PROVABLE SECURITY OF RM Code BASED FHE SCHEME

Ratnakumari Challa[1], Vijayakumari G[2]

[1] Department of Computer Science and Engineering, RGUKT RK Valley, Kadapa, Andhra Pradesh, India
[2]Department of Computer Science and Engineering, JNTUH, Hyderabad, Telangana, India

E-mail address: ratnamala3784@gmail.com, vijayakumarigunta@gmail.com

Received ## Mon. 20##, Revised ## Mon. 20##, Accepted ## Mon. 20##, Published ## Mon. 20##

Abstract: Based on the hardness of decoding noisy codewords, coding theory was recently recognized as a potential and well established cryptographic primitive. While keeping the benefit of decoding noisy codewords, coding theory may be utilized to develop implementable homomorphic encryption systems with restricted capacity. Using error correction codes (ECC), a code-based homomorphic encryption system [1] [2] has been proposed and implemented. These designs have evolved into intriguing attempts at developing an implementable partial homomorphic encryption (PHE) method that does not require a computationally demanding bootstrapping phase. The techniques allow for unlimited number of addition operations while keeping the ciphertext size constant. However, using the existing schemes, multiplicative homomorphism is not instantly malleable. The current study is an effort to provide a system for designing an fully homomorphic encryption (FHE) utilizing any error correcting code based on coding theory. The development of a code-based homomorphic encryption scheme with homomorphic addition operations is a straightforward approach, however homomorphic multiplication is not possible with existing developments. The current work tries to show that a Reed-Muller (RM) error correcting code may be utilized to successfully develop multiplication operations, therefore transforming it fully homomorphic. While keeping the benefit of computational cost of ECC, the code-based homomorphic encryption method is effectively turned into the RM code -based FHE scheme (RMFHE). The RMFHE structure has been theoretically proven, and extensive experiments at various security levels have been conducted [14]. In this study, the hardness of RMFHE scheme to an attacker is reduced to the intractable decisional synchronized codeword problem (DCSP).

Keywords: Homomorphic encryption, Coding theory, Decisional synchronized codeword problem (DSCP), Intractable, Erroneous codewords, Error correction codes, Indistinguishability under chosen plain text attack (IND-CPA)

1. INTRODUCTION

Homomorphic encryption (HE) is a powerful cryptographic primitive used in cloud computing to provide privacy and secrecy. In recent decades, HE in cloud computing has been an active study topic. To ensure privacy and security, cloud computing employs a unique type of encryption [3]. The Homomorphic encryption enables other parties to perform computations on ciphertext without knowing its underlying information. Fully homomorphic encryption (FHE) is a kind of HE that permits arbitrary calculations on encrypted data.

Gentry created the first FHE in 2009. Later, with specific security assumptions, multiple theoretical and practical FHE scheme designs were built based on various mathematical and theoretical difficulties.

Later advances to increase the efficiency of practical Gentry job implementation. It has been a key focus of study to improve the efficiency of Gentry's work, which was created based on ideal lattices, by Smart et al. [5], Gentry and Halevi [4].

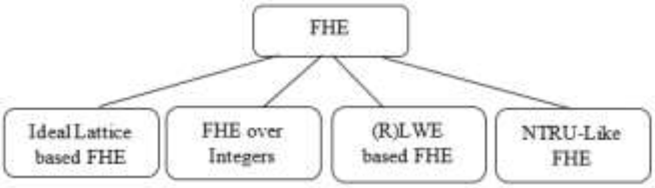

Figure. 1: FHE families after Gentry's break through

E-mail:ratnamala3784@gmail.com

Since the development of the NTRU system, cryptography has also utilized ideal lattices [6]. Because they feature an extra structure that improves their representation and allows for quicker computations, ideal lattices are more powerful than regular lattices.

Similarly, Van Dijk et al. [7] developed FHE over integers based on Approximate GCD problems, Brakerski et al. [8] developed FHE over integers based on Ring Learning with Error (RLWE) issues, and López-Alt et al. [9] proposed a NTRU-like FHE. These are the principal varieties of FHE families, and all related works are classified into four major groups, as shown in Figure 1.

Apart from these constructions, based on Gentry's break through seminal work [10], a homomorphic encryption approach based on coding theory concept [1] [11] has been developed.

The core public key concepts are limited in code-based public key HE systems [1]. Armknecht et al. proposed an alternative HE technique based on error correcting codes (ECC). The system is confined to the building of symmetric key homomorphic encryption due to structural considerations; it allows a specific number of new encryptions; and it is incompatible with bootstrapping. However, with certain pre-computations, the approach may be implemented using any error correction technique, with practical execution of various functions like encipherment, decipherment and homomorphic operations completed in less than one second [2]. On the contrary, the degree of the product codeword will be raised for homomorphic multiplication operations, implying that decoding the product codeword does not provide the product of its corresponding plaintexts. As a result, the method has not been improved to allow for unlimited multiplication operations. Under known attacks, a homomorphic encryption system based on Reed-Muller codes has been proposed and studied [13].

The FHE families are the primary attempts of FHE advances to solve concerns of efficiency, security, and practicality. The majority of those strategies focused on reducing key and ciphertext size, eliminating squashing techniques, accomplishing FHE creation without bootstrapping, reducing message expansion using packed ciphertexts, and obtaining mechanism with straightforward, easy, and quick calculations. Nonetheless, designing a technique which can manage all these challenges efficiently remains open.

Contribution: The current work reported in this study aims to provide a fully Homomorphic encryption scheme with simple and quick calculations that is suitable for secure cloud-based activities. Appropriately, a new technique for development of FHE based on error correcting codes with simple decryption and reasonable time complexities is proposed [14]. The primary goal of this effort is to concentrate on the provable security of the RMFHE scheme, i.e., to make cracking the RMFHE scheme as difficult as solving the DSCP problem, which is known to be intractable.

2. RM CODE (RMC) BASED HE SCHEME

The RMFHE system is a code-based FHE symmetric key method that employs RM error-correcting codes. The RM code-based systems use the fundamental notion of coding theory-based encryption, in which the plaintext is encoded first to produce the codeword. The erroneous codeword is then computed by adding artificial errors to the codeword, which is then considered ciphertext, as seen in figure 2.

In the ciphertext, error places are deemed bad, whereas rest (error-free positions) are considered good. A symmetric key determines the bad and good positions for each encryption. It is also feasible to fulfil the homomorphism by preserving error-free locations in the ciphertext using component-wise operations.

An $[n, k, d\ ]$ – code with $k < n,$ is a linear subspace $\mathcal{C} \in \mathbb{F}_q^n$ finite field of order q, and it is true that $\boldsymbol{w}_1 + \boldsymbol{w}_2 \in \mathcal{C}$ and $|\boldsymbol{w}_1 + \boldsymbol{w}_2\ | \geq d$ for any two codewords $\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathcal{C}$. Here, every codewords $\boldsymbol{w} \in \mathcal{C}$ are deemed *Error-free* or *Correct codewords*, but $\boldsymbol{w} \in \mathbb{F}_q^n \backslash \mathcal{C}$ are assumed as *Erroneous codewords*.

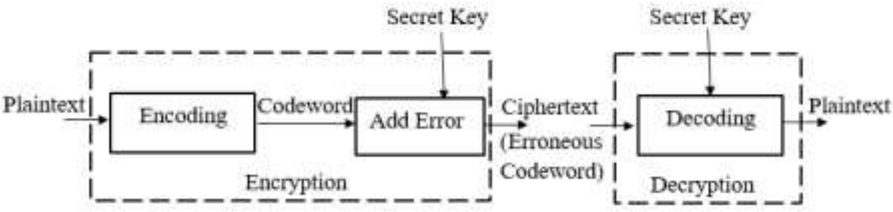

Figure. 2: Basic idea of RMFHE

Erroneous codewords are represented as $\boldsymbol{w}' = \boldsymbol{w} + \boldsymbol{e}$, where $\boldsymbol{e}\ \in \mathbb{F}_q^n \backslash \boldsymbol{0}$ is refereed to as the *Error codeword* or *Error vector*. In the *error vector*, the locations where errors taken place are called bad locations and are

E-mail:ratnamala3784@gmail.com

represented as $supp(\boldsymbol{e})$, whereas the rest are assumed as error-free or good positions and are represented as $I = [n]\backslash supp(\boldsymbol{e})$. For a linear code $\mathcal{C}$, the set of erroneous codeword with error-free locations described by I is then defined as follows:

$$\mathcal{C}(I) = \{\boldsymbol{w}+\boldsymbol{e} \mid \boldsymbol{w} \in \mathcal{C}, \boldsymbol{e} \in \mathbb{F}_q^n\backslash\boldsymbol{0}, supp(\boldsymbol{e}) \subseteq [n]\backslash I\}$$

The message is recovered using a decoding algorithm from an erroneous codeword with known error positions only if $|\boldsymbol{e}| < d$. Decoding of $\boldsymbol{w}' \in \mathcal{C}(I)$ is represented by $Decode(\boldsymbol{w}', I)$.

The componentwise addition of two codewords $\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathcal{C}$ results in a codeword $\boldsymbol{w} \in \mathcal{C}$ which is considered as valid one, and sum of its underlying messages is obtained by decoding the resultant (valid) codeword, because the RMC-based HE scheme supports additive homomorphism.

The scheme's additive homomorphism is represented as $Decode\left(\sum_{j=1}^{l}\boldsymbol{w}_j, I\right) = \sum_{j=1}^{l}\boldsymbol{m}_j$, assuming $|I|$ error-free locations are enough for uniquely decoding the codeword, i.e., number of bad locations $(n\backslash|I|)$ are less than d, where $\boldsymbol{w}_j \in \mathcal{C}(I)$.

In contrast to homomorphic addition, componentwise multiplication of two codewords $\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathcal{C}$ may not produce a valid codeword and its decoding does not imply the product of its underlying messages. As a result, a proposal is presented to adopt a new structural representation of Reed-Muller codes to allow homomorphic multiplication by converting RMC-based HE schemes to FHE.

3. RMC BASED FHE SCHEME (RMFHE)

The primary goal of this study is to develop an FHE based on coding theory and error correcting codes. To accommodate unlimited Mod 2 multiplication operations over ciphertexts, the codeword is encoded as a matrix rather than a vector, which can be interpreted as a modification to the representation of the original codeword, and thus is named as Codeword matrix. Expanded codeword is an alternative name for Codeword matrix. The following is the formal definition of an expanded RM codeword:

Definition. 3.1 Codeword matrix:

Let $\mathcal{C} \subset \mathbb{F}_2^n$ denotes a $[n,k,d]$ a linear code, with minimum hamming distance d, then for $k < n$. RMC in this instance, is represented by $\mathcal{C}_M$ and it consists of a collection of codeword matrices $\{\mathrm{W}_1, \mathrm{W}_2, \mathrm{W}_3, \ldots\}$, where $\mathrm{W}_i \in (\mathbb{F}_2^n)^k$ obtained from $\boldsymbol{m}_i \times \mathrm{G}_{rm}$ for the message $\boldsymbol{m}_i \in \mathbb{F}_2^k$, $i = 1,2,3,\ldots$, and $\mathrm{G}_{rm} \in (\mathbb{F}_2^n)^k$ is a generator matrix of $RM(r,m)$. For two codewords $\mathrm{W}_1, \mathrm{W}_2 \in \mathcal{C}_M$, $\mathrm{W}_1 + \mathrm{W}_2 \in \mathcal{C}_M$ and $\mathrm{W}_1 \cdot \mathrm{W}_2 \in \mathcal{C}_M$ i.e., both homomorphic addition and multiplication applies on the code $\mathcal{C}_M$.

A original RM codeword $\boldsymbol{w} \in \mathcal{C}$ corresponding to the codeword matrix $\mathrm{W} \in \mathcal{C}_M$, is immediately decodable using Reed-Muller majority logic.

Definition. 3.2 Conversion of Codeword matrix to original RM

Let $\mathrm{W} \in \mathcal{C}_M$ be a codeword matrix with set of vectors $\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{w}_3, \ldots, \boldsymbol{w}_k$, such that $\mathrm{W} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{w}_3, \ldots, \boldsymbol{w}_k) \in (\mathbb{F}_2^n)^k$, $\boldsymbol{w}_i \in \mathbb{F}_2^n$, $\forall i = 1,2,\ldots,k$. The original RM codeword $\boldsymbol{w} \in \mathcal{C}$ is then produced by summing all of these vectors i.e., $\boldsymbol{w} = \boldsymbol{w}_1 + \boldsymbol{w}_2 + \boldsymbol{w}_3 + \cdots + \boldsymbol{w}_k$. By using majority logic decoding, the plaintext is computed from RM codeword $\boldsymbol{w} \in \mathcal{C}$.

The code $\mathcal{C}_M$ supports both addition and multiplication operations over the codeword matrices. The addition of two codewords $\mathrm{W}_1, \mathrm{W}_2 \in \mathcal{C}_M$ results in a codeword $\mathrm{W} \in \mathcal{C}_M$, and its decoding yields the Mod 2 addition of its underlying messages and is denoted as $Decode\left(\sum_{j=1}^{l}\mathrm{W}_j\right) = \sum_{j=1}^{l}\boldsymbol{m}_j$ where $\mathrm{W}_j \in \mathcal{C}_M$ and $\boldsymbol{m}_j \in \mathbb{F}_2^k$. Similarly, multiplication two codewords $\mathrm{W}_1, \mathrm{W}_2 \in \mathcal{C}_M$ produce a codeword $\mathrm{W} \in \mathcal{C}_M$, and its decoding produce the multiplication of its corresponding messages and is denoted by $Decode\left(\prod_{j=1}^{l}\mathrm{W}_j\right) = \prod_{j=1}^{l}\boldsymbol{m}_j$ where $\mathrm{W}_j \in \mathcal{C}_M$ and $\boldsymbol{m}_j \in \mathbb{F}_2^k$.

In this scenario, codeword matrix $\mathrm{W} \in \mathcal{C}_M$ considered as an error-free, so its original RM codeword $\boldsymbol{w} \in \mathcal{C}$ is also error free. As a result, during majority logic decoding, all checksums of that codeword give the same value. In the codeword matrix W, if errors assumed in any columns imply the errors in its corresponding RM codeword at the same (column) positions. As a result, the relevant checksums (which are associated with errors) yield result that differ from other checksums. As a result, the plaintext bit shall be computed from the majority of all checksums. Because ECC are capable to auto correct, for each codeword, this decoding can correct up to $(\frac{d}{2} - 1)$ errors automatically. To convert this RM encoding to encryption, in specified (fixed) columns indicated by the secret key, a set of artificial errors must be inserted into the codeword matrix.

The error (bad) and error-free (good) columns may be recognized using the secret key in this situation. In our scenario, the codeword matrix with errors is referred to as erroneous codeword matrix which is represented by W', and is assumed as the ciphertext. These erroneous codeword matrices are represented by W'. For it to be secure, the number of error columns containing artificial errors must exceed its auto correcting capability.

The majority logic decoding can now be considered as decryption with known error and error-free positions (using secret key). In this case, the checksums associated with error positions are ignored, and the plaintext bits are derived as a majority of the rest where they all yield the same value. The codeword matrix's number of error columns must be fewer than the code's distance d. More than d errors in the codeword matrix result in incorrect decryption.

Definition 3.3. Erroneous codeword matrix:

Let $\mathrm{W} \in \mathcal{C}_M$ is codeword matrix (error-free), for an error matrix $\mathrm{E} \in \mathbb{E}$, the erroneous codeword matrix is described as $\mathrm{W}' = \mathrm{W} + \mathrm{E}$.

An error matrix E consists of collection of k vectors and each of size n bits, i.e., $\mathrm{E} = \{\boldsymbol{e}_1, \boldsymbol{e}_2, \dots, \boldsymbol{e}_k\}$, where $\boldsymbol{e}_1$ is a null vector and $\boldsymbol{e}_i \in \mathbb{F}_2^n\ \forall i = 2, 3, \dots, k$ is an error vector. For all error vectors $\boldsymbol{e}_i \in \mathbb{F}_2^n$ where $i = 2, 3, \dots, k$, the error positions are fixed by secret key and these positions are represented by $supp(\boldsymbol{e}_i)$. The rest are error-free (good) locations, and are indicated by $I = [n] \backslash supp(\boldsymbol{e}_i)$. Each error vector is made up of 0's in error-free positions and 0 or 1randomly in the error positions.

For $I \subset [n]$, a collection of erroneous codewords is represented by $\mathcal{C}_M(I) = \{\ \mathrm{W} + \mathrm{E} \mid \mathrm{W} \in \mathcal{C}_M, \mathrm{E}\ \subset \mathbb{E},$ and any error vector $\boldsymbol{e}_i \in \mathrm{E}, \forall i = 2,3, \dots, k,\ supp(\boldsymbol{e}_i) \subseteq [n] \backslash I\}$. Each $\mathrm{W}' \in \mathcal{C}_M(I)$is then decoded in two phases, and is denoted Decode(W', I). The message $\boldsymbol{m} \in \mathbb{F}_2^k$ may then be retrieved from $\boldsymbol{w}'$ using RM majority logic decoding after $\boldsymbol{w}' \in \mathcal{C}(I)$is generated from $\mathrm{W}' \in \mathcal{C}_M(I)$as defined in definition 3.2.

The following Theorem 3.1 is given to describe the homomorphic addition and multiplication operations over erroneous codeword matrices.

Theorem: 3.1 (Homomorphic Addition and Multiplication):

Let $\mathcal{C}_M$ be the RMC, for messages $\boldsymbol{m}_i \in \mathbb{F}_2^k$, $\mathrm{W}_i \in \mathcal{C}_M(I)$ is the erroneous codeword .

Code $\mathcal{C}_\mathrm{M}$ is closed under Addition and Multiplications:

$$\sum_{j=1}^{\ell} \mathrm{W}_j \in\ \mathcal{C}_M(I) \text{ and } \prod_{j=1}^{\ell} \mathrm{W}_j \in\ \mathcal{C}_M(I)$$

Code $\mathcal{C}_\mathrm{M}$ supports Homomorphic Addition:

$$Decrypt\left(\sum_{j=1}^{\ell} \mathrm{W}_j\,, I\right) = \sum_{j=1}^{\ell} \boldsymbol{m}_j$$

Code $\mathcal{C}_\mathrm{M}$ supports Homomorphic Multiplication:

$$Decrypt\left(\ \prod_{j=1}^{\ell} \mathrm{W}_j\ \,, I\right) = \prod_{j=1}^{\ell} \boldsymbol{m}_j$$

As a result, the code $\mathcal{C}_M$ – the original RM error correction code with modified representation of codeword structure in matrix form is suitable for construction of FHE schemes.

RMFHE Scheme - Algorithm

The RMFHE scheme is made up of four functions:

- $K \leftarrow Setup(r, m)$: Based on $\mathcal{C}$ and its $\mathcal{C}_M$, the $Setup$ function choose a small integer t smaller than distance d of $\mathcal{C}$ and selects parameter I of size t. It selects the permutation pattern S for $\mathcal{C}_M$. $Setup$ function returns the secret key $K = (I, S)$.
- $\mathrm{C} \leftarrow Encrypt(\boldsymbol{m},\ K)$: $Encrypt$ function accepts two inputs: a plaintext $\boldsymbol{m} \in \mathbb{F}$ and secret key $K = (I, S)$. It initially computes a codeword $\mathrm{W} \in \mathcal{C}_M$ for a given plaintext $\boldsymbol{m}$ using *Encode* function. For an error matrix $\mathrm{E} \in \mathbb{E}$ with k vectors $(\boldsymbol{e}_1, \boldsymbol{e}_2 \dots, \boldsymbol{e}_k)$, erroneous codeword $\mathrm{W}' = \mathrm{W} + \mathrm{E}$ is then computed. The vector $\boldsymbol{e}_1$of E is a null vector, i.e., $\boldsymbol{e}_1 \in 0^n$ while the subsequent vectors are error vectors $\boldsymbol{e}_i \in \mathbb{F}^n$ for each $i = 2, \dots, k$ are randomly chosen vectors such that $supp(\boldsymbol{e}_i) \subseteq [n] \backslash I$. As a result, erroneous codeword $\mathrm{W}' \in \mathcal{C}_M(I)$. To compute the ciphertext $\mathrm{C} \in (\mathbb{F}_2^n)^k$, the permutation σ_S is applied to W' , i.e., $\mathrm{C} = \sigma_S(\mathrm{W}')$.
- $\boldsymbol{m} \leftarrow Decrypt(\mathrm{C},\ K)$: $Decrypt$ function accepts two inputs: a ciphertext C and secret key $K = (I, S)$. Using inverse permutation σ_S', from ciphertext C the erroneous codeword W' is first generated. Then the plaintext $\boldsymbol{m}$ is derived from the erroneous codeword W' using decode function $Decode(\mathrm{W}', I)$.

- $\mathrm{C}'' \leftarrow Add(\mathrm{C}, \mathrm{C}')$: The output ciphertext $\mathrm{C}'' = \mathrm{C}'' = (\mathrm{C} +\ \mathrm{C}')$, i.e., for the two specified input ciphertexts, it is computed as encryption of addition of its underlying plaintexts.
- $\mathrm{C}'' \leftarrow Mult(\mathrm{C}, \mathrm{C}'\ )$: The output ciphertext $\mathrm{C}'' = \mathrm{C}'' = (\mathrm{C}.\mathrm{C}')$, i.e., for the two specified input ciphertexts, it is computed as encryption of multiplication of its underlying plaintexts.

4. SECURITY PROOF

The RMFHE security is based on coding theory assumptions. The number of encryptions is limited to fewer than n due to structural constraints of RM Code. Under these settings, it is possible to reduce its security to the known decoding problem. In the RMFHE scheme, the ciphertext is an erroneous codeword in which the error-free (and error) positions and permutation pattern constitute a secret key.

The use of permutation in particular is quite rare, providing additional security while making evaluation of security harder. To a certain extent, a weaker variation of the RMFHE scheme is investigated in this study. To be more specific, an attacker knows the permutation pattern in the weaker variant of RMFHE. Hence, while keeping the secret key (error-free positions in the codeword)

confidential, the permutation pattern is made available to the attacker.
It gives the following insights:

- The weaker variant of RMFHE may be reduced to the well-known decoding problem, i.e., decoding a noisy codeword without knowledge of the error-free and error locations (secret key).
- The best known technique for decoding a noisy codeword has been demonstrated to be intractable. [11].

Following that, it is said that discovering permutations is tough. That is, keeping permutations hidden provides an extra degree of security.

A. Weaker RMFHE equivalent to DSCP problem

The ciphertext in RMFHE scheme is the erroneous codeword matrix which has errors in the columns indicated by the secret key. The errors that can be added in the columns are based on the master error matrix based on the structure of the generator matrix of the code. The main idea of RMFHE security proof is to identify the good and bad columns of the erroneous codeword matrix (ciphertext) as a solution, under certain conditions of the parameter t. The t is the number of good columns of an $[n, k, d]$ code $\check{\mathcal{C}}_M$. For t, $t \geq (n - \frac{d}{2})$ the solution is polynomially bound by its majority logic decoding algorithm; and for $t < (n - d)$, the solution is not possible as it leads to incorrect decoding. Therefore, the number of good columns t must be taken in between $(n - d)$ and $(n - \frac{d}{2})$, i.e., $(n - d) < t \leq (n - \frac{d}{2})$.

Now, it requires to try a set of t columns of the erroneous matrix to solve for the solution. The same is applied to the remaining $(n - t)$ columns to validate and check the solution. This process takes $\binom{n}{t}$ iterations for all possible combination of t columns out of n columns, in order to correctly identify the good and erroneous columns. Hence, the total effort that is required to compute the good and erroneous columns is approximately equal to $\binom{n}{t} \cdot (n - t)$. From the current state of knowledge, the best known algorithm to solve this is as hard as the PR problem. In PR Problem, for given a set of instances with parameters n, k, t, $t \geq \frac{(n+k)}{2}$, where n is the total number of points given in the instance with at least t number of points belong to the the polynomial of degree k, such that $(n > t > k)$, has a solution to recover t points by checking all possibilities with complexity proportional to $\binom{n}{t}$. An adversary, applying Lagrange interpolation method, trying all possible subsets of size t until finding the resultant polynomial agrees all points, takes combinatorial time, which is intractable.

For every encryption in RMFHE, error-free (and error) columns stay in the fixed and consistent locations. Decoding many codewords with errors at certain locations all the time is therefore equivalent to computing plaintexts from several ciphertexts. It is a special decoding and is also known as decisional synchronised code problem (DSCP). In a nutshell, DSCP means a challenge to determine good and error locations in an erroneous codeword. It implies that once decoded, good and error positions may be clearly recognized. The ciphertext in RMFHE is an erroneous codeword matrix with extra errors embedded intentionally in the columns indicated with the secret key. Recognizing error-free and error columns in this instance is same as distinguishing error-free and error locations in DCSP.

As a result, the RMFHE security can be reduced to DSCP.

Definition. 4.1 Decisional synchronized codeword problem (DSCP):

Let $\check{\mathcal{C}}_M$ is used to represent an $[n, k, d]$ code, the sampler $\mathcal{S}$ is defined for an input $\left(\check{\mathcal{C}}_M, t, \ell\right) : {}^{d}/_{2} \leq t < d$ and $\ell \leq (\frac{d}{2} - 1)$, as follows:

1. Choose $I \subset [n]$ of t size at random.
2. Choose ℓ codeword matrices $\breve{\mathrm{W}}_1, \breve{\mathrm{W}}_2, \breve{\mathrm{W}}_3, \dots, \breve{\mathrm{W}}_\ell \in \check{\mathcal{C}}_M$ such that $\breve{\mathrm{W}}_i \neq \breve{\mathrm{W}}_j$ for every pair $i, j = 1,2,3, \dots, \ell$
3. Select ℓ distinct error matrices $\mathrm{E}_1, \mathrm{E}_2, \mathrm{E}_3, \dots, \mathrm{E}_\ell \in \mathbb{E}$, for each first vector $\boldsymbol{e}_1$ of E_ℓ is a null vector and each vector $\boldsymbol{e}_i \in \mathrm{E}_\ell$ $\forall i = 2,3, \dots, k$, $\boldsymbol{e}_i |\, supp(\boldsymbol{e}_i) \subseteq [n] \backslash I$.
4. for $i = 1, 2, 3, \dots, \ell$, calculate $\mathrm{C}_i = \breve{\mathrm{W}}_i + \mathrm{E}_i \in (\mathbb{F}_2^n)^k$.
5. Return $\mathrm{C} \coloneqq (\mathrm{C}_1, \mathrm{C}_2, \mathrm{C}_3, \dots, \mathrm{C}_\ell) \in ((\mathbb{F}_2^n)^k)^\ell$.

$\mathcal{S}^{Bad}$ and $\mathcal{S}^{Good}$ refer to two changes to the sampler $\mathcal{S}$. $\mathcal{S}^{Bad}$ and $\mathcal{S}^{Good}$ operate similarly to the sampler $\mathcal{S}$; however, each of the $\mathcal{S}^{Good}$ and $\mathcal{S}^{Bad}$ samplers randomly chooses an extra input i from I and $[n] \backslash I$ correspondingly and gives (i, C) as output. (i, C) is referred to as a DSCP instance. It differentiates between two samplers, $\mathcal{S}^{Bad}$ and $\mathcal{S}^{Good}$. A probabilistic polynomial time algorithm (PPT algorithm) A is described as

$$Adv_{\check{\mathcal{C}}_M, t, \ell}^{DSCP,\ A} = \left| \Pr\left[A\left(\mathcal{S}^{Good}(\check{\mathcal{C}}_M, t, \ell)\right) = 1\right] - \Pr\left[A\left(\mathcal{S}^{Bad}(\check{\mathcal{C}}_M, t, \ell)\right) = 1\right] \right|$$

The probability is determined using the random coins from A and the samplers. With regard to a security parameter s , the DSCP$[\check{\mathcal{C}}_M, t, \ell]$ assumption holds if $Adv_{\check{\mathcal{C}}_M,t,\ell}^{DSCP,\ A} = MAX_A\ Adv_{\check{\mathcal{C}}_M,t,\ell}^{DSCP,\ A}$ is negligible.

Only when the sampler size is $\ell \leq (n - t)$, i.e., $\ell \leq d - 1$, an adversary have a negligible advantage in identifying good and bad positions in the original DSCP given by Armknecht et al [2]. In the scheme, an adversary has a negligible advantage in identifying error and error-free columns only when the sampler size is $\ell \leq (\frac{d}{2} - 1)$. It is the foundation for RMFHE reduction proof. If an adversary discovers the error and error-free locations of a DSCP instance (with maximum of ℓ samples), it is likewise possible to detect the error and error-free columns of an erroneous codeword in an RMFHE instance with $\ell \leq (\frac{d}{2} - 1)$ sample size.

As a result, cryptanalysis of RMFHE scheme for determining error-free and error locations (secret key) is equivalent to DSCP hardness. RMFHE is semantically secured as long as the weaker scheme is as hard as DSCP.

In semantic security, even if the attacker has picked two plaintexts from the pool of plaintexts, it is important that no adversary obtain even partial knowledge of these plaintext. In a game, if the adversary catches this, he can take encryptions of plaintexts adaptively selected. The adversary must differentiate between the encryption of two randomly selected plaintexts. The given theory on the pseudo randomness of sampled instances is offered to demonstrate the security of sampled instances:

Theorem. 4.1

To distinguish between the uniform distribution $\mathbb{U}_{k\times n}^{\ell} \in ((\mathbb{F}_2^n)^k)^\ell$ and $\mathbb{D}_{\check{\mathcal{C}}_M,t,\ell}$ induced by sampler $\mathcal{S}$ from definition 4.1, In this case, An adversary is a distinguisher denoted by $\mathcal{A}$ and it holds that

$$Adv_{\check{\mathcal{C}}_M,t,\ell}^{DST,\ A} = \left|\Pr\left[\mathcal{A}(\check{\mathrm{C}}) = 1 \mid \check{\mathrm{C}} \leftarrow \mathbb{D}_{\check{\mathcal{C}}_M,t,\ell}\right] - \Pr\left[\mathcal{A}(\check{\mathrm{C}}) = 1 \mid \check{\mathrm{C}} \leftarrow \mathbb{U}_{k\times n}^{\ell}\right]\right|$$

$\mathcal{A}$ has $\frac{1}{2}$ probability to determine it using the random coins. The DST assumption is true for a security parameter s if $Adv_{\check{\mathcal{C}}_M,t,\ell}^{DST,\ A} = MAX_A\ Adv_{\check{\mathcal{C}}_M,t,\ell}^{DST,\ A}$ is negligible. The distinguisher's advantage is is $Adv_{\check{\mathcal{C}}_M,t,\ell}^{DST,\ A} = \frac{1}{2} + \varepsilon(.)$. In terms of security parameters, the $\varepsilon(.)$ function is negligible.

Proof : The size of the input sample is ℓ, implying that the sample contains ℓ possible matrices. There will be precisely 2^{k-1} random matchings for each given candidate matrix of the input sample. The distinguisher provides no substantial advantage with maximum of ℓ candidates because the size of the key size is in between $d/2$ to $d - 1$,

By random coin problem, the distinguisher's advantage is 0.5 and over and above it he has negligible advantage, it is denoted as $\varepsilon(.)$, where

$$\varepsilon(.) = \frac{\ell.(2^{k-1})}{2^k.2^{k-1}} = \frac{\ell}{2^k}$$

Even when the security parameter is set to a smaller value, this $\varepsilon(.)$ is insignificant.

■

In a controlled manner, random codeword matrices must be produced by an attacker in order to do the reduction. This is conceivable provided that a special encoding of $0 \in \mathbb{F}$ supported by the codeword matrix, which is specified as follows:

Definition 4.2 (Special Encoding of 0):

$\check{\mathcal{C}}_M$ is a special evaluation code that come up with special encoding of 0, if $\exists\breve{\mathrm{W}} = (\breve{\boldsymbol{w}}_1,\ \breve{\boldsymbol{w}}_2, \breve{\boldsymbol{w}}_3, \dots, \breve{\boldsymbol{w}}_k) \in \check{\mathcal{C}}_M$, such that $\breve{\boldsymbol{w}}_1 + \breve{\boldsymbol{w}}_2 + \breve{\boldsymbol{w}}_3 + \cdots + \breve{\boldsymbol{w}}_k = \breve{\boldsymbol{w}}$ and $\breve{\boldsymbol{w}} = (w_1, w_2, \dots, w_n) \in \check{\mathcal{C}} \mid w_i \neq 0$, $\forall i \in [n]$ and $Decode(\breve{\boldsymbol{w}}) = 0$.

The design of transformations on code matrices plays a significant part in the security reduction process, as seen here:

Proposition 4.1. (Transformation τ):

Let $\check{\mathcal{C}}_M^2 \subseteq \mathcal{C}_M$ be a code which permits special encoding of 0, that means $\exists\breve{\mathrm{W}} = (\breve{\boldsymbol{w}}_1,\ \breve{\boldsymbol{w}}_2, \breve{\boldsymbol{w}}_3, \dots, \breve{\boldsymbol{w}}_k) \in \check{\mathcal{C}}_M$, such that $\breve{\boldsymbol{w}}_1 + \breve{\boldsymbol{w}}_2 + \breve{\boldsymbol{w}}_3 + \cdots + \breve{\boldsymbol{w}}_k = \breve{\boldsymbol{w}}$ and $\breve{\boldsymbol{w}} = (w_1, w_2, \dots, w_n) \in \check{\mathcal{C}} : w_i \neq 0$, $\forall i \in [n]$ and $Decode(\breve{\boldsymbol{w}}) = 0$. Then a transformation τ: $(\mathbb{F}^n)^k \times \mathbb{F}^k \rightarrow (\mathbb{F}^n)^k$ (probabilistic mapping) for every message $\boldsymbol{m} \in \mathbb{F}^k$ as

- The output of transformation $\tau(\mathrm{V}, \boldsymbol{m})$ is uniformly random vector, iff $\mathrm{V} \in (\mathbb{F}^n)^k$ is uniformly random
- For a codeword matrix $\mathrm{V} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \dots, \boldsymbol{w}_n) \in \check{\mathcal{C}}_M(I)$, the output of transformation $\mathrm{W} = \tau(\mathrm{V}, \boldsymbol{m})$ is a codeword in $\check{\mathcal{C}}_M(I)$, i.e., V and W consists same error-free and error columns,. It is correct that generating codeword $\boldsymbol{w}$ from W and $Decode(\boldsymbol{w}, I)$ gives the second input $\boldsymbol{m}$ of transformation τ, i.e., W is distributed uniformly over the set of encryption of $\boldsymbol{m}$.
- The outcome of transformation $\mathrm{W} = \tau(\mathrm{V}, \boldsymbol{m})$ is a codeword in $\check{\mathcal{C}}_M(I)$ for a codeword $\mathrm{V} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \dots, \boldsymbol{w}_n) \in \check{\mathcal{C}}_M(I)$, which means that V and W have the same error-free columns. It is correct that extracting codeword $\boldsymbol{w}$ from W and $Decode(\boldsymbol{w}, I)$ yields the second input $\boldsymbol{m}$ of

transformation τ, i.e., W is distributed evenly over the set of encryption of $\boldsymbol{m}$.

Proof. Transformation τ takes $(\mathrm{V}, \boldsymbol{m})$ as input and first selects the random codeword matrix $\mathrm{W}' \in \mathcal{C}_M$ as encoding for $\boldsymbol{m}$. The output W is then computed as $\mathrm{W} = \breve{\mathrm{W}} \cdot \mathrm{V} + \mathrm{W}'$. Because $\breve{\mathrm{W}}$ is assumed to have all non-zero entries, the product $\breve{\mathrm{W}} \cdot \mathrm{V}$ is uniformly random when $\mathrm{V} \in (\mathbb{F}^n)^k$ is any uniformly random vector. As a consequence, W is nothing more than a uniformly random shift of another uniformly random vector.

In another scenario, $\mathrm{V} \in \check{\mathcal{C}}_M(I)$ is a noisy encoding of some unknown message $\widetilde{\boldsymbol{m}}$ under $\check{\mathcal{C}}_M$. $\breve{\mathrm{W}} \cdot \mathrm{V} \in \mathcal{C}_M(I)$ as $\check{\mathcal{C}}_M^2 \subseteq \mathcal{C}_M$, according to Theorem 3.1. In terms of additive property, $\mathrm{W} \in \hat{\mathcal{C}}_M(I)$ follows, and Theorem 3.1 implies that

$$Decode\left(\breve{\mathrm{W}} \cdot \mathrm{V}\right) = Decode\left(\breve{\mathrm{W}}\right) \cdot Decode(\mathrm{V}) = 0 \cdot \widetilde{\boldsymbol{m}} = 0$$

This means that

$$\begin{aligned} Decode(\mathrm{W}) &= Decode\left(\breve{\mathrm{W}} \cdot \mathrm{V} + \mathrm{W}'\right) \\ &= Decode\left(\breve{\mathrm{W}} \cdot \mathrm{V}\right) + Decode(\mathrm{W}') \\ &= 0 + \boldsymbol{m} = \boldsymbol{m} \end{aligned}$$

As a result, W is essentially the constant shift of a uniformly random encoding of $\boldsymbol{m}$, which proves it. ∎

Theorem 4.2. The RMFHE scheme $\mathrm{S}_{RMFHE} = (Setup, Encrypt, Decrypt, Evaluate)$ provides semantic security for the parameters $(\check{\mathcal{C}}_M, t, \ell)$, provided the likelihood of distinguishing between encryptions is $negl(\,)$ and the number of queries is restricted to ℓ including the challenge query.

Proof: Consider $\mathcal{A}^{SS}$ to be a probabilistic polynomial time (PPT) adversary that breaks the semantic security of the RMFHE scheme S_{RMFHE} in ℓ number of queries, including the challenge query. Now, we show how to turn $\mathcal{A}^{SS}$ in $\mathcal{A}^{dst}$ that distinguishes between the distributions between $\mathbb{D}_{\check{\mathcal{C}}_M,t,\ell}$ & $\mathbb{U}_{k\times n}^{\ell}$ as specified in Theorem 4.1. If the advantage to distinguisher $\mathcal{A}$ is negligible, then it follows the same that the advantage to $\mathcal{A}^{dst}$ is also negligible. Consequently, this must be true for $\mathcal{A}^{SS}$ also to prove the semantic security.

If $\mathrm{C} \coloneqq (\mathrm{C}_1, \mathrm{C}_2, \mathrm{C}_3, \dots, \mathrm{C}_\ell) \in ((\mathbb{F}_2^n)^k)^\ell$ is drawn from uniform distribution $\mathbb{U}_{k\times n}^{\ell}$ then $\mathrm{C}_l \in (\mathbb{F}_2^n)^k$ is uniformly random. As a result, the response C_l is also independent of the challenge $(\boldsymbol{m}_0, \boldsymbol{m}_1)$. Hence, $\mathcal{A}^{SS}$ cannot derive any information about b, demonstrating that his advantage is zero here. If $\mathrm{C} \coloneqq (\mathrm{C}_1, \mathrm{C}_2, \mathrm{C}_3, \dots, \mathrm{C}_\ell) \in ((\mathbb{F}_2^n)^k)^\ell$ is drawn from $\mathbb{D}_{\check{\mathcal{C}}_M,t,\ell}$, then $\mathrm{C}_l \in \check{\mathcal{C}}_M(I)$ for secret key I.

The response C_l is a valid encryption of $\boldsymbol{m_l}$. As a result, $\mathcal{A}^{SS}$ would see them as valid encryption, and for a given plaintext, any encryption might be feasible. $\mathcal{A}^{SS}$ has a non-negligible advantage in properly guessing b.

The rest of the proof follows the standard arguments. $\mathcal{A}^{dst}$ runs $\mathcal{A}^{SS}$ frequently enough to estimate $\mathcal{A}^{SS}$'s advantage. If advantage is negligible, $\mathcal{A}^{dst}$ assumes that C was uniformly sampled from $((\mathbb{F}_2^n)^k)^\ell$. Otherwise, it thinks C was drawn from $\mathbb{D}_{\check{\mathcal{C}}_M,t,\ell}$

5. CONCLUSION

The work mainly focused on the construction of implementable FHE using error correction codes. The FHE using Reed Muller codes (RMFHE) is devised and experimented with different parameters [14]. The structure of Reed Muller codes represented in matrix form in order to support both Mod 2 addition and multiplication operations over the ciphertexts. RMFHE is proved semantically secured under decisional synchronized codeword problem (DSCP) through security reduction method.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to all those who have extended support to us for this work.

REFERENCES

[1] A. Bogdanov and C. H. Lee, "Homomorphic Encryption from Codes", Cryptology ePrint. Archive, Report 2011/622. 2.

[2] F. Armknecht, D. Augot, L. Perret, A.R. Sadeghi, "On Constructing Homomorphic Encryption Schemes from Coding Theory", In: Chen L. (eds) Cryptography and Coding. IMACC 2011. Lecture Notes in Computer Science, vol 7089. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-25516-8_

[3] D. Tourky, M. ElKawkagy and A. Keshk, "Homomorphic encryption the "Holy Grail" of cryptography," 2016 2nd IEEE International Conference on Computer and Communications (ICCC), 2016, pp. 196-201, doi: 10.1109/CompComm.2016.7924692.

[4] C. Gentry and S. Halevi, "Implementing Gentry's Fully-Homomorphic Encryption Scheme", In: Advances in Cryptology - Proceedings of EUROCRYPT'11, Lecture Note in Computer Science (LNCS), Vol 6632, Springer-Verlag, 2011, pp. 129-148.

[5] N. P. Smart and F. Vercauteren, "Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes", In: Public Key Cryptography - Proceedings of the 13th International Conference on Practice and Theory in Public Key Cryptography (PKC'10), Lecture Notes in Computer Science (LNCS), Vol 6056, Springer-Verlag,2010, pp. 420-443.

[6] J. Hoffstein, J. Pipher, J.Silverman, "NTRU: A Ring-Based Public Key Cryptosystem", In: Proceedings of the 3rd International Symposium on Algorithmic Number Theory (ANTS-III), ANTS'98, Lecture Notes in Computer Science (LNCS), Vol 1423, Springer-Verlag, 1998, pp. 267-288

[7] M. V. Dijk, C. Gentry, S. Halevi, V. Vaikuntanathan, "Fully homomorphic encryption over the integers", Proceedings of Eurocrypt, Vol. 6110 of LNCS, H. Gilbert (Ed), Springer, 2010, pp. 24-43.

[8] Z. Brakerski, C. Gentry, V. Vaikuntanathan, "Fully Homomorphic Encryption without Bootstrapping", In Proceedings of ITCS'12 the

3rd Innovations in Theoretical Computer Science Conference, pp.309-325, 2011.

[9] A. López-Alt, E. Tromer, V. Vaikuntanathan, “On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption”, In Proceedings of the forty-fourth annual ACM symposium on Theory of computing. ACM, 2012, 1219–1234.

[10] C. Gentry, “A Fully Homomorphic Encryption Scheme”, PhD thesis, Stanford University, 2009.

[11] L. Demarest, B. Fuller, A. Russell, “Hardness of Decoding Random Linear Codes in the Exponent” 2018, Cryptology ePrint Archive, Report 2018/1005, https://ia.cr/2018/1005.

[12] A. Kiayias and M. Yung, “Cryptographic hardness based on the decoding of reed-solomon codes”, IEEE Transactions on Information Theory, 2008, 54(6):2752–2769.

[13] RatnaKumari Challa, VijayaKumari Gunta, Reed-Muller Code Based Symmetric Key Fully Homomorphic Encryption Scheme. ICISS 2016: 499-508.

[14] RatnaKumari Challa, VijayaKumari Gunta, Towards the Construction of Reed-Muller Code Based Symmetric Key FHE. Ingénierie des Systèmes d Inf. 26(6): 585-590 (2021).

Ratnakumari Challa.

She has received her M.Tech degree in Computer Science from University of Hyderabad, India in 2009. She is an Assistant Professor in Department of Computer Science and Engineering, RGUKT-RK Valley, Andhra Pradesh, India. She is currently pursuing Ph.D in JNTUH, Hyderabad, India. Her research interests include Security, Cloud Computing and Image processing

G Vijayakumari. She has received her Ph.D degree from University of Hyderabad, India in 2002 . She is a professor in Department of Computer Science and Engineering, JNTUH, Hyderabad. India. Her research interests include Algorithms, Security and Cloud Computing.