Security and Efficiency Analysis of The Hamming Distance Computation Protocol Based On Oblivious Transfer

Mehmet Sabır Kiraz, Ziya Alper Genç, Süleyman Kardaş

***Abstract*—In Financial Cryptography 2013, Bringer, Chabanne and Patey proposed two biometric authentication schemes between a prover and a verifier where the verifier has biometric data of the users in plain form. The protocols are based on secure computation of Hamming distance in the two-party setting. Their first scheme uses Oblivious Transfer (OT) and provides security in the semi-honest model. The other scheme uses Committed Oblivious Transfer (COT) and is claimed to provide full security in the malicious case.**

In this paper, we show that their protocol against malicious adversaries is not actually secure. We propose a generic attack where the Hamming distance can be minimized without knowledge of the real input of the user. Namely, any attacker can impersonate any legitimate user without prior knowledge. We propose an enhanced version of their protocol where this attack is eliminated. We provide a simulation based proof of the security of our modified protocol. In addition, for efficiency concerns, the modified version also utilizes Verifiable Oblivious Transfer (VOT) instead of COT. The use of VOT does not reduce the security of the protocol but improves the efficiency significantly.

***Index Terms*—Biometric Identification, Authentication, Hamming distance, Privacy, Committed Oblivious Transfer.**

1. Introduction

Recently, several commercial organizations have invested in secure electronic authentication systems to reliably verify identity of individuals. Biometric authentication mechanisms is one of the wide-spread popular technology because of the cost-effective improvements in sensor technologies and in the efficiency of matching algorithms [21]. The biometric data (i.e. templates) of a user is inherently unique. This uniqueness provides the reliability of the individual to be securely authenticated for accessing to an environment when the biometric data is kept as secret. The biometric data cannot be directly used with conventional encryption techniques because these data are inherently noisy [27]. Namely, whenever two sample of data extracted from the same fingerprint, these data would not be exactly same. In this context, in order to eliminate noisy nature of the biometric templates, several error correction techniques have been proposed in the literature [18], [25], [24].

Biometric authentication over insecure network raises more security and privacy issues. The primary security issue is the protection of the plain biometric templates against malicious adversary because they cannot be replaced with a new one, once they are compromised. The common biometric authentication system is as follows. For each user, the biometric template is stored in a database during the *enrollment* phase. In the *verification* phase, a new fresh acquisition of a user is compared to the template of the same individual stored in the database. The verification phase can either be processed within the smart card (i.e, on-card matching), or in the system outside the card (i.e, off-card matching) [35]. Since the biometric template is not necessarily transferred to the outside environment, the on-card matching technique protects the template. In both techniques, the authentication protocol should not expose the biometric template without the user's agreement. In order to ensure privacy of the user, the biometric template should stored in an encrypted form in the database and no one including the server side can learn any information on link between the user and her biometric data. But still, it should be possible to verify whether a user is authentic [6].

In order to thwart, the security and privacy issue described above for the biometric authentication, several matching algorithms are proposed in the literature. Many of them utilize the computation of the Hamming distance of two binary biometric templates. Note that the Hamming distance does not reveal any significant information to any polynomially bounded adversary. In this context, in Financial Cryptography 2013, Bringer *et al.* [9] have proposed two secure Hamming distance computation schemes based on Oblivious Transfer. In their proposals, the authors integrate the advantages of both biometrics and cryptography in order to improve the overall security and privacy of an authentication system. The first scheme based solely on 1-out-of-2 Oblivious Transfer (OT) and it achieves full security in the semi-honest setting and one-sided security in the malicious setting. The second scheme uses Committed Oblivious Transfer (COT) and is claimed to provide full security against malicious adversaries.

A. Contributions

In a recent paper Bringer *et al.* [9] propose two schemes, called SHADE, for secure computation of Hamming distance using Oblivious Transfer (OT). The protocol uses OT in the semi-honest setting and Committed Oblivious Transfer (COT) of [28] in the presence of malicious adversary. The main contributions of this paper are summarized as follows:

- In this paper, we first revisit the Hamming distance computation protocol of Bringer *et al.* [9]. We show

Mehmet Sabır Kiraz (corresponding author), Ziya Alper Genç and Süleyman Kardaş and is with the Department of Cryptology, TUBITAK BILGEM UEKAE, Kocaeli, Turkey (e-mail: {mehmet.kiraz, ziya.genc, suleyman.kardas}@tubitak.gov.tr

Ziya Alper Genç is also with the İstanbul Şehir University and Süleyman Kardaş is also with Sabancı University, Istanbul, Turkey.

that their protocol is insecure in the malicious model. Namely, the full scheme in [9] which is claimed to be secure, has a severe weakness on the computation of Hamming distance. We show that this weakness allows any malicious adversary to violate the completeness of the protocol and impersonate any legitimate user.
The protocol flaw resides in the method used for validation of the inputs of a user. The protocol forces the user using zero-knowledge protocols to submit valid inputs, i.e. pairs of integers (x, y) that differ by 1. The method succeeds at checking the difference, however, it fails at validation of the pairs, i.e. a malicious party can submit invalid pairs $(x - 2^{-1}, y + 2^{-1})$. Since the Hamming distance is computed by summing each side and computing the difference, any adversary without any prior knowledge can authenticate himself. Note that we believe this attack technique to be of independent interest and can be applied to other schemes.
- In order to eliminate this severe weakness, we propose a new method for input validation. This way, we removed the fault in the protocol and enhanced the security of it. We also show that the computational complexity of the fixed protocol is comparable with the insecure protocol. Moreover, we optimize the new input validation method for biometric authentication systems. We prove the security of our protocol using ideal/real simulation paradigm in the standard model [10], [16], [31] and [1].
- Lastly, we consider the efficiency of the protocol and show that running a COT is not necessary in the second option of the protocol. We show that VOT is sufficient instead of using complete COT protocol which contains additional commitments and zero-knowledge proofs [12]. This leads a considerable improvement in the computational complexity of the protocol.

B. *Organization*

Section 2 gives the related work on the computation of Hamming distance and biometric authentication systems. Section 3 provides the security and privacy model for crypt-analyzing biometric authentication protocol. Section 4 reviews the two schemes in the protocol, basic scheme which uses OTs and full scheme based on COT of bit-strings. In Section 5, we present an attack to their full scheme and show that their protocol is insecure. In Section 6, we propose a security fix and discuss the efficiency of their protocol in the malicious model. Here, we show that VOT is sufficient instead of COT. In Section 7 we prove our fixed protocol using simulation-based paradigm. This will significantly improve the efficiency of the protocol will be shown in Section 8. Section 9 concludes the paper.

2. Related Work

There has been a large amount of research done on the security and efficiency of biometric authentication systems. In this section, we review the most recent works for biometric authentication.

Hamming distance together with Oblivious Transfers is one of the most elegant tools used in biometric authentication systems. For example, Jarrous and Pinkas propose *bin*HDOT protocol [23] to compute Hamming distance based on 1-out-of-2 Committed Oblivious Transfer with Constant Difference (COTCD) of Jarecki and Shmatikov [22] and Oblivious Polynomial Evaluation (OPE) of Hazay and Lindell [19]. The protocol also uses commitments and zero-knowledge proofs to guarantee that each party follows the protocol. The protocol provides full security in the malicious model. One OPE protocol and n COTCDs are invoked to compute the Hamming distance between two strings of n bits.

The SCiFI (Secure Computation of Face Recognition) of Osadchy *et al.* is the first secure face identification system which is well suited for real-life applications [33]. The SCiFI system consist of two parts: a client and a server. The server prepares face recognition database that contains representations of face images. This computation is done offline. In the verification phase, the client prepares her face representation and then a cryptographic protocol which uses uses Paillier encryption and Oblivious Transfer is run between the server and the client. The authors implemented a complete SCiFI system in which a face is represented with a string of 900 bits. The authors designed the system by aiming the minimal online overhead: the most significant requirement for computing Hamming distance between this length of bit strings is 8 invocations of 1-out-of-2 OTs.

Bringer *et al.* [8] used biometric authentication/identification for access control. Note that it is important to securely store the biometric template to the server. Using conventional encryption schemes for securing the biometric template can provide a strong protection. Note that conventional cryptography requires exact match while biometrics always have a threshold value, therefore biometric authentication over the encrypted domain is a challenging task. In this paper, a cryptographic scheme is given for biometric identification over an encrypted domain which uses Bloom Filters with Storage and Locality-Sensitive Hashing. This paper is interesting since it proposes the first biometric authentication/identification scheme over encrypted binary templates which is stored in the server's database.

In another paper, Bringer *et al.* [7] proposed a security model for biometric-based authentication protocols, relying the Goldwasser-Micali cryptosystem [17]. This system allows the biometric match to be performed in the encrypted domain in such a way that the server cannot identify which user is authenticating. The proposed system requires storage of biometric templates in plain form. In order to protect the privacy, the system ensures that the biometric feature stored in the database cannot be explicitly linked to any identity, but the DB only verifies whether the received data belongs to an identity in the database.

Erkin *et al.* [15] propose a privacy preserving face recognition system on encrypted messages which is based on the standard Eigenface recognition system [36]. In their protocol design, they utilized semantically secure Paillier homomorphic public-key encryption schemes and Damgård, Geisler and Krøigaard (DGK) cryptosystem [13], [14]. Later, Sadeghi *et al.* give an improvement on the efficiency of this system [34]. In this study, they merge the eigen-face recognition algorithm using homomorphic encryption and Yao's garbled circuits. Their protocol improves the scheme proposed by Erkin *et al.* significantly, i.e. it has only a constant number of $O(1)$ rounds and most of the computation and communication performed during the pre-computation phase.

Tuyls *et al.* [37] propose a template protection scheme for fingerprint based authentication in order to protect biometric data. During the enrollment phase, Alice's biometric features X is extracted, the Helper Data [38] W, that is required by error-correction mechanism is computed, a one-way hash function $\mathcal{H}$ is applied to S and the data (Alice, W, $\mathcal{H}(S)$) is stored to the server. Here, S is a randomly chosen secret value such that $G(X, W)=S$ for a shielding function G [32]. During the verification phase, after Alice's noisy biometric data $\overline{X}$ is extracted, the server sends W back to the sensor. The sensor computes $\overline{S} = G(\overline{X},W)$ and $\mathcal{H}(\overline{S})$. Then, the server compares $\mathcal{H}(S)$ with $\mathcal{H}(\overline{S})$, and grants access if the results are equal. The Helper Data is sent over the public channel, i.e. an adversary may obtain W. The authors however design the system in such a way that the adversary obtains minimal information about X by capturing W.

Kulkarni *et al.* [30] propose a biometric authentication scheme based on Iris Matching. Their scheme uses the *somewhat* homomorphic encryption scheme of Boneh *et al.* [5] which allows an arbitrary number of addition of ciphertexts but supports only one multiplication operation between the ciphertexts. The scheme is based on Paillier encryption and bilinear pairing. This scheme consists of two phases: Enrollment phase and Verification phase. During the Enrollment phase, first the necessary keys are generated by the server and sent to the client securely. Secondly, the client's biometric data is XORed with the key, and a mask value is XORed with a mask key. Both XORed values are sent to the server. During the Verification (authentication) phase, the client sends an encryption of an authenticated biometric data to compute the distance. The protocol is proven to be secure in the semi-honest model.

Kerschbaum *et al.* [26] propose an authentication scheme in a different setting. Namely, assume that there are two parties where each have a fingerprint template. They would like to learn whether the templates match, i.e. generated from the same fingerprint. However, they do not want to reveal the templates if there is no match. Their protocol uses secure multi-party computation which is secure only in the semi-honest model.

Barni *et al.* propose a privacy preserving authentication scheme for finger-code templates by using homomorphic encryption which is secure only in the semi-honest model [3], [4]. Their protocol allows the use of the Euclidean distances to compare fingerprints in such a way that the biometric data is reduced for computing a smaller encrypted value that is sent to the server.

3. Security and Privacy Model

We adopt the standard simulation-based definition of ideal/real security paradigm ideal/real simulation paradigm in the standard model which is already highlighted in [10], [16], [31] and [1] and prove that our protocol achieves this notion and secure against a static and active adversary. In this simulation-based security, the view of a protocol execution in a real setting is compared (a statistical/computational indistinguishable manner) as if the computation is executed in an ideal setting where the parties send inputs to a trusted third party $\mathcal{F}$ that performs the computation and returns its result.

In an ideal setting, the parties send their inputs x and y to a trusted third party $\mathcal{F}$ who computes $f(x, y)$ (which is the output of the Hamming distance in our setting) and sends $f_1(x, y)$ to the first party and $f_2(x, y)$ to the second party ($f_1(x, y)$ and $f_2(x, y)$ can be $\perp$ if only one party is required to learn the output). Note that the adversary who controls one of the parties can choose to send any input it wishes to the trusted third party $\mathcal{F}$, while the honest party always sends its specified input. In a real execution of a protocol, one of the parties is assumed to be corrupted under the complete control of an adversary $\mathcal{A}$. Note that we always assume that the adversary $\mathcal{A}$ corrupts one of the two parties at the beginning of the protocol execution and is fixed throughout the computation (is known static adversary model).

Informally, a protocol is secure if for every real-model adversary $\mathcal{A}$ interacting with an honest party running the protocol, there exists an ideal-model adversary $\mathcal{S}$ interacting with the trusted party computing f, such that the output of the adversary and the honest party in the real model is computationally indistinguishable from the output of simulator and the honest party in the ideal model. More formally,

Definition 3.1. *(Simulation-based security) Let f and the protocol Π as above. We say that the protocol Π_f securely computes the ideal functionality $\mathcal{F}$ if for any probabilistic polynomial-time real-world adversary $\mathcal{A}$, there exists a probabilistic polynomial-time an ideal-model adversary $\mathcal{S}$ (called the simulator) such that*

$$\mathsf{REAL}_{\Pi_{\mathcal{F}},\mathcal{A}}(x, y)_{x,y \text{ s.t. } |x|=|y|} \approx \mathsf{IDEAL}_{\mathcal{F},\mathcal{S}}(x, y)_{x,y \text{ s.t. } |x|=|y|}$$

Note that the above definition implies that the parties already know the input lengths (by the requirement that $|x|$ = $|y|$).

Note that VOT and COT protocols are used as sub-protocols. In [11], [2], it is shown that it is sufficient to analyze the security of a protocol in a hybrid model in which the parties interact with each other and assumed to have access to a trusted third party that computes a VOT (resp. COT) protocol for them. Thus, in the security analysis of our protocol the simulator will play the role of the trusted third party for VOT (resp. COT) functionality when simulating the corrupted party. Roughly speaking, in the hybrid model, parties run an arbitrary protocol like in the real model, but have access to a trusted third party that computes a functionality (in our case VOT or COT) like in the ideal model. A protocol is secure if any attack on the real model can be carried out in the hybrid model.

4. The Basic and the Full Scheme of Bringer *et al.*

In this section, we briefly describe the basic and the full scheme in [9] used for computation of Hamming distance between two bit strings. The basic scheme uses oblivious transfer (OT) and provides full security when the parties are semi-honest and one-sided security in the malicious model. The full scheme uses committed oblivious transfer (COT) [28] and zero-knowledge proofs of knowledge [12] to compute the Hamming distance in malicious model. Each scheme has two options to select the party which computes and outputs the result meaning that each party may act as a server and the other as a client.

A. The Basic Scheme

The basic scheme is designed to provide secure and efficient method for computing the Hamming distance between two bit strings in semi-honest model. The intuition behind this protocol is that if both parties are semi-honest, the OT protocols is sufficient to preserve privacy.

The basic scheme in [9] is roughly as follows:

Let P_1 and P_2 have the inputs $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$ respectively. At the first step, P_1 randomly picks $r_1, \ldots, r_n \in_R \mathbb{Z}_{n+1}$ and computes $R = \sum_{i=1}^{n} r_i$. For each $1 \leq i \leq n$, the parties run oblivious transfer in which P_1 acts as the sender and P_2 acts as the receiver. Namely, P_1 inputs $(r_i + x_i, r_i + \overline{x_i})$ where $\overline{x_i} = 1 - x_i$ and P_2 inputs y_i. At the end of OT, P_2 receives $t_i = (r_i + x_i)$ if $y_i = 0$ and $(t_i = r_i + \overline{x_i})$ otherwise. Next, P_2 computes $T = \sum_{i=1}^{n} t_i$. In the last step,

- 1^{st} **Option:** P_2 sends T to P_1.
  P_1 computes and outputs $T - R$.
- 2^{nd} **Option:** P_1 sends R to P_2.
  P_2 computes and outputs $T - R$.

In the case of one party is malicious, the privacy of the honest party is still provided because of the flexibility at the end of the protocol.

Compared to the related protocols for secure computation of Hamming distance in the semi honest model, the basic scheme of Bringer *et al.* [9] is the most efficient protocol as they proved in Section 6 of [9].

The authors also mention that the basic scheme can be optimized by using the state of the art techniques, i.e. extended oblivious transfer, as first proposed by Ishai *et al.* in [20] and later improved in [29]. This technique leads to an efficient construction which extends k OTs to n OTs ($k < n$) in the random oracle model that is secure against only semi-honest adversaries (note that hash functions can be replaced with RO model in the real case).

B. The Full Scheme

The full scheme of Bringer *et al.* considers the case where the parties are assumed to be malicious. Note that running OT protocol does not prevent a party from modifying her input. Secondly, the receiver may send a different value than the actual OT output that she computes. In order to prevent such scenarios, the authors propose to use the 1-out-of-2 Committed Oblivious Transfer (COT) protocol of Kiraz *et al.* presented in [28]. Though, in Section 5, we show that the idea of input validation for P_1 is not sufficient and can be exploited with success.

Before we proceed, let's continue with the description of the full scheme.

At the first step of the protocol, P_2 commits to her inputs y_i's and proves that each y_i is either 0 or 1. At the same time, P_1 generates random r_i's from the plaintext space of the commitments scheme and computes $R = \sum_{i=1}^{n} r_i$ but this time she commits to $(a_i, b_i) = (r_i + x_i, r_i + \overline{x_i})$. P_1 publishes $A_i = \mathsf{Commit}(a_i, \alpha_i)$ and $B_i = \mathsf{Commit}(b_i, \beta_i)$ where α_i, β_i are random values from the plaintext space of the commitments scheme. Furthermore, she proves that her inputs a_i's and b_i's differ by 1. Next, the COT protocol is run for each i. At the end of each COT, P_2 receives $t_i = r_i + (x_i \oplus y_i)$ and both parties receive $C_i = \mathsf{Commit}(t_i, \tau_i)$ where τ_i is a random value from the plaintext space of the commitments scheme. When all the COTs are run, P_2 computes the sum $T = \sum_{i=1}^{n} t_i$.

At this point, there are again two options:

- 1^{st} **Option:** P_2 computes $C = \mathsf{Commit}(T, \tau_i) = C_1 \odot \ldots \odot C_n$. P_2 sends T to P_1 and proves that C commits to T. P_1 computes $C = C_1 \odot \ldots \odot C_n$ and checks the proof. If all verifications are successful, P_1 outputs $T - R$.
- 2^{nd} **Option:** P_1 computes $K = \mathsf{Commit}(2R + n, \rho) = A_1 \odot \ldots A_n \odot B_1 \odot \ldots \odot B_n$. P_1 sends R to P_2 and proves that K commits to $2R + n$. P_2 computes $K = A_1 \odot \ldots A_n \odot B_1 \odot \ldots \odot B_n$ and checks that $K = \mathsf{Commit}(2R + n, \rho)$. If all verifications are successful, P_2 outputs $T - R$.

The authors in [9] claims that the above scheme is fully secure against malicious parties. However, in the next section we show that a malicious P_1 can easily authenticate himself to P_2.

5. Security and Efficiency Analysis of the Protocol Bringer *et al.*

We are now ready to describe the protocol flaw of the full scheme in detail. The security flaw is due to the proof for validation of P_1's input bits. The flaw allows a malicious P_1 to minimize the Hamming distance between her and P_2's inputs. Namely, an adversary can successfully authenticate to a server without the knowledge of a valid user's biometric data. In the next section, we will propose a fix for this flaw by redesigning a new proof for validation. We show that the complexity of the new proof for the validation of P_1's input bits for biometric authentication systems is significantly reduced.

Furthermore, we will also analyze the protocol from the efficiency perspective and show that the complexity of the protocol can be significantly improved. COT protocol is basically designed as a sub-protocol in order to prevent possible malicious behaviors between sender and receiver, where the committed output of COT is expected to be used in further parts of the system. However, the committed outputs of COT are not used in the case that P_1 computes the Hamming distance. Hence, we will point out that verifiable OT will be sufficient in the case P_1 computes the Hamming distance. This will avoid to compute n commitments together with the zero-knowledge proofs (for each run of COT protocol). Namely, we will improve the efficiency of the protocol by using VOT instead of COT when P_1 is the server.

A. Attack to the Full Scheme

The protocol is insecure in the case where P_1 is malicious. This is because P_1 is free in the sense that she can commit to any pair such that the absolute value of the difference of the encryption values is 1, i.e. P_1 proves that $|b_i - a_i| = 1$ where the pair (a_i, b_i) is supposed to be $(r_i + x_i, r_i + \overline{x_i})$. However, a malicious P_1 may choose encrypted values in a special way together with the proof that every pair has difference 1. Our attack uses the fact that at the end of each COT, P_2 receives either $t_i = r_i + g_i$ or $t_i = r_i + h_i$ and computes the sum $T = \sum_{i=1}^{n} t_i$ (note that g_i is expected to be equal to x_i and h_i to $\overline{x_i}$). However, with a careful choosing of g_i's and h_i's, some g_i's can be neutralized by some h_i's in this sum. Hence, by this approach $T = \sum_{i=1}^{n} t_i$ is closer to $R = \sum_{i=1}^{n} r_i$.

Without loss of generality assume that #0's in P_2's input Y is ℓ (i.e. #1's in Y is $n - \ell$). The adversary must know the information about the bias. Note that this is the most general case and in the next section we propose a practical attack for biometric authentication schemes where the inputs are uniformly distributed over an n-bit string. To be more concrete, the attack is given as follows:

- P_2 commits to her inputs y_i's and proves that each y_i is either 0 or 1. P_1 then generates random r_i's and computes $R = \sum_{i=1}^{n} r_i$.
- Next, instead of following the protocol, P_1 can compute $(a_i, b_i) = (r_i - (1 - \ell/n), r_i + \ell/n)$ and can publish $A_i = \mathsf{Commit}(a_i, \alpha_i)$ and $B_i = \mathsf{Commit}(b_i, \beta_i)$ where α_i, β_i are random values from the plaintext space of the commitments scheme. Note that for each i, $|b_i - a_i| = 1$ and hence, the proofs will pass successfully.
- At the end of each COT, P_2 receives either $t_i = r_i - (1 - \ell/n)$ or $t_i = r_i + \ell/n$. After COTs are run, P_2 computes the sum

$$\begin{aligned} T &= \sum_{i=1}^{n} t_i \\ &= \sum_{i|y_i=0} (r_i - (1 - \ell/n)) + \sum_{i|y_i=1} (r_i + \ell/n) \\ &= -\ell(1 - \ell/n) + (n - \ell)\ell/n + \sum_{i=1}^{n} r_i \\ &= \sum_{i=1}^{n} r_i \\ &= R. \end{aligned}$$

Therefore, the Hamming distance $d_H(X, Y) = T - R$ will be equal to 0. Hence, without knowledge of the real X, P_1 fools P_2 into outputting an incorrect hamming distance value without being detected.

The authors in [9] uses the COT protocol of Kiraz *et al.* [28]. Note that this COT scheme uses threshold ElGamal encryption as a commitment mechanism, i.e. $\mathsf{Commit}(x_i, \alpha_i) = \mathsf{Enc}(x_i, \alpha_i)$ where $x_i, \alpha_i \in G$ where G is a large finite cyclic group (of a prime order). This guarantees the existence of the inverse of n. Hence, P_1 may use $(a_i, b_i) = (r_i - (1 - \ell n^{-1}), r_i + \ell n^{-1})$ as input.

B. Applying the Attack Scenario to Biometric Authentication Systems

The attack method described in previous section can be directly applied to biometric authentication systems with a higher success rate. For biometric authentication systems the input bit-strings of P_2 (which is generated from a biometric template) is expected to be independent and identically distributed. That is, there are nearly equal number of zeros and ones in an input string. Below, we show that this fact easily allows an adversary to minimize the Hamming distance and authenticate himself to a biometric authentication system:

1) P_2 commits to her inputs y_i's and proves that each y_i is either 0 or 1.
2) P_1 picks random r_i's and computes $R = \sum_{i=1}^{n} r_i$.
3) Instead of computing $(a_i, b_i) = (r_i + x_i, r_i + \bar{x}_i)$, P_1 computes $(a_i, b_i) = (r_i - 2^{-1}, r_i + 2^{-1})$ in order to make the commitments $A_i = \mathsf{Commit}_{P_1,i}(a_i, \alpha_i)$ and $B_i = \mathsf{Commit}_{P_1,i}(b_i, \beta_i)$. The authors in [9] uses homomorphic encryption as the commitment mechanism. Since those cryptosystems work in a group of prime order, the multiplicative inverse of 2 always exists, i.e. P_1 can commit to $(a_i, b_i) = (r_i - 2^{-1}, r_i + 2^{-1})$. Next P_1 proves that $|b_i - a_i| = 1$ which always holds. Note that P_1 does not prove the validity of her input, i.e, she does not prove that the x_i's are equal to either 0 or 1.
4) COTs are run and, in one half of the COTs (because of the uniform distributed inputs), P_2 receives $t_i = r_i - 2^{-1}$ and $t_i = r_i + 2^{-1}$ in the other half.
5) P_2 computes $T \leftarrow \sum_{i=1}^{n} t_i$. Since y_i's are equally distributed, i.e. the numbers of 0s and 1s in $\{y_1, \ldots, y_n\}$ are equal, P_2 computes $T = \left(\sum_i r_i + 2^{-1}\right) + \left(\sum_i r_i - 2^{-1}\right) = \sum_{i=1}^{n} r_i = R$.
6) Using the 2^{nd} option, $K = \mathsf{Commit}_{P_2,i}(2R + n, \rho) = A_1 \odot \ldots \odot A_n \odot B_1 \odot \ldots \odot B_n$.
7) P_1 sends R and the proof that K commits to $2R + n$ to P_2.
8) P_2 computes $d_H(X, Y) = T - R = 0$ and authenticates P_1.

C. Our solution for the attack

The weakness of the full scheme is due to the wrong method used for validation of the input pairs $\{(a_i, b_i), \forall i = 1, \ldots, n\}$. A malicious P_1 can exploit this weakness as described in the previous section. Therefore, designing cryptographic protocols should be carefully checked against these kinds of tricks.

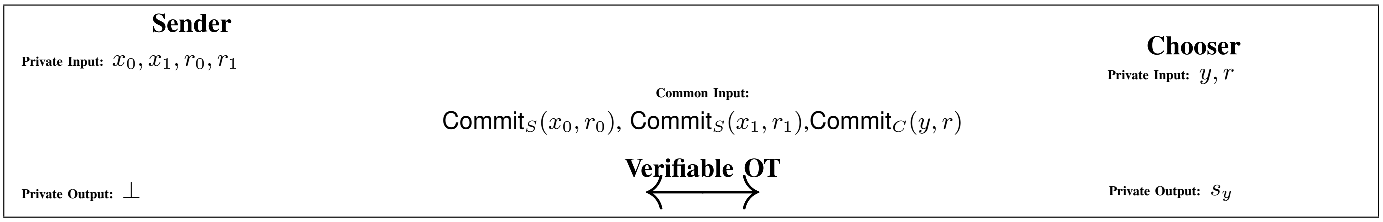

Fig. 1: Verifiable Oblivious Transfer

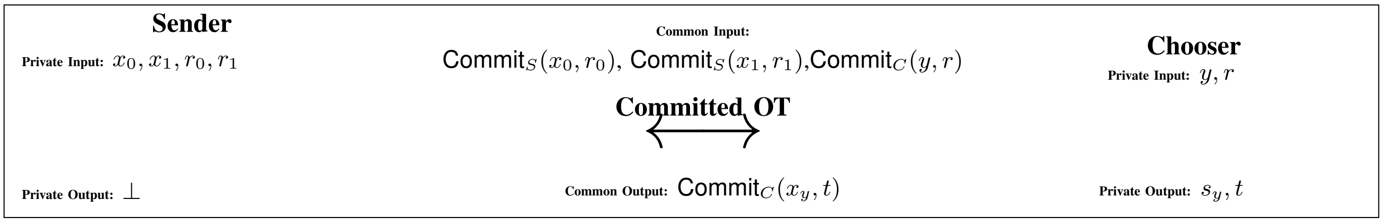

Fig. 2: Committed Oblivious Transfer

As a security fix, we modify the step in which P_1 generates random r_i values. Namely, after generating each r_i, P_1 will compute and publish $A_i = \mathsf{Commit}(r_i + x_i), B_i = \mathsf{Commit}(r_i + \overline{x})$ and $\mathsf{Commit}(r_i)$. Moreover, P_1 will send the proof of:

$$((a_i - r_i) = 0 \vee (b_i - r_i) = 0) \wedge |b_i - a_i| = 1$$

that is equivalent to

$$(a_i + b_i - 2r_i = 1) \wedge |b_i - a_i| = 1$$

This statement contains one more relation than the original proof in [9]. Although the computation cost of the protocol is slightly increased, the validation process is now secure. To see this, assume that $(a_i + b_i - 2r_i = 1) \wedge |b_i - a_i| = 1$ is true. This implies $|b_i - a_i| = 1$ and $(a_i + b_i - 2r_i = 1)$. Here, there are two cases:

$$a_i = b_i + 1 \Rightarrow 2b_i + 1 - 2r_i = 1 \Rightarrow b_i = r_i, a_i = r_i + 1$$
$$b_i = a_i + 1 \Rightarrow 2a_i + 1 - 2r_i = 1 \Rightarrow a_i = r_i, b_i = r_i + 1$$

In Section 7 we provide the security analysis of the improved scheme.

1) More Efficient Solution for Biometric Authentication: Biometric authentication systems are designed to tolerate a small level of errors. In general, the measure process is not perfect in most environments and thus, instead of exact match, a biometric systems authenticates a party that matches with a small error to prevent false negatives.

The authentication process must also have a small complexity to compute the result in the fastest way. Therefore each party must prove nothing more then the necessary and sufficient data for validation of her input.

These motivations lead us to design a more efficient proof that can be used in the biometric authentication systems. Namely, after generating and publishing the commitments to a_i, b_i, r_i as in the previous section, P_1 will send the proof of:

$$a_i + b_i - 2r_i = 1$$

The above relation has a smaller complexity than $|b_i - a_i| = 1$ while it still provides higher security. This input validation method detects our attack, i.e. an adversary may input $(a_i, b_i) = (r_i - 2^{-1}, r_i + 2^{-1})$ and pass the validation but its Hamming distance will be $\frac{n}{2}$ which is the expected Hamming distance of a random input with length n.

D. Efficiency Enhancements

In this section, we present some improvements for the efficiency of the protocol. First, we reduce the computational complexity of the protocol using VOT instead of COT without sacrificing the security. Namely, COT will not be necessary in the case where P_2 computes the final Hamming distance. Next we will reduce the complexity of the proof for the validity of P_1's inputs in the case of biometric authentication.

1) COT versus VOT: Verifiable OT and Committed OT are natural combination of $\binom{2}{1}$-OT and commitments. Let Commit_S and Commit_C be commitments by Sender and Chooser respectively. The functionality of committed OT is illustrated in Figure 2.

Verifiable OT is defined if the $\mathsf{Commit}_C(x_y, t)$ is not required as output. We show that the basic protocol in [9] does not have to use COT in the case that the server computes the result.

We note two aspects:

$$\text{What to transfer} \begin{cases} \text{bits} & x_0, x_1 \in \{0,1\} \\ \text{strings} & x_0, x_1 \in \{0,1\}^k \end{cases}$$

$$\text{Committed Output} \begin{cases} \text{yes} \rightarrow \textbf{Committed OT} \\ \text{no} \ \rightarrow \textbf{Verifiable OT} \end{cases}$$

2) Efficiency Improvement Using VOT: In this section, we point out a computational complexity reduction. Note that COT is run for the malicious case in [9]. COT requires the receiver to obtain the output together with its commitment to this value. In the beginning of the protocol, the input of P_1 is an n-bit string $X = (x_1, \dots, x_n)$ and the input of P_2 is an n-bit string $Y = (y_1, \dots, y_n)$. After running the protocol there are two options:

- P_1 obtains the Hamming distance $d_H(X, Y)$ and P_2 obtains nothing
- P_2 obtains the Hamming distance $d_H(X, Y)$ and P_1 obtains nothing

In case P_2 computes the Hamming distance, the committed values from the output of COT will not be used. In such case, these commitments are not necessary to be computed and, therefore VOT will be sufficient to use. We realized this observation after writing the COT protocol explicitly with the overall protocol instead of using as a black box. If P_1 computes the Hamming distance COT will still be necessary to use.

6. Our Fixed and Improved Scheme

We have made modifications to the full scheme in [9]: we fix the security weakness described in Section 5 and improve the efficiency of the protocol as mentioned in Section 5. Now, we give the corrected scheme with all details:

Inputs:

- P_1 inputs an n-bit string $X = (x_1, \ldots, x_n)$
- P_2 inputs an n-bit string $Y = (y_1, \ldots, y_n)$

Outputs:

- 1^{st} Option: P_1 obtains $d_H(X,Y)$ and P_2 obtains nothing
- 2^{nd} Option: P_2 obtains $d_H(X,Y)$ and P_1 obtains nothing

Protocol:

1) P_2 commits to her inputs y_i's and proves that each of y_i is either 0 or 1.
2) P_1 generates random r_i's from the plaintext space of Commit and computes $R = \sum_{i=1}^{n} r_i$.
3) P_1 commits to $(a_i, b_i, r_i) = (r_i + x_i, r_i + \overline{x_i}, r_i)$. P_1 publishes $A_i = \mathsf{Commit}(a_i, \alpha_i), B_i = \mathsf{Commit}(b_i, \beta_i)$ and $\mathsf{Commit}(r_i, \gamma_i)$.
4) P_1 proves that $(|a_i - r_i| = 0 \vee |b_i - r_i| = 0) \wedge |b_i - a_i| = 1$.
5) For each $i = 1, \ldots, n$, a COT is run where
   - P_1 acts as the sender and P_2 as the receiver.
   - P_2's selection bit is y_i.
   - P_1's input bit is (a_i, b_i).
   - The output obtained by P_2 is $t_i = r_i + (x_i \oplus y_i)$ and τ_i.
   - Both parties obtain $C_i = \mathsf{Commit}(t_i, \tau_i)$.
6) P_2 computes $T = \sum_{i=1}^{n} t_i$
7) 1^{nd} **Option: Run VOT**
   a) P_1 computes $K = \mathsf{Commit}(2R + n, \rho) = A_1 \odot \ldots A_n \odot B_1 \odot \ldots \odot B_n$.
   b) P_1 sends R to P_2 and proves that K commits to $2R + n$.
   c) P_2 computes $K = A_1 \odot \ldots A_n \odot B_1 \odot \ldots \odot B_n$ and checks that $K = \mathsf{Commit}(2R + n, \rho)$.
   d) If all verifications are successful, P_2 outputs $T - R$.

   2^{st} **Option: Run COT**

   a) P_2 computes $C = \mathsf{Commit}(T, \tau_i) = C_1 \odot \ldots \odot C_n$.
   b) P_2 sends T to P_1 and proves that C commits to T.
   c) P_1 computes $C = C_1 \odot \ldots \odot C_n$ and checks the proof.
   d) If all verifications are successful, P_1 outputs $T - R$.

7. Security Analysis of Our Scheme

A cryptographic protocol is secure if the view of an adversary in a real protocol execution can be generated from the information the adversary has (i.e., its input and output). In this section, we show that the security of the protocol by constructing a simulator that is given only the input and output of the "corrupted" party, and generates a view that is indistinguishable from the view of the adversary in a real protocol execution [10], [16], [31], [1]. This implies that the adversary learns no information from the real protocol because it could generate anything from what it sees in such an execution by itself.

Theorem 7.1. *The proposed protocol, which is shown in Figure 3, is secure in the presence of static malicious adversaries.*

Proof. We show that given a party is corrupted, there exists a simulator that can produce a view to the adversary that is statistically indistinguishable from the view in the real protocol execution based on its private decryption share as well as public information.

Case-1-P_1 is corrupted. Let $\mathcal{A}_{P_1}$ be an adversary corrupting P_1. We construct a simulator $\mathcal{S}_{P_1}$ and show that the view of the adversary $\mathcal{A}_{P_1}$ in the simulation with $\mathcal{S}_{P_1}$ is statistically close to its view in a hybrid execution of the protocol with a trusted party running the VOT (resp. COT) protocol. Since we assume that the VOT (resp. COT) protocol is secure, we analyze the security of the protocol in the hybrid model with a trusted party computing the VOT (resp. COT) functionality. Note that the simulator S_{P_1} knows X, sk_{P_1} and $d_H(X,Y)$. The simulator proceeds as follows:

1) $\mathcal{S}_{P_1}$ picks arbitrary $\tilde{Y} = \tilde{y_1} \ldots \tilde{y_n}$ and computes $\tilde{\mathsf{Commit}}_{P_{2,i}}$. $\mathcal{S}_{P_1}$ can simulate the proofs since it knows the committed input values $\tilde{y_i}$'s and sk_{P_1}.
2) In case of VOT is run:
   a) $\mathcal{S}_{P_1}$ first extracts the input of R_{P_1} from VOT functionality in the hybrid model, then sends the input to the trusted party and learns the output value $\tilde{t_i}$.
   b) $\mathcal{S}_{P_1}$ computes $\tilde{T} = \sum_{i=1}^{n} \tilde{t_i}$ and computes $\mathsf{Commit}_{P_{2,i}}(2R + n, \rho) = \prod_{i=1}^{n} A_i B_i$ as in the real protocol.

   In case of COT is run:

   a) $\mathcal{S}_{P_1}$ first extracts the input of $\mathcal{R}_{P_1}$ from COT functionality in the hybrid model, then sends the input to the trusted party and learns the output value $\tilde{t_i}$, $\tilde{\tau_i}$ and $\tilde{C_i} = \mathsf{Commit}(\tilde{t_i}, \tilde{\tau_i})$ $\forall i = 1, \ldots, n$.
   b) $\mathcal{S}_{P_1}$ computes $\tilde{T} = \sum_{i=1}^{n} \tilde{t_i}$ and $\mathsf{Commit}(\tilde{T}, \tilde{\tau}) = \prod_{i=1}^{n} \tilde{C_i}$ as in the real protocol.
   c) $\mathcal{S}_{P_1}$ can simulate the proof since it knows the committed input value $\tilde{T}$'s and sk_{P_1}.

Consequently, each step of the proposed authentication protocol for the simulator is simulated and this completes the simulation for the malicious verifier. The transcript is consistent and statistically indistinguishable from the verifier's view when interacting with honest P_2.

Case-2-P_2 is corrupted. Let $\mathcal{A}_{P_2}$ be an adversary corrupting P_2, we construct a simulator $\mathcal{S}_{P_2}$ as follows. Since we assume that the COT (resp. VOT) protocol is secure, we analyze the security of the protocol in the hybrid model with a trusted party computing the COT (resp. VOT) functionality. Note that the simulator $\mathcal{S}_{P_2}$ knows $Y = y_1 \ldots y_n, sk_{P_2}$ and $d_H(X,Y)$. The simulator proceeds as follows:

1) $\mathcal{S}_{P_2}$ picks arbitrary $\tilde{X} = \tilde{x_1} \ldots \tilde{x_n}$.
2) $\mathcal{S}_{P_2}$ picks $\tilde{r_i} \in_R \mathbb{Z}_q^*$ and computes $\tilde{R}_{P_2} = \sum_{i=1}^{n} \tilde{r_i}$. Next, $\mathcal{S}_{P_2}$ computes $(\tilde{a}_i, \tilde{b}_i) = (\tilde{r}_i + \tilde{x}_i, \tilde{r}_i + \bar{\tilde{x}}_i)$ $\forall i = 1 \ldots n$. $\mathcal{S}_{P_2}$ computes $\tilde{A}_i$, $\tilde{B}_i$ and $\tilde{R}_i$ as in the real protocol. $\mathcal{S}_{P_2}$ can again simulate the proofs since he knows the committed input values and sk_{P_2}.
3) In case VOT is run:
   a) $\mathcal{S}_{P_2}$ first extracts the input of $\mathcal{R}_{P_1}$ from VOT functionality in the hybrid model and then sends the input to the trusted party. S_{P_2} next computes $\tilde{K} = \mathsf{Commit}_{P_{2,i}}(2\tilde{R}+n, \rho)$. $\mathcal{S}_{P_2}$ can simulate the proof since it knows the committed input values and sk_{P_2}.

   In case COT is run:

   a) $\mathcal{S}_{P_2}$ first extracts the input of $\mathcal{R}_{P_1}$ from COT functionality in the hybrid model and then sends the input to the trusted party and learn C_i $\forall i = 1, \ldots, n$. $\mathcal{S}_{P_2}$ computes $\mathsf{Commit}(\tilde{T}, \tau) = \prod_{i=1}^{n} \tilde{C}_i$.

Consequently, each step of the proposed authentication protocol for the simulator is simulated and this completes the simulation for the malicious verifier. The transcript is consistent and statistically indistinguishable from the verifier's view when interacting with honest P_1. $\square$

$\mathbf{P}_1$		$\mathbf{P}_2$
$X = x_1 \dots x_n$ where $x_i \in \{0,1\}$, sk_{P_1}		$Y = y_1 \dots y_n$, $y_i \in \{0,1\}$, sk_{P_2}
		Compute $\mathsf{Commit}_{P_{2,i}}(y_i, \delta_i)$ $\forall i = 1 \dots n$, $\delta_i \in_R \mathbb{Z}_q^*$
	$\xleftarrow{\mathsf{Commit}_{P_{2,i}} + \text{Proofs}, \forall i=1\dots n}$	
Pick $r_1, \dots, r_n \in_R \mathbb{Z}_q^*$		
Compute $R = \sum_{i=1}^{n} r_i$		
Compute $(a_i, b_i) = (r_i + x_i, r_i + \bar{x_i})$ $\forall i = 1 \dots n$		
Compute $A_i = \mathsf{Commit}_{P_{1,i}}(a_i, \alpha_i)$ $\forall i = 1 \dots n$		
Compute $B_i = \mathsf{Commit}_{P_{1,i}}(b_i, \beta_i)$ $\forall i = 1 \dots n$		
Compute $R_i = \mathsf{Commit}_{P_{1,i}}(r_i, \gamma_i)$ $\forall i = 1 \dots n$		
	$\xrightarrow{<A_i, B_i + \text{Prove that } ((a_i - r_i) = 0 \text{ or } (b_i - r_i) = 0) \text{ and } \lvert b_i - a_i = 1 \rvert, \forall i = 1 \dots n>}$	
1^{st} **Option:**		
	$\xleftrightarrow{<\mathsf{VOT}((A_i, B_i); \mathsf{Commit}_{P_{2,i}}(y_i, \delta_i)) : \forall i = 1 \dots n>}$	
		Obtain t_i where $t_i = r_i + x_i \oplus y_i$ $\forall i = 1 \dots n$
		Compute $T \leftarrow \sum_{i=1}^{n} t_i$
Compute $K = \mathsf{Commit}_{P_{2,i}}(2R + n, \rho) = \prod_{i=1}^{n} A_i B_i$		Compute $\mathsf{Commit}_{P_{2,i}}(2R + n, \rho) = \prod_{i=1}^{n} A_i B_i$
	$\xrightarrow{R + \text{Prove that } K \text{ commits to } (2R + n)}$	
		$d_H(X, Y) = T - R$
2^{nd} **Option:**		
	$\xleftrightarrow{<\mathsf{COT}((A_i, B_i); \mathsf{Commit}_{P_{2,i}}(y_i, \delta_i)) : \forall i = 1 \dots n>}$	
		Obtain t_i, τ_i and $C_i = \mathsf{Commit}(t_i, \tau_i)$ where $t_i = r_i + x_i \oplus y_i$ $\forall i = 1 \dots n$
		Compute $T = \sum_{i=1}^{n} t_i$
Compute $\mathsf{Commit}(T, \tau) = \prod_{i=1}^{n} C_i$		Compute $C = \mathsf{Commit}(T, \tau) = \prod_{i=1}^{n} C_i$
	$\xleftarrow{T + \text{Prove that } C = \mathsf{Commit}(T, \tau)}$	
$d_H(X, Y) = T - R$		

Fig. 3: Our Improved Scheme

8. Complexity Analysis Of Our Fixed Protocol

In this section, we analyze the computational complexity of our fixed protocol and compare it with the full scheme of Bringer *et al.* [9]. In our protocol, the number of invoked zero-knowledge proofs and multiplication of ciphertexts remain the same. However, we have improved the efficiency of the protocol significantly by replacing n COTs with n VOTs in the second option of the protocol where P_2 computes the final Hamming distance. In this way, we show that n number commitments, $2n$ number of partial decryption and $2n$ number of ZK proofs can be removed. The number of commitments of P_1 is increased from $2n$ to $3n$ in order to guarantee the validity of P_1's inputs. This is the price that should be paid to make the protocol secure. The complexity comparison of the full scheme of Bringer *et al.* [9] and our fixed protocol is illustrated in Figure 4.

	Scheme of Bringer *et al.*		Our Fixed Scheme	
	P_1	P_2	P_1	P_2
Commitments	$3n$	n	$2n$	n
ZK proofs	n			
OTs	n COTs		1^{st} opt: n COTs 2^{nd} opt: n VOTs	
Multiplication of ciphertexts	1^{st} opt: n 2^{nd} opt: $2n$			

Fig. 4: Complexity Comparison

Our analysis shows that the additional cost of the security fix is only n number of commitments made by P_1, independent of the party which computes the final Hamming distance. However, in the case that P_2 computes the final Hamming distance, the computational savings that can be achieved by replacing the n COTs with n VOTs are far more larger. In general, a COT protocol requires one more flow than a VOT protocol in which the chooser recommits to its received value and proves that the new commitment equals to her previous committed input. In particular, the full scheme in [9] uses the COT scheme of [28] where each run of a COT protocol requires one commitment, two partial decryption of a ciphertext and two zero-knowledge proofs in addition to a VOT protocol. As a result, we avoid unnecessary use of two zero-knowledge proofs and two partial decryptions. Consequently, we improve the efficiency of the protocol significantly while we establish the security of the protocol.

9. Conclusion

Bringer *et al.* [9] proposed two Hamming distance computation schemes which can be applied to biometric authentication systems. In semi-honest setting, the basic scheme in [9] is the most efficient up to date. However, their full scheme is insecure in the malicious case.

In this paper, we show that the full scheme of Bringer *et al.* [9] has a critical security issue by attacking to the scheme. In our attack, we show that an adversary without having any prior knowledge can convince the verifier in a practical way. Moreover, we fix the protocol by placing a robust method for input validation without adding a significant cost. We also enhance the efficiency of their protocol significantly by showing that Verifiable Oblivious Transfer (VOT) is sufficient to use instead of Committed Oblivious Transfer (COT) in the second option of the full scheme. VOT reduction avoids the unnecessary computation of one commitment, two zero-knowledge proofs and two partial decryption of the ciphertext for each bit of the input.

REFERENCES

[1] abhi shelat and Chih hao Shen. Two-output secure computation with malicious adversaries. In *Advances in Cryptology EUROCRYPT 2011*, volume 6632 of *Lecture Notes in Computer Science*, pages 386–405. Springer Berlin Heidelberg, 2011.

[2] Yonatan Aumann and Yehuda Lindell. Security against covert adversaries: Efficient protocols for realistic adversaries. *Journal of Cryptology*, 23(2), 2010.

[3] M. Barni, T. Bianchi, D. Catalano, M. Di Raimondo, R.D. Labati, P. Failla, D. Fiore, R. Lazzeretti, V. Piuri, A. Piva, and F. Scotti. A privacy-compliant fingerprint recognition system based on homomorphic encryption and fingercode templates. In *Proceedings of the Fourth IEEE International Conference on Biometrics: Theory Applications and Systems (BTAS), 2010*, pages 1–7, 2010.

[4] Mauro Barni, Tiziano Bianchi, Dario Catalano, Mario Di Raimondo, Ruggero Donida Labati, Pierluigi Failla, Dario Fiore, Riccardo Lazzeretti, Vincenzo Piuri, Fabio Scotti, and Alessandro Piva. Privacy-preserving fingercode authentication. In *Proceedings of the 12th ACM Workshop on Multimedia and Security*, MMSec '10, pages 231–240. ACM, 2010.

[5] Dan Boneh, Eu-Jin Goh, and Kobbi Nissim. Evaluating 2-dnf formulas on ciphertexts. volume 3378 of *Lecture Notes in Computer Science*, pages 325–341. Springer Berlin Heidelberg, 2005.

[6] Julien Bringer and Herv Chabanne. An authentication protocol with encrypted biometric data. volume 5023 of *Lecture Notes in Computer Science*, pages 109–124. Springer Berlin Heidelberg, 2008.

[7] Julien Bringer, Hervé Chabanne, Malika Izabachène, David Pointcheval, Qiang Tang, and Sébastien Zimmer. An application of the goldwasser-micali cryptosystem to biometric authentication. In *Proceedings of the 12th Australasian conference on Information security and privacy*, ACISP'07, pages 96–106. Springer-Verlag, 2007.

[8] Julien Bringer, Hervé Chabanne, and Bruno Kindarji. *Security and Communication Networks*, 4(5):548–562, 2011.

[9] Julien Bringer, Herve Chabanne, and Alain Patey. Shade: Secure Hamming distance computation from oblivious transfer. In *Financial Cryptography and Data Security*, volume 7862 of *Lecture Notes in Computer Science*, pages 164–176. Springer Berlin Heidelberg, 2013.

[10] Ran Canetti. Security and composition of multi-party cryptographic protocols. *JOURNAL OF CRYPTOLOGY*, 13:2000, 1998.

[11] Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *FOCS*, pages 136–145, 2001.

[12] Ronald Cramer, Ivan Damgård, and Berry Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In *Proceedings of the 14th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '94, pages 174–187, London, UK, 1994. Springer-Verlag.

[13] Ivan Damgård, Martin Geisler, and Mikkel Krøigaard. Efficient and secure comparison for on-line auctions. In *Information Security and Privacy*, volume 4586 of *Lecture Notes in Computer Science*, pages 416–430. Springer, 2007.

[14] Ivan Damgård, Martin Geisler, and Mikkel Krøigaard. A correction to efficient and secure comparison for on-line auctions. *IJACT*, 1(4):323–324, 2009.

[15] Zekeriya Erkin, Martin Franz, Jorge Guajardo, Stefan Katzenbeisser, Inald Lagendijk, and Tomas Toft. Privacy-preserving face recognition. In *Proceedings of the 9th International Symposium on Privacy Enhancing Technologies*, PETS '09, pages 235–253, Berlin, Heidelberg, 2009. Springer-Verlag.

[16] Oded Goldreich. *Foundations of Cryptography: Volume 2, Basic Applications*. Cambridge University Press, 2004.

[17] Shafi Goldwasser and Silvio Micali. Probabilistic encryption & how to play mental poker keeping secret all partial information. In *Proceedings of the fourteenth annual ACM symposium on Theory of computing*, pages 365–377. ACM, 1982.

[18] Feng Hao, Ross Anderson, and John Daugman. Combining crypto with biometrics effectively. *IEEE Trans. Comput.*, 55(9):1081–1088, September 2006.

[19] Carmit Hazay and Yehuda Lindell. Efficient oblivious polynomial evaluation with simulation-based security (manuscript). *IACR Cryptology ePrint Archive*, page 459, 2009.

[20] Yuval Ishai, Joe Kilian, Kobbi Nissim, and Erez Petrank. Extending oblivious transfers efficiently. In *Advances in Cryptology - CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 145–161. Springer, 2003.

[21] Anil K. Jain, Arun Ross, and Salil Prabhakar. An introduction to biometric recognition. *IEEE Trans. on Circuits and Systems for Video Technology*, 14:4–20, 2004.

[22] Stanislaw Jarecki and Vitaly Shmatikov. Efficient two-party secure computation on committed inputs. In *Advances in Cryptology - EUROCRYPT 2007*, volume 4515 of *Lecture Notes in Computer Science*, pages 97–114. Springer Berlin Heidelberg, 2007.

[23] Ayman Jarrous and Benny Pinkas. Secure Hamming distance based computation and its applications. volume 5536 of *Lecture Notes in Computer Science*, pages 107–124. Springer Berlin Heidelberg, 2009.

[24] Ari Juels and Martin Wattenberg. A fuzzy commitment scheme. In *Proceedings of the 6th ACM Conference on Computer and Communications Security*, CCS '99, pages 28–36, New York, NY, USA, 1999. ACM.

[25] Sanjay Ganesh Kanade, Dijana Petrovska-Delacrtaz, and Bernadette Dorizzi. Cancelable iris biometrics and using error correcting codes to reduce variability in biometric data. In *CVPR*, pages 120–127. IEEE, 2009.

[26] Florian Kerschbaum, Mikhail J. Atallah, David M'Raihi, and John R. Rice. Private fingerprint verification without local storage. In *Biometric Authentication*, volume 3072 of *Lecture Notes in Computer Science*, pages 387–394. Springer Berlin Heidelberg, 2004.

[27] T. A. M. Kevenaar, G. J. Schrijen, M. van der Veen, A. H. M. Akkermans, and F. Zuo. Face recognition with renewable and privacy preserving binary templates. In *Proceedings of the Fourth IEEE Workshop on Automatic Identification Advanced Technologies*, AUTOID '05, pages 21–26, Washington, DC, USA, 2005. IEEE Computer Society.

[28] Mehmet Sabir Kiraz, Berry Schoenmakers, and Jose Villegas. Efficient committed oblivious transfer of bit strings. In *Information Security*, volume 4779 of *Lecture Notes in Computer Science*, pages 130–144. Springer Berlin Heidelberg, 2007.

[29] Vladimir Kolesnikov and Ranjit Kumaresan. Improved ot extension for transferring short secrets. In *Advances in Cryptology CRYPTO 2013*, volume 8043 of *Lecture Notes in Computer Science*, pages 54–70. Springer, 2013.

[30] Rohan Kulkarni and Anoop Namboodiri. Secure Hamming distance based biometric authentication. pages 1–6, June 2013.

[31] Yehuda Lindell and Benny Pinkas. An efficient protocol for secure two-party computation in the presence of malicious adversaries. In *Advances in Cryptology - EUROCRYPT 2007*, volume 4515 of *Lecture Notes in Computer Science*, pages 52–78. Springer Berlin Heidelberg, 2007.

[32] Jean-Paul Linnartz and Pim Tuyls. New shielding functions to enhance privacy and prevent misuse of biometric templates. In *Audio- and Video-Based Biometric Person Authentication*, volume 2688 of *Lecture Notes in Computer Science*, pages 393–402. Springer Berlin Heidelberg, 2003.

[33] Margarita Osadchy, Benny Pinkas, Ayman Jarrous, and Boaz Moskovich. Scifi - a system for secure face identification. pages 239–254, May 2010.

[34] Ahmad-Reza Sadeghi, Thomas Schneider, and Immo Wehrenberg. Efficient privacy-preserving face recognition. In *Proceedings of the 12th International Conference on Information Security and Cryptology*, ICISC'09, pages 229–244, Berlin, Heidelberg, 2010. Springer-Verlag.

[35] Anongporn Salaiwarakul and MarkD. Ryan. Analysis of a biometric authentication protocol for signature creation application. In Kanta Matsuura and Eiichiro Fujisaki, editors, *Advances in Information and Computer Security*, volume 5312 of *Lecture Notes in Computer Science*, pages 231–245. Springer Berlin Heidelberg, 2008.

[36] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *J. Cognitive Neuroscience*, 3(1):71–86, January 1991.

[37] Pim Tuyls, Anton H.M. Akkermans, Tom A.M. Kevenaar, Geert-Jan Schrijen, Asker M. Bazen, and Raimond N.J. Veldhuis. Practical biometric authentication with template protection. volume 3546 of *Lecture Notes in Computer Science*, pages 436–446. Springer Berlin Heidelberg, 2005.

[38] Pim Tuyls and Jasper Goseling. Capacity and examples of template-protecting biometric authentication systems. In *Biometric Authentication*, volume 3087 of *Lecture Notes in Computer Science*, pages 158–170. Springer Berlin Heidelberg, 2004.

M ehmet Sabir Kiraz received his M.S. degree from Max-Planck Institute for Computer Science, Saarbrucken, Germany in 2003. He received his Doctor of Philosophy degree from Eidhoven Technical University, the Netherlands in 2008. He is currently working at TUBITAK BILGEM National Research Enstitute of Electronics and Cryptology, Turkey. His current research interests include threshold cryptography, cryptographic protocols, privacy, (biometric) authentication and key management, secure multi-party computation and electronic voting.

Z iya Alper Genc is currently working at TUBITAK BILGEM National Research Enstitute of Electronics and Cryptology, Turkey. He is also M.S. student at Istanbul Sehir University. His research interests are in the are of cyptographic protocols, privacy, authentication.

S uleyman Kardas received his M.S. degree from Bilkent University, Ankara, Turkey in 2009. He is currently working at TUBITAK BILGEM National Research Enstitute of Electronics and Cryptology, Turkey. He is also a Ph.D. student in Computer Engineering from Sabanci University, Istanbul. His research interests are in the areas of cryptographic protocols, privacy, (biometric) authentication, secure multi-party computation and RFID systems.