SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks 2015; **00**:1–13
DOI: 10.1002/sec

RESEARCH ARTICLE

Security and Efficiency Analysis of the Hamming Distance Computation Protocol Based on Oblivious Transfer

Mehmet Sabır Kiraz[1]*, Ziya Alper Genç[1,3], Süleyman Kardaş[1,2]

[1]TÜBİTAK BİLGEM UEKAE, Kocaeli, Turkey
[2]Batman University, Faculty of Engineering and Architecture, Batman, Turkey
[3]İstanbul Şehir University, İstanbul, Turkey

ABSTRACT

Bringer *et al.* proposed two cryptographic protocols for the computation of Hamming distance. Their first scheme uses Oblivious Transfer and provides security in the semi-honest model. The other scheme uses Committed Oblivious Transfer and is claimed to provide full security in the malicious case. The proposed protocols have direct implications to biometric authentication schemes between a prover and a verifier where the verifier has biometric data of the users in plain form.
In this paper, we show that their protocol is not actually fully secure against malicious adversaries. More precisely, our attack breaks the soundness property of their protocol where a malicious user can compute a Hamming distance which is different from the actual value. For biometric authentication systems, this attack allows a malicious adversary to pass the authentication without knowledge of the honest user's input with at most $O(n)$ complexity instead of $O(2^n)$, where n is the input length. We propose an enhanced version of their protocol where this attack is eliminated. The security of our modified protocol is proven using the simulation-based paradigm. Furthermore, as for efficiency concerns, the modified protocol utilizes Verifiable Oblivious Transfer which does not require the commitments to outputs which improves its efficiency significantly.

KEYWORDS

Biometric Identification; Authentication; Hamming distance; Privacy; Committed Oblivious Transfer

***Correspondence**

Corresponding author's phone: +90 (262) 648 1945. E-mail: mehmet.kiraz@tubitak.gov.tr

Received . . .

1. INTRODUCTION

Recently, several commercial organizations have invested in secure electronic authentication systems to reliably verify identity of individuals. Biometric authentication systems are receiving a lot of public attention and becoming a crucial solution to many authentication and identity management problems because of cost-effective improvements in sensor technologies and in efficiency of matching algorithms [1]. Biometric data (i.e. templates) of a user is inherently unique. This uniqueness provides assurance to individuals to be securely authenticated for accessing an environment provided that the biometric data is kept as a secret. The biometric data cannot be directly used with conventional encryption techniques because the data itself is inherently noisy [2]. Namely, whenever two samples of data are extracted from the same fingerprint, they will not be exactly the same. In this context, in order to eliminate the noisy nature of the biometric templates, several error correction techniques were proposed in the literature [3, 4, 5].

Biometric authentication over an insecure network raises more security and privacy issues. The primary security issue is the protection of the plain biometric templates against a malicious adversary because they cannot be replaced with new ones, 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 a smart card (i.e, on-card matching), or in a system outside the card (i.e, off-card matching) [6]. Since the biometric template is not necessarily transferred to outside environment, the on-card matching technique

Prepared using secauth.cls [Version: 2010/06/28 v2.00]

protects the template. In both techniques, authentication protocols should not expose the biometric template without the user's agreement. In order to ensure privacy of the user, the biometric template should be stored in an encrypted form in a database and no one, including the server, can learn any information on the biometric data in plain form. But still, it should be possible to verify whether a user is authentic [7].

In order to thwart the security and privacy issues described above for 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, Workshop on Applied Homomorphic Cryptography (WAHC 13) (co-located with Financial Cryptography 2013), Bringer *et al.* [8] 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 is solely based 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. One can of course use one of the efficient OT protocols for the semi-honest setting [9, 10]. The second scheme uses Committed Oblivious Transfer (COT) and is claimed to provide full security against malicious adversaries. Since the protocol of Bringer *et al.* computes Hamming distance of bit strings, the authors utilizes the only COT protocol of bit strings that is due to Kiraz *et al.* [11].

1.1. Contributions

In this paper, we first revisit the Hamming distance computation protocol SHADE of Bringer *et al.* [8]. [12] generalizes and proposes improvements over [8] in the semi-honest setting. We show that SHADE is in fact not sound in the malicious model. More precisely, we show that the full scheme has a weakness allowing any malicious adversary to violate soundness property of the protocol, i.e., a different value of Hamming distance from the actual one.

The protocol flaw resides in the method used for validation of the inputs of a user. Using zero-knowledge proofs, the protocol aims to force the user 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 bogus pairs $(\tilde{x}, \tilde{y})$ and can pass the verification steps without being detected. Since SHADE computes the Hamming distance by using the outputs of COT, a verifier would compute an incorrect Hamming distance. We would like to highlight that any fake Hamming distance can be set in advance. As a practical example for biometric authentication, we show that a malicious adversary can pass the authentication by running the algorithm at most $O(n)$ times (instead of running $O(2^n)$ times, where n is the input length.). Last but not least, an adversary with knowledge of the distribution of inputs can mount a more powerful attack. Note that this attack is of independent interest and may be applied to other schemes.

In order to eliminate this weakness, we propose a new method for input validation. This way, we remove the fault in the protocol and enhance 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 the ideal/real simulation paradigm in the standard model [13, 14, 15] and [16].

Lastly, we consider the efficiency of the protocol and show that running a COT is not necessary in the full scheme of the protocol. We show that VOT is sufficient instead of using complete COT protocol which contains additional commitments and zero-knowledge proofs [17]. This leads to a considerable improvement in the computational complexity of the protocol.

1.2. Organization

Section 2 gives related work on the computation of Hamming distance and biometric authentication systems. Section 3 provides security and privacy model for biometric authentication protocols. Section 4 reviews the two schemes in the protocol, the basic scheme which uses OTs and the full scheme based on COT of bit-strings. In Section 5, we present an attack on the full scheme of Bringer *et al.* which breaks the soundness property. 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 the simulation-based paradigm. The complexity analysis of the proposed protocol is shown in Section 8. Finally, Section 9 concludes the paper.

2. RELATED WORK

There has been a large amount of research done on the security and efficiency of the 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 the *bin*HDOT protocol [18] to compute Hamming distance based on 1-out-of-2 Committed Oblivious Transfer with Constant Difference (COTCD) of Jarecki and Shmatikov [19] and Oblivious Polynomial Evaluation (OPE) of Hazay and Lindell [20]. The protocol also uses commitments and zero-knowledge proofs to guarantee that each party follows the protocol. This protocol provides full

Security Comm. Networks 2015; **00**:1–13 © 2015 John Wiley & Sons, Ltd.
DOI: 10.1002/sec
Prepared using secauth.cls

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 [21]. SCiFI system consists of two parts: a client and a server. The server prepares a face recognition database that contains representations of face images. This computation is done offline. In the verification phase, a client prepares her face representation and then a cryptographic protocol which uses Paillier encryption and Oblivious Transfer running 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.* [22] used biometric authentication/identification for access control. Note that it is important to securely store the biometric template on the server and using conventional encryption schemes for securing the biometric template can provide a strong protection. Note also that conventional cryptography requires an exact match while biometrics always have a threshold value, therefore biometric authentication over the encrypted domain is a challenging task. In [22], a cryptographic scheme is given for biometric identification over an encrypted domain which uses Bloom Filters with Storage and Locality-Sensitive Hashing. Their paper is interesting because 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.* [23] proposed a security model for biometric-based authentication protocols, relying on the Goldwasser-Micali cryptosystem [24]. 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.* [25] propose a privacy preserving face recognition system on encrypted messages which is based on the standard Eigenface recognition system [26]. In their protocol design, they utilized semantically secure Paillier homomorphic public-key encryption schemes and Damgård, Geisler and Krøigaard (DGK) cryptosystem [27, 28]. Later, Sadeghi *et al.* make an improvement over the efficiency of this system [29] by merging the eigenface recognition algorithm using homomorphic encryption and Yao's garbled circuits. Their protocol improves the scheme proposed by Erkin *et al.* significantly since it has only a constant number of rounds and most of the computation and communication is performed during the pre-computation phase. Schneider and Zohner [30] provide an improvement over [29] and [21] by using the GMW protocol [31].

Tuyls *et al.* [32] propose a template protection scheme for fingerprint based authentication in order to protect biometric data. During the enrollment phase, client's biometric features X is extracted, the Helper Data [33] W is computed (that is required by the error-correction mechanism), a one-way hash function $\mathcal{H}$ is applied to S and the data (client, W, $\mathcal{H}(S)$) is stored on the server. Here, S is a randomly chosen secret value such that $G(X, W)=S$ for a shielding function G [34]. During the verification phase, after client'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. Tuyls *et al.* however designed the system in such a way that the adversary obtains minimal information about X by capturing W.

Kulkarni *et al.* [35] propose a biometric authentication scheme based on Iris Matching. Their scheme uses the *somewhat* homomorphic encryption scheme of Boneh *et al.* [36] which allows an arbitrary number of additions of ciphertexts but supports only one multiplication operation between the ciphertexts. The scheme is based on Paillier encryption and bilinear pairings. This scheme consists of two phases: Enrollment phase and Verification phase. During the Enrollment phase, necessary keys are first generated by the server and then 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 the authenticated biometric data to compute the distance. The protocol is proven to be secure in the semi-honest model.

Kerschbaum *et al.* [37] propose an authentication scheme in a different setting. In particular, they assume that there are two parties where each of them has 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 is secure only in the semi-honest model using secure multi-party computation as a building block.

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 [38, 39]. 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 in the standard model which is already highlighted in [13, 14, 15] and [16]. In 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 an ideal functionality $\mathcal{F} = (\mathcal{F}_1, \mathcal{F}_2)$ that performs the computation and returns its result.

In an ideal setting, the parties send their inputs x and y to an ideal functionality $\mathcal{F}$ who computes $\mathcal{F}(x, y)$ (which is the output of the Hamming distance in our setting) and sends $\mathcal{F}_1(x, y)$ to the first party and $\mathcal{F}_2(x, y)$ to the second party ($\mathcal{F}_1(x, y)$ or $\mathcal{F}_2(x, y)$ 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 to the functionality $\mathcal{F}$, while the honest party always sends its specified input. In a real execution of a protocol $\Pi_{\mathcal{F}}$ for a functionality $\mathcal{F}$, one of the parties is assumed to be corrupted under the complete control of an adversary $\mathcal{A}$. Note that we 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 (as it is known as static adversary model).

Informally, a protocol $\Pi_{\mathcal{F}}$ 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 $\mathcal{F}$ and the protocol $\Pi_{\mathcal{F}}$ be as above. We say that the protocol $\Pi_{\mathcal{F}}$ 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 also that VOT and COT protocols are used as subprotocols. In [40, 41], 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 plays 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 of [8] 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) [11] and zero-knowledge proofs of knowledge [17] 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.

4.1. 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 are sufficient to preserve privacy.

The basic scheme in [8] which is secure against semi-honest adversaries is as follows:

Two parties P_1 and P_2 are willing to compute the Hamming distance of their private inputs $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_n\}$, 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 $i = 1, \cdots, n$, the parties run an OT protocol in which P_1 acts as the sender and P_2 acts as the receiver. More precisely, P_1 inputs $(r_i + x_i, r_i + \tilde{x}_i)$ where $\tilde{x}_i = 1 - x_i$ and P_2 inputs y_i. At the end of the OT protocol, P_2 receives $t_i = (r_i + x_i)$ if $y_i = 0$ and $(t_i = r_i + \tilde{x}_i)$ 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. Next, P_1 outputs $T - R$.
- 2^{nd} **Option:** P_1 sends R to P_2. Next, P_2 outputs $T - R$.

The privacy is still guaranteed in the presence of semi-honest adversaries as they proved in Section 6 of [8]. Furthermore, the efficiency of the basic scheme of Bringer *et al.* [8] was further improved in [12]. 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 [42] and later improved in [43]. 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

Security Comm. Networks 2015; **00**:1–13
DOI: 10.1002/sec
Prepared using secauth.cls

semi-honest adversaries (note that hash functions can be replaced with RO model in the real case).

4.2. 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 [11] (see Figure 1). 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 (refer to [8] for more details).

- At the first step of the protocol, P_2 commits to her input bits y_i's and proves in zero-knowledge [17] that each y_i is either equal to 0 or equal to 1.
- At the same time, P_1 generates random elements r_i's from the plaintext space of the commitment scheme and computes $R = \sum_{i=1}^{n} r_i$. Next, she commits to a_i and b_i where $(a_i, b_i) = (r_i + x_i, r_i + \tilde{x}_i)$ [*]. Let's denote $\mathsf{Commit}(M)$ for the commitment functionality of a message M [†]. P_1 publishes the commitments $A_i = \mathsf{Commit}(a_i)$ and $B_i = \mathsf{Commit}(b_i)$. Furthermore, using these commitments she proves that a_i and b_i differ by 1 for each i.
- 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 $C_i = \mathsf{Commit}(t_i)$. When all the COTs are run, P_2 computes the sum $T = \sum_{i=1}^{n} t_i$.
- At this point, there are two options:
  - 1^{st} **Option:** P_2 computes $C = C_1 \cdots C_n$, and because of the underlying homomorphic property we have $\mathsf{Commit}(T) = C$ [8]. P_2 sends T to P_1 and proves in zero-knowledge that C indeed commits to T. P_1 also computes $C = C_1 \cdots C_n$ and verifies the proof. If all verifications are successful, P_1 outputs $T - R$.
  - 2^{nd} **Option:** P_1 computes $K = \mathsf{Commit}(2R + n) = A_1 \cdots A_n \cdot B_1 \cdots B_n$. P_1 sends R to P_2 and proves in zero-knowledge that K indeed commits to $2R + n$. P_2 computes $K = A_1 \cdots A_n \cdot B_1 \cdots B_n$ and verifies that $K = \mathsf{Commit}(2R + n)$. If all verifications are successful, P_2 outputs $T - R$.

The authors in [8] claim that the above scheme is fully secure against malicious adversaries. However, in the next section we show that a malicious P_1 can easily break the soundness property of the scheme.

5. SECURITY AND EFFICIENCY ANALYSIS OF THE PROTOCOL BY 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 change the Hamming distance between her input and P_2's input. In the next section, we propose a solution to fix the flaw by designing 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 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 point out that Verifiable Oblivious Transfer is sufficient in the case that P_1 computes the Hamming distance. This eliminates to compute n commitments together with the zero-knowledge proofs (for each run of COT protocol). In this way, we improves the efficiency of the protocol by using VOT instead of COT when P_1 is the server.

5.1. Attack to the Full Scheme

The protocol is not sound 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 encrypted 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 $(r_i + x_i, r_i + \tilde{x}_i)$. However, a malicious P_1 may choose invalid pairs in a special way together with the proofs that difference between each pair is equal to 1. Our attack uses the fact that at the end of each COT, P_2 receives either $t_i = r_i + g$ or $t_i = r_i + h$ and computes the sum $T = \sum_{i=1}^{n} t_i$, where g, h are within the finite cyclic group. Note that g is expected to be equal to x_i and h to $\tilde{x}_i$. However, with a careful choosing of g's and h's, some g's can be neutralized by some h's in this sum. Hence, the soundness property of the protocol can be violated. In fact,

[*] The commit functionality of [11] is basically a (2,2)-threshold homomorphic encryption scheme (e.g., ElGamal [44], Paillier [45]). Let $(pk_{P_1,P_2}, (sk_{P_1}, sk_{P_2}))$ denote public and private key pairs of the encryption scheme where pk_{P_1,P_2} is the common public key, and sk_{P_1}, sk_{P_2} are the corresponding private key shares of P_1 and P_2, respectively.

[†] Note that because of the underlying encryption scheme Commit includes randomness and public key, and we hide them for the sake of simplicity.

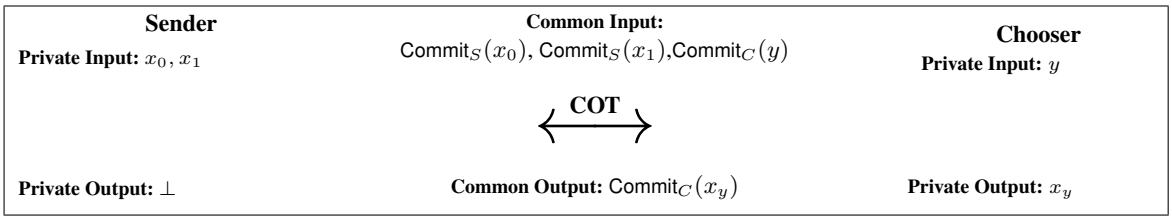

Figure 1. Committed Oblivious Transfer

the security proof of [8] does not explicitly use the zero-knowledge proof of the statement leading to the flaw in their security analysis.

Before we describe the attack it is important to highlight that the underlying COT scheme uses threshold ElGamal encryption as a commitment mechanism, i.e. $\mathsf{Commit}(x_i) = \mathsf{Enc}(x_i)$ where $x_i \in G$ where G is a large finite cyclic group (of a prime order) [11]. This guarantees the existence of the inverse of n.

Without loss of generality assume that #0's in P_2's input Y is ℓ (i.e., #1's in Y is $n-\ell$). A predetermined fake Hamming distance can be computed with the knowledge of #0's (similarly #1's) in P_2 as follows: a malicious P_1 uses $(a_i, b_i) = (r_i + g, r_i + h)$ for an arbitrary Hamming distance $\mathsf{HD} = \ell g + (n-\ell)h$ such that $g - h = 1$, where g, h are the group elements. Then,

$$\mathsf{HD} = \ell g + (n-\ell)(g-1) = ng - n + \ell.$$

For an example, if a malicious P_1 desires Hamming distance HD to be 0 then she chooses $g = 1 - \ell n^{-1}$. Next, h= g-1= $-\ell n^{-1}$. Hence, P_1 may use (a_i, b_i)=$(r_i + (1 - \ell n^{-1}), r_i - \ell n^{-1})$ as input. 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 computes (a_i, b_i)=$(r_i + (1 - \ell n^{-1}), r_i - \ell n^{-1})$ and publishes $A_i = \mathsf{Commit}(a_i)$ and $B_i = \mathsf{Commit}(b_i)$. Note that for each i, $|b_i - a_i| = 1$ and hence, the proofs pass successfully.
- At the end of each COT, P_2 receives either $t_i = r_i + (1 - \ell n^{-1})$ or $t_i = r_i - \ell n^{-1}$. After COTs are run, P_2 computes the sum

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

Therefore, the Hamming distance $d_H(X, Y)$=$T - R$ is equal to 0. We stress that the weakness in the scheme is destructive as we prove that a relatively insignificant information leakage causes computation of a completely inaccurate result. Namely, without knowledge of the real X, P_1 fools P_2 into outputting an incorrect Hamming distance value without being detected. Furthermore, a malicious P_1 with the prior knowledge of ℓ is capable of manipulating HD by computing the values g and h using the above-mentioned equation. This is interesting because Hamming distance is not necessarily equal to 0 or 1. For example, in [46], the authors propose a privacy-preserving protocol for iris-based authentication using Yao's garbled circuits. They show that Hamming distance between two iris codes owned by the same person is rarely close to 0 (and similarly it is rarely close to n for different persons). Therefore, the scalability feature of our attack can be easily adopted to various general settings.

In this part, we propose the most general case and in the next section we give a practical attack for biometric authentication schemes reducing the computational complexity of an attacker from $O(2^n)$ to $O(n)$, where n is the input length. Namely, an attacker without any prior knowledge can authenticate herself using only n trials instead of 2^n.

5.2. A Special Case: Apply the Attack to Biometric Authentication Systems

In the previous section, we described the most general case, i.e., for any system that uses the proposed Hamming distance protocol. We now apply the proposed attack as a practical example on biometric authentication systems with full success. Note that the matching procedure

Security Comm. Networks 2015; **00**:1–13 © 2015 John Wiley & Sons, Ltd.
DOI: 10.1002/sec
Prepared using secauth.cls

for fingerprint, palm print or iris actually measures the Hamming distance between the two bit-strings X and Y that encode the biometric sample and template (e.g., [7, 47, 35]).

The attack basically consists of n runs of the proposed attack method to successfully authenticate to the system, where n is the input length. In general, for an n-bit string $Y = (y_1, \ldots, y_n)$, an attacker must roughly try 2^n search for X to pass the authentication successfully and it is infeasible for large n. However, using the proposed attack a corrupted P_1 i can authenticate the system after at most n trials (because the number of 0s or 1s in Y is between 0 and n, i.e., $0 \leq \ell \leq n$). More precisely, starting $\ell = 1$ until $\ell = n$ a corrupted P_1 runs the proposed attack method, and because $0 \leq \ell \leq n$ the authentication is successful with at most n trials (without any knowledge of the real input X).

5.3. Apply the Attack for Uniformly Distributed Inputs

This attack can also be directly applied to uniformly distributed bit strings X and Y. In this scenario 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 bit string. Below, we show that this fact easily allows an adversary to minimize the Hamming distance and successfully deceive a verifier:

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 $(a_i, b_i) = (r_i + x_i, r_i + \tilde{x}_i)$, P_1 computes $(a_i, b_i) = (r_i - 2^{-1}, r_i + 2^{-1})$ in order to make the commitments $A_i = \mathsf{Commit}_{P_{1,i}}(a_i)$ and $B_i = \mathsf{Commit}_{P_{1,i}}(b_i)$. The authors in [8] 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 $T = \sum_{i=1}^{n} t_i$. Since y_i's are equally distributed, i.e. the numbers of 0s and 1s in $\{y_1, \ldots, y_n\}$ are nearly equal, P_2 computes $T = \left(\sum_i r_i + 2^{-1}\right) + \left(\sum_i r_i - 2^{-1}\right) = \sum_{i=1}^{n} r_i = R \pm k2^{-1}$ for some small $k \ll n$.
6. Using the 2^{nd} option, $K = \mathsf{Commit}_{P_{2,i}}(2R + n) = A_1 \cdots A_n \cdot B_1 \cdots B_n$.
7. P_1 sends R and the proof that K commits to $2R + n$ to P_2.
8. P_2 computes $d_H(X, Y) = T - R = k$ where $k \ll n$ and successfully authenticates P_1 since k will be less than the threshold value.

5.4. Our Solution for the Attack

The weakness of the full scheme is due to the zero-knowledge proof of a wrong statement used for validation of the input pairs $\{(a_i, b_i), \forall i = 1, \ldots, n\}$. A malicious P_1 can easily exploit this weakness as described in the previous section. Therefore, logical statements of zero-knowledge proofs should be carefully checked against these kinds of adversarial behaviors.

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 computes and publishes $A_i = \mathsf{Commit}(r_i + x_i), B_i = \mathsf{Commit}(r_i + \tilde{x}_i)$ and $R_i = \mathsf{Commit}(r_i)$. Next, P_1 sends the zero-knowledge proof of the following statement

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

that is equivalent to

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

using the commitments A_i, B_i and R_i. This new statement contains one more relation than the one in the original proof of [8]. Although the computation cost of the protocol is slightly increased, the validation process now assures the security of the protocol.

Note that if the new statement $(a_i + b_i - 2r_i = 1) \wedge |b_i - a_i| = 1$ is true then only one of the following two cases can occur:

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

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

5.4.1. 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 system 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 than 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

sends 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 is an efficient solution for our attack in the case of biometric authentication. Note that an adversary may input $(a_i, b_i) = (r_i - 2^{-1}, r_i + 2^{-1})$ and pass the validation but its Hamming distance is $\frac{n}{2}$ which is the expected value of Hamming distance between two random inputs with length n.

5.5. 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 is not necessary in the case where P_2 computes the final Hamming distance. Next we reduce the complexity of the proof for the validity of P_1's inputs in the case of biometric authentication.

5.5.1. COT versus VOT

Verifiable Oblivious Transfer (like COT) [48] is also a natural combination of $\binom{2}{1}$-OT and commitments. Let Commit_S and Commit_C be commitments by Sender and Chooser respectively. In a VOT protocol, the Sender has (x_0, x_1), the Chooser has $y \in \{0,1\}$ and the commitments $\mathsf{Commit}_S(x_0)$, $\mathsf{Commit}_S(x_1)$,$\mathsf{Commit}_C(y)$ are common input. At the end of the protocol the Chooser learns x_y and the sender has no output. Note that the difference with COT is that commitment to the output x_y is not computed, i.e., VOT is defined if the $\mathsf{Commit}_C(x_y)$ is not required as output. The functionality of VOT is illustrated in Figure 2.

We note here the two main aspects of COT vs. VOT:

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

$$\text{Committed Output} \begin{cases} \text{yes} \rightarrow \textbf{COT} \\ \text{no} \rightarrow \textbf{VOT} \end{cases}$$

We show that the basic protocol in [8] does not have to use COT in the case that the server computes the result (i.e., VOT is already sufficient because it is not necessary to compute the final commitment.).

5.5.2. Efficiency Improvement Using VOT

In this section, we point out a computational complexity reduction. Note that COT is run for the malicious case in [8]. 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 $X = (x_1, \ldots, x_n)$ and the input of P_2 is an n-bit string $Y = (y_1, \ldots, y_n)$. 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 is not used. In such case, these commitments are not necessary to be computed, and therefore VOT is 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 is still necessary to use.

6. OUR FIXED AND IMPROVED SCHEME

We made the modifications to the full scheme of [8] in order to fix the security weakness described in Section 5 and improve the efficiency of the protocol as mentioned in Section 5.5. 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 $(a_i, b_i, r_i) = (r_i + x_i, r_i + \tilde{x}_i, r_i)$. P_1 publishes $A_i = \mathsf{Commit}(a_i), B_i = \mathsf{Commit}(b_i)$ and $R_i = \mathsf{Commit}(r_i)$.
4. P_1 proves that $(|a_i - r_i| = 0 \vee |b_i - r_i| = 0) \wedge |b_i - a_i| = 1$ using A_i, B_i and R_i.
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 $t_i = r_i + (x_i \oplus y_i)$.
   - Both parties obtain $C_i = \mathsf{Commit}(t_i)$.
6. P_2 computes $T = \sum_{i=1}^{n} t_i$
7. 1^{st} **Option: Run VOT**
   (a) P_1 computes $K = \mathsf{Commit}(2R + n) = A_1 \cdots A_n \cdot B_1 \cdots B_n$.

Security Comm. Networks 2015; **00**:1–13 © 2015 John Wiley & Sons, Ltd.
DOI: 10.1002/sec
Prepared using secauth.cls

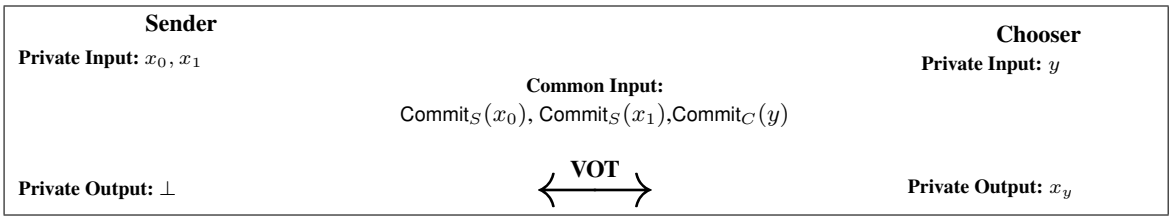

Figure 2. Verifiable Oblivious Transfer

(b) P_1 sends R to P_2 and proves that K commits to $2R + n$.
(c) P_2 computes $K = A_1 \cdot \ldots A_n \cdot B_1 \cdots B_n$ and checks that $K = \mathsf{Commit}(2R + n)$.
(d) If all verifications are successful, P_2 outputs $T - R$.

2^{nd} Option: Run COT

(a) P_2 computes $C = \mathsf{Commit}(T) = C_1 \cdots C_n$.
(b) P_2 sends T to P_1 and proves that C commits to T.
(c) P_1 computes $C = C_1 \cdots C_n$ and verifies 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 proved the security of the proposed protocol by constructing a simulator, which is given only the input and output of the "corrupted" party, and generating a view that is indistinguishable from the view of the adversary in a real protocol execution [13, 14, 15, 16]. 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} for the 1^{st} option where VOT is run (in the 2^{nd} the simulator also knows $d_H(X, Y)$). The simulator proceeds as follows:

1. $\mathcal{S}_{P_1}$ picks arbitrary $\tilde{Y} = \tilde{y}_1 \ldots \tilde{y}_n$ and computes $\mathsf{Commit}_{P_{2,i}}$. $\mathcal{S}_{P_1}$ 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 $\mathsf{Commit}_{P_{2,i}}(2R + n) = \prod_{i=1}^{n} A_i B_i$ 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 $\tilde{t}_i$ and $\tilde{C}_i = \mathsf{Commit}(\tilde{t}_i)$ $\forall i = 1, \ldots, n$.
   (b) $\mathcal{S}_{P_1}$ computes $\tilde{T} = \sum_{i=1}^{n} \tilde{t}_i$ and $\mathsf{Commit}(\tilde{T}) = \prod_{i=1}^{n} \tilde{C}_i$ as in the real protocol.
   (c) $\mathcal{S}_{P_1}$ can simulate the proof since it knows the committed input value $\tilde{T}$'s, $d_H(X, Y)$ 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

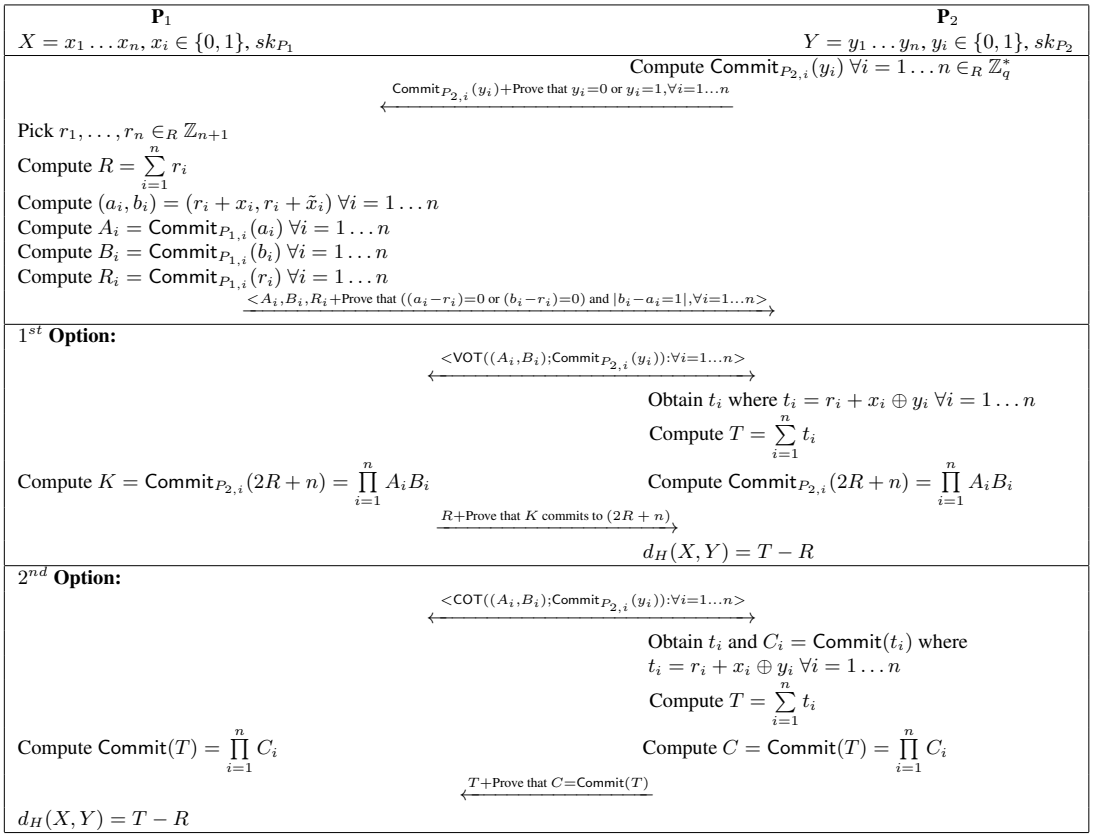

Figure 3. Our Improved Scheme

model with a trusted party computing the COT (resp. VOT) functionality. Note that the simulator $\mathcal{S}_{P_2}$ knows $Y = y_1 \dots y_n, sk_{P_2}$ and $d_H(X,Y)$ for the 1^{st} option where VOT is run (in the 2^{nd} the simulator does not know $d_H(X,Y)$). The simulator proceeds as follows:

1. $\mathcal{S}_{P_2}$ picks arbitrary $\tilde{X} = \tilde{x}_1 \dots \tilde{x}_n$.
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 $(\tilde{a}_i, \tilde{b}_i) = (\tilde{r}_i + \tilde{x}_i, \tilde{r}_i + \tilde{x}_i)$ $\forall i = 1 \dots n$. $\mathcal{S}_{P_2}$ 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 $\tilde{K} = \mathsf{Commit}_{P_{2,i}}(2\tilde{R} + n)$. $\mathcal{S}_{P_2}$ can simulate the proof since it knows the committed input value R, $d_H(X,Y)$ 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 $C_i\ \forall i = 1, \dots, n$. $\mathcal{S}_{P_2}$ computes $\mathsf{Commit}(\tilde{T}) = \prod_{i=1}^{n} \tilde{C}_i$.

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. □

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.* [8]. In our protocol, the number of invoked zero-knowledge proofs and multiplication of ciphertexts remain the same. However, we 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 commitments, $2n$ partial decryptions and $2n$ ZK proofs can be removed. The number of commitments of P_1 is increased from $2n$ to $3n$ in order to guarantee the

DOI: 10.1002/sec
Prepared using secauth.cls

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.* [8] and our fixed protocol is illustrated in Figure I.

	Scheme of Bringer *et al.*		Our Fixed Scheme	
	P_1	P_2	P_1	P_2
Commitments	$2n$	n	$3n$	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: $2n$			

Table I. Complexity Comparison

Our analysis shows that the additional cost of the security fix is only n 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 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 [8] uses the COT scheme of [11] 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.* [8] proposed two Hamming distance computation schemes which can be applied to biometric authentication systems. Their basic scheme is secure in the semi-honest setting. However, their full protocol is not sound in the malicious model.

In this paper, we show that the full scheme of Bringer *et al.* [8] has an issue with respect to soundness. In our attack, we show that an adversary without having any prior knowledge can make the verifier compute an incorrect Hamming distance. In the case of biometric authentication systems, a malicious user can easily authenticate without any information about the honest party. Namely, the complexity of the security of the system is reduced from $O(2^n)$ to $O(n)$, where n is the input length. 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. The VOT reduction avoids the unnecessary computation of one commitment, two zero-knowledge proofs and two partial decryptions of the ciphertext for each bit of the input.

ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers for their useful comments. Kiraz's work is partly supported by a grant from Ministry of Development of Turkey provided to the Cloud Computing and Big Data Research Lab Project. Kiraz's work is also partly supported by the COST Action CRYPTACUS (IC1403).

REFERENCES

1. Jain AK, Ross A, Prabhakar S. An introduction to biometric recognition. *Circuits and Systems for Video Technology, IEEE Transactions on* Jan 2004; **14**(1):4–20, doi:10.1109/TCSVT.2003.818349.
2. A M Kevenaar T, Schrijen GJ, van der Veen M, Akkermans AHM, Zuo F. Face recognition with renewable and privacy preserving binary templates. *Automatic Identification Advanced Technologies, 2005. Fourth IEEE Workshop on*, 2005; 21–26, doi: 10.1109/AUTOID.2005.24.
3. Hao F, Anderson R, Daugman J. Combining crypto with biometrics effectively. *IEEE Trans. Comput.* Sep 2006; **55**(9):1081–1088, doi:10.1109/TC.2006.138.
4. Kanade S, Petrovska-Delacretaz D, Dorizzi B. Cancelable iris biometrics and using error correcting codes to reduce variability in biometric data. *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2009; 120–127, doi:10.1109/CVPR.2009.5206646.
5. Juels A, Wattenberg M. A fuzzy commitment scheme. *Proceedings of the 6th ACM Conference on Computer and Communications Security*, CCS '99, ACM: New York, NY, USA, 1999; 28–36, doi:10.1145/319709.319714.
6. Salaiwarakul A, Ryan MD. Analysis of a biometric authentication protocol for signature creation application. *Advances in Information and Computer Security*, *Lecture Notes in Computer Science*, vol. 5312. Springer Berlin Heidelberg, 2008; 231–245, doi:10.1007/978-3-540-89598-5_16.
7. Bringer J, Chabanne H. An authentication protocol with encrypted biometric data. *Progress in Cryptology – AFRICACRYPT 2008*, *Lecture Notes in Computer Science*, vol. 5023. Springer Berlin Heidelberg, 2008; 109–124, doi:10.1007/978-3-540-68164-9_8.
8. Bringer J, Chabanne H, Patey A. Shade: Secure hamming distance computation from oblivious transfer. *Financial Cryptography and Data Security*,

Lecture Notes in Computer Science, vol. 7862. Springer Berlin Heidelberg, 2013; 164–176, doi:10.1007/978-3-642-41320-9_11.

9. Asharov G, Lindell Y, Schneider T, Zohner M. More efficient oblivious transfer and extensions for faster secure computation. *2013 ACM SIGSAC Conference on Computer and Communications Security, CCS'13, Berlin, Germany, November 4-8, 2013*, Sadeghi A, Gligor VD, Yung M (eds.), ACM, 2013; 535–548.
10. Chou T, Orlandi C. The simplest protocol for oblivious transfer. *IACR Cryptology ePrint Archive* 2015; **2015**:267. URL http://eprint.iacr.org/2015/267.
11. Kiraz MS, Schoenmakers B, Villegas J. Efficient committed oblivious transfer of bit strings. *Information Security*, *Lecture Notes in Computer Science*, vol. 4779. Springer Berlin Heidelberg, 2007; 130–144, doi:10.1007/978-3-540-75496-1_9.
12. Bringer J, Chabanne H, Favre M, Patey A, Schneider T, Zohner M. Gshade: Faster privacy-preserving distance computation and biometric identification. *Proceedings of the 2Nd ACM Workshop on Information Hiding and Multimedia Security*, IH&MMSec '14, ACM: New York, NY, USA, 2014; 187–198.
13. Canetti R. Security and composition of multiparty cryptographic protocols. *Journal of Cryptology* 2000; **13**(1):143–202, doi:10.1007/s001459910006.
14. Goldreich O. *Foundations of Cryptography: Volume 2, Basic Applications*. Cambridge University Press: New York, NY, USA, 2004.
15. Lindell Y, Pinkas B. An efficient protocol for secure two-party computation in the presence of malicious adversaries. *Advances in Cryptology - EUROCRYPT 2007*, *Lecture Notes in Computer Science*, vol. 4515. Springer Berlin Heidelberg, 2007; 52–78, doi:10.1007/978-3-540-72540-4_4.
16. Shelat A, Shen CH. Two-output secure computation with malicious adversaries. *Proceedings of the 30th Annual International Conference on Theory and Applications of Cryptographic Techniques: Advances in Cryptology*, EUROCRYPT'11, Springer-Verlag, 2011; 386–405.
17. Cramer R, Damgård I, Schoenmakers B. Proofs of partial knowledge and simplified design of witness hiding protocols. *Advances in Cryptology – CRYPTO '94*, *Lecture Notes in Computer Science*, vol. 839. Springer Berlin Heidelberg, 1994; 174–187, doi:10.1007/3-540-48658-5_19.
18. Jarrous A, Pinkas B. Secure hamming distance based computation and its applications. *Applied Cryptography and Network Security*, *Lecture Notes in Computer Science*, vol. 5536. Springer Berlin Heidelberg, 2009; 107–124, doi:10.1007/978-3-642-01957-9_7.
19. Jarecki S, Shmatikov V. Efficient two-party secure computation on committed inputs. *Advances in Cryptology - EUROCRYPT 2007*, *Lecture Notes in Computer Science*, vol. 4515. Springer Berlin Heidelberg, 2007; 97–114, doi:10.1007/978-3-540-72540-4_6.
20. Hazay C, Lindell Y. Efficient oblivious polynomial evaluation with simulation-based security. Cryptology ePrint Archive, Report 2009/459 2009. http://eprint.iacr.org/.
21. Osadchy M, Pinkas B, Jarrous A, Moskovich B. Scifi - a system for secure face identification. *Security and Privacy (SP), 2010 IEEE Symposium on*, 2010; 239–254, doi:10.1109/SP.2010.39.
22. Bringer J, Chabanne H, Kindarji B. Identification with encrypted biometric data. *Security and Communication Networks* 2011; **4**(5):548–562, doi:10.1002/sec.206.
23. Bringer J, Chabanne H, Izabachéne M, Pointcheval D, Tang Q, Zimmer S. An application of the goldwasser-micali cryptosystem to biometric authentication. *Information Security and Privacy*, *Lecture Notes in Computer Science*, vol. 4586. Springer Berlin Heidelberg, 2007; 96–106, doi:10.1007/978-3-540-73458-1_8.
24. Goldwasser S, Micali S. Probabilistic encryption & how to play mental poker keeping secret all partial information. *Proceedings of the Fourteenth Annual ACM Symposium on Theory of Computing*, STOC '82, ACM, 1982; 365–377, doi:10.1145/800070.802212.
25. Erkin Z, Franz M, Guajardo J, Katzenbeisser S, Lagendijk I, Toft T. Privacy-preserving face recognition. *Privacy Enhancing Technologies*, *Lecture Notes in Computer Science*, vol. 5672. Springer Berlin Heidelberg, 2009; 235–253, doi:10.1007/978-3-642-03168-7_14.
26. Turk M, Pentland A. Eigenfaces for recognition. *J. Cognitive Neuroscience* Jan 1991; **3**(1):71–86, doi:10.1162/jocn.1991.3.1.71.
27. Damgård I, Geisler M, Krøigaard M. Efficient and secure comparison for on-line auctions. *Information Security and Privacy*, *Lecture Notes in Computer Science*, vol. 4586. Springer Berlin Heidelberg, 2007; 416–430, doi:10.1007/978-3-540-73458-1_30.
28. Damgård I, Geisler M, Kroigard M. A correction to efficient and secure comparison for on line auctions. *Int. J. Appl. Cryptol.* Aug 2009; **1**(4):323–324, doi:10.1504/IJACT.2009.028031.
29. Sadeghi AR, Schneider T, Wehrenberg I. Efficient privacy-preserving face recognition. *Proceedings of the 12th International Conference on Information Security and Cryptology*, ICISC'09, Springer-Verlag, 2010; 229–244.
30. Schneider T, Zohner M. GMW vs. yao? efficient secure two-party computation with low depth circuits. *Financial Cryptography and Data Security - 17th International Conference, FC 2013, Okinawa, Japan, April 1-5, 2013, Revised Selected Papers*, *Lecture Notes in Computer Science*, vol. 7859, Sadeghi A (ed.), Springer, 2013; 275–292.

31. Goldreich O, Micali S, Wigderson A. How to play any mental game. *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, ACM: New York, NY, USA, 1987; 218–229.
32. Tuyls P, Akkermans AHM, Kevenaar TAM, Schrijen GJ, Bazen AM, Veldhuis RNJ. Practical biometric authentication with template protection. *Audio- and Video-Based Biometric Person Authentication*, *Lecture Notes in Computer Science*, vol. 3546. Springer Berlin Heidelberg, 2005; 436–446, doi:10.1007/11527923_45.
33. Tuyls P, Goseling J. Capacity and examples of template-protecting biometric authentication systems. *Biometric Authentication*, *Lecture Notes in Computer Science*, vol. 3087. Springer Berlin Heidelberg, 2004; 158–170, doi:10.1007/978-3-540-25976-3_15.
34. Linnartz JP, Tuyls P. New shielding functions to enhance privacy and prevent misuse of biometric templates. *Audio- and Video-Based Biometric Person Authentication*, *Lecture Notes in Computer Science*, vol. 2688. Springer Berlin Heidelberg, 2003; 393–402, doi:10.1007/3-540-44887-X_47.
35. Kulkarni R, Namboodiri A. Secure hamming distance based biometric authentication. *Biometrics (ICB), 2013 International Conference on Biometrics Compendium, IEEE*, 2013; 1–6, doi:10.1109/ICB.2013.6613008.
36. Boneh D, Goh EJ, Nissim K. Evaluating 2-dnf formulas on ciphertexts. *Theory of Cryptography*, *Lecture Notes in Computer Science*, vol. 3378. Springer Berlin Heidelberg, 2005; 325–341, doi:10.1007/978-3-540-30576-7_18.
37. Kerschbaum F, Atallah MJ, M'Raïhi D, Rice JR. Private fingerprint verification without local storage. *Biometric Authentication*, *Lecture Notes in Computer Science*, vol. 3072. Springer Berlin Heidelberg, 2004; 387–394, doi:10.1007/978-3-540-25948-0_54.
38. Barni M, Bianchi T, Catalano D, Di Raimondo M, Labati RD, Failla P, Fiore D, Lazzeretti R, Piuri V, Piva A, *et al.*. A privacy-compliant fingerprint recognition system based on homomorphic encryption and fingercode templates. *Biometrics: Theory Applications and Systems (BTAS), 2010 Fourth IEEE International Conference on*, 2010; 1–7, doi:10.1109/BTAS.2010.5634527.
39. Barni M, Bianchi T, Catalano D, Di Raimondo M, Donida Labati R, Failla P, Fiore D, Lazzeretti R, Piuri V, Scotti F, *et al.*. Privacy-preserving fingercode authentication. *Proceedings of the 12th ACM Workshop on Multimedia and Security*, MM&Sec '10, ACM, 2010; 231–240, doi:10.1145/1854229.1854270.
40. Canetti R. Universally composable security: a new paradigm for cryptographic protocols. *Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on*, 2001; 136–145, doi:10.1109/SFCS.2001.959888.
41. Aumann Y, Lindell Y. Security against covert adversaries: Efficient protocols for realistic adversaries. *Theory of Cryptography*, *Lecture Notes in Computer Science*, vol. 4392. Springer Berlin Heidelberg, 2007; 137–156, doi:10.1007/978-3-540-70936-7_8.
42. Ishai Y, Kilian J, Nissim K, Petrank E. Extending oblivious transfers efficiently. *Advances in Cryptology - CRYPTO 2003*, *Lecture Notes in Computer Science*, vol. 2729. Springer Berlin Heidelberg, 2003; 145–161, doi:10.1007/978-3-540-45146-4_9.
43. Kolesnikov V, Kumaresan R. Improved ot extension for transferring short secrets. *Advances in Cryptology – CRYPTO 2013*, *Lecture Notes in Computer Science*, vol. 8043. Springer Berlin Heidelberg, 2013; 54–70, doi:10.1007/978-3-642-40084-1_4.
44. Elgamal T. A public key cryptosystem and a signature scheme based on discrete logarithms. *Information Theory, IEEE Transactions on* Jul 1985; **31**(4):469–472, doi:10.1109/TIT.1985.1057074.
45. Paillier P. Public-key cryptosystems based on composite degree residuosity classes. *Advances in Cryptology — EUROCRYPT '99*, *Lecture Notes in Computer Science*, vol. 1592. Springer Berlin Heidelberg, 1999; 223–238, doi:10.1007/3-540-48910-X_16.
46. Luo Y, Cheung ScS, Pignata T, Lazzeretti R, Barni M. An efficient protocol for private iris-code matching by means of garbled circuits. *19th IEEE International Conference on Image Processing (ICIP'12)* 2012; :2653–2656.
47. Baig A, Bouridane A, Kurugollu F, Qu G. Fingerprint - iris fusion based identification system using a single hamming distance matcher. *Bio-inspired Learning and Intelligent Systems for Security, 2009. BLISS '09.*, 2009; 9–12, doi:10.1109/BLISS.2009.4.
48. Kiraz MS. Secure and fair two-party computation. PhD Thesis, Technische Universiteit Eindhoven, the Netherlands 2008.