Chan, Dawn Xiaodong Song, and Adrian Perrig. Multi-Dimensional Range Query over Encrypted Data. In *IEEE Symposium on Security and Privacy*, pages 350–364. IEEE Computer Society, 2007.
34. Dawn Xiaodong Song, David Wagner, and Adrian Perrig. Practical Techniques for Searches on Encrypted Data. In *IEEE Symposium on Security and Privacy*, pages 44–55. IEEE Computer Society, 2000.
35. Marten van Dijk, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. Fully Homomorphic Encryption over the Integers. In Henri Gilbert, editor, *EUROCRYPT*, volume 6110 of *Lecture Notes in Computer Science*, pages 24–43. Springer, 2010.