MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Information-Theoretically Secure Three-Party Computation with One Corrupted Party

Wang, Y.; Ishwar, P.; Rane, S.

TR2013-077 July 2013

Abstract

The problem in which one of three pairwise interacting parties is required to securely compute a function of the inputs held by the other two, when one party may arbitrarily deviate from the computation protocol (active behavioral model), is studied. An information-theoretic characterization of unconditionally secure computation protocols under the active behavioral model is provided. A protocol for Hamming distance computation is provided and shown to be unconditionally secure under both active and passive behavioral models using the information theoretic characterization. The difference between the notions of security under the active and passive behavioral models is illustrated by examining a protocol for computing quadratic and Hamming distances that is secure under the passive model, but is insecure under the active model.

IEEE International Symposium on Information Theory (ISIT)

Information-Theoretically Secure Three-Party Computation with One Corrupted Party

Ye Wang
Mitsubishi Electric Research Laboratories
Cambridge, MA, USA
Email: yewang@merl.com

Prakash Ishwar
Dep. of Electrical & Computer Eng.
Boston University, Boston, USA
Email: pi@bu.edu

Shantanu Rane
Mitsubishi Electric Research Laboratories
Cambridge, MA, USA
Email: rane@merl.com

***Abstract*—The problem in which one of three pairwise interacting parties is required to securely compute a function of the inputs held by the other two, when one party may arbitrarily deviate from the computation protocol (active behavioral model), is studied. An information-theoretic characterization of unconditionally secure computation protocols under the active behavioral model is provided. A protocol for Hamming distance computation is provided and shown to be unconditionally secure under both active and passive behavioral models using the information-theoretic characterization. The difference between the notions of security under the active and passive behavioral models is illustrated by examining a protocol for computing quadratic and Hamming distances that is secure under the passive model, but is insecure under the active model.**

I. Introduction

The subject of secure multiparty computation (SMC) is concerned with the design and analysis of distributed protocols that allow a mutually untrusting group to securely compute functions of their private inputs while not revealing any more information than must be inherently revealed by the computation itself. In this broad domain (see [1] for a detailed overview) one can consider computational or unconditional (information-theoretic) definitions of security, active or passive behavioral models, and the utilization of additional communication primitives, e.g., shared randomness via multi-terminal sources and/or channels. In this paper, we study secure computation involving three parties that can communicate via pairwise authenticated and error-free bitpipes where one party is required to compute a function of the inputs held by the other two. Our focus is on unconditional security and the active behavioral model with up to one party arbitrarily deviating from the protocol.

The scenario of three-party computation with one actively deviating party is interesting since no security guarantees are available in this scenario for the general SMC protocols of [2], [3]. For the active behavioral model with only pairwise communication, the protocols of [2], [3] are secure only if *strictly* less than a third of the parties are compromised. Thus, non-trivial security guarantees are only available for a minimum of four parties. Conversely, certain computations, such as Byzantine agreement [4], are provably impossible in a three-party setting. However, other non-trivial computations may be possible, but require specialized techniques. A characterization of all functions that can be securely computed in a three-party setting with one actively deviating party is currently unavailable.

The formulation of security in the active behavioral model requires careful consideration of the notions of correctness and privacy since a party may arbitrarily deviate from the protocol. A deviating party can always affect the integrity of the computation by simply changing its input data. This, however, should not be considered a security weakness since such an attack could also be mounted against a "trusted genie" who can receive all inputs, perform all computations, and deliver the results to the designated parties. A deviating party's ability to influence the computation or affect the privacy should, ideally, not exceed what could be done against such a trusted genie. Therefore, in the active behavioral model, a protocol is said to be secure if it adequately *simulates* a trusted genie that facilitates the computation. This is formalized in the literature as the real versus ideal model simulation paradigm for SMC (see [5]). The passive behavioral model, in contrast, assumes that all parties will adhere to the protocol, but may attempt to infer additional information from the "view" available to them from the protocol. To assess the security of a protocol in the passive behavioral model, one only needs to check that the protocol correctly computes the function while revealing no more information than what can be inherently inferred from the result of the computation.

In our three-party problem setup, Alice has input X, Bob has input Y, and Charlie wants to compute the function $f(X, Y)$. In Section II, we define security based on the real versus ideal model simulation paradigm and develop an equivalent information-theoretic characterization that generalizes conditions developed for two parties in [6]. In Section III, we present a simple finite-field arithmetic-based protocol for computing Hamming distance and show that it is unconditionally secure under both active and passive behavioral models using the information-theoretic characterization. In Section IV, we illustrate the difference between the notions of security under active and passive behavioral models by constructing a protocol with the techniques of [2] for computing the quadratic and Hamming distances. This protocol is designed for and

The second author acknowledges support from the US NSF under award number #0915389 and MERL. The views and conclusions contained in this article are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the US NSF or MERL.

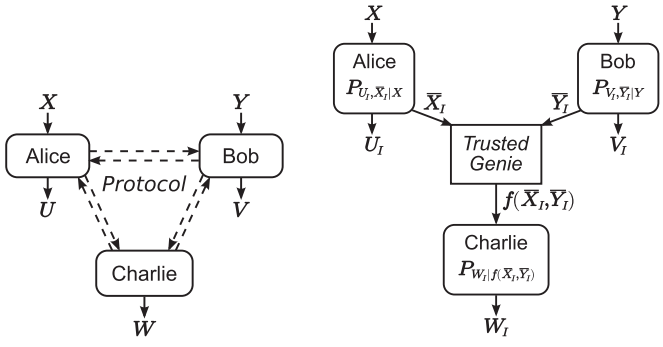
secure under the passive behavioral model, but is shown to be insecure under the active behavioral model.

II. Information-Theoretic Security Conditions

We first define security for the active behavioral model, then derive equivalent information-theoretic conditions, and finally present information-theoretic conditions for the passive behavioral model. For convenience, our development is suited to the specific case where only Alice and Bob have inputs and Charlie computes an output. However, one could also generalize this development to a scenario with all parties contributing an input and computing an output.

A. Real versus Ideal Model Simulation Paradigm

A protocol Π for three-party computation is a triple of algorithms (A, B, C) that are intended to be executed by Alice, Bob, and Charlie, respectively. These algorithms may include instructions for processing inputs (X for Alice and Y for Bob), generating local randomness, performing intermediate local computations, sending messages to and receiving/processing messages from other parties, and producing local outputs. The outputs produced by Alice, Bob, and Charlie will be denoted by U, V, and W, respectively. A protocol Π is the "real model" for three-party computation (cf. Figure 1(a)).



(a) Real Model (b) Ideal Model

Fig. 1. A protocol is secure if any attack against it in the real model (a) can be equivalently mounted against the trusted genie in the ideal model (b).

In the "ideal model" for three-party computation, there is an additional fourth party: a trusted genie that facilitates the computation (cf. Figure 1(b)). An ideal model protocol $\overline{\Pi}_I$ is a triple of algorithms $(\overline{A}_I, \overline{B}_I, \overline{C}_I)$ that have a very specific structure: Alice's algorithm $\overline{A}_I$ consists solely of an independent random functionality that takes as an input only X and outputs U_I and $\overline{X}_I$, and can be modeled as a conditional distribution $P_{U_I,\overline{X}_I|X}$. Likewise, Bob's algorithm $\overline{B}_I$ is an independent random functionality that takes as an input only Y and outputs V_I and $\overline{Y}_I$, and can be modeled as a conditional distribution $P_{V_I,\overline{Y}_I|Y}$. The random variables $\overline{X}_I$ and $\overline{Y}_I$ represent the inputs that Alice and Bob give to the trusted genie, and U_I and V_I respectively represent Alice and Bob's outputs. The trusted genie receives $(\overline{X}_I, \overline{Y}_I)$ from Alice and Bob, computes $f(\overline{X}_I, \overline{Y}_I)$ and sends this to Charlie. If either Alice or Bob refuse to send their input to the trusted genie or send an invalid input, e.g., inputs not belonging to the proper alphabets $\mathcal{X}$ or $\mathcal{Y}$, then the genie assumes a valid default input. Charlie's algorithm $\overline{C}_I$ is a random functionality that takes $f(\overline{X}_I, \overline{Y}_I)$ as input and produces W_I as output, and can be modeled as a conditional distribution $P_{W_I|f(\overline{X}_I,\overline{Y}_I)}$.

Definition 1 (Honest Ideal Model Protocol): The ideal model protocol $\Pi_I = (A_I, B_I, C_I)$ is called "honest" if $U_I = V_I = \emptyset, \overline{X}_I = X, \overline{Y}_I = Y, W_I = f(\overline{X}_I, \overline{Y}_I) = f(X, Y)$.

In our problem, at most one party may actively deviate from the protocol, and no collusions form between any parties. This motivates the following definition that captures the active behavioral model of interest.

Definition 2 (Admissible Deviation): A protocol $\overline{\Pi} = (\overline{A}, \overline{B}, \overline{C})$ is an admissible deviation of $\Pi = (A, B, C)$ if at most one of $(\overline{A}, \overline{B}, \overline{C})$ differs from (A, B, C).

In the real versus ideal model simulation paradigm, a real model protocol is considered to be secure if it can be shown that for every attack against the protocol – captured through the above notion of an admissible deviation of a protocol – a statistically equivalent attack can be mounted against the honest ideal model protocol in the ideal model. The following definition makes this notion precise.

Definition 3 (Security Against Active Behavior): A three-party protocol $\Pi = (A, B, C)$ securely computes $f(X, Y)$ under the active behavioral model if, for every real model protocol $\overline{\Pi} = (\overline{A}, \overline{B}, \overline{C})$ that is an admissible deviation of Π and for any distribution $P_{X,Y}$ on inputs $(X, Y) \sim P_{X,Y}$, there exists an ideal model protocol $\overline{\Pi}_I = (\overline{A}_I, \overline{B}_I, \overline{C}_I)$ that is an admissible deviation of the honest ideal model protocol Π_I, where the same players are honest, such that

$$P_{U,V,W|X,Y} = P_{U_I,V_I,W_I|X,Y}, \tag{1}$$

where (U, V, W) are the outputs of the protocol $\overline{\Pi}$ in the real model with inputs (X, Y) and (U_I, V_I, W_I) are the outputs of the protocol $\overline{\Pi}_I$ in the ideal model with inputs (X, Y).

Contained within the above definition of security is the requirement that a secure protocol must ensure that Charlie will correctly compute the function if none of the parties deviate from the protocol. This is because no deviation is an admissible deviation and corresponds to the honest ideal model protocol which results in correct computation of the function. Privacy requirements against a deviating party are also contained within this security definition since the deviating party may include arbitrary additional information in its output. The above security definition precludes this additional output information from containing any information that could not be obtained by the party deviating in the ideal model. This definition provides perfect security, however one could weaken the definition with the equality of (1) replaced by an "ϵ-closeness" requirement, as done in [7] for two parties.

B. Security Conditions for the Active Behavioral Model

The following theorem describes information-theoretic conditions that are equivalent to the security conditions given by Definition 3. These conditions provide an alternative way to test whether a given protocol is secure under the active behavioral model directly in the real model without explicit

reference to an ideal model or a trusted genie. In contrast, Definition 3 needs to refer to an ideal model.

Theorem 1: A real-model three-party protocol $\Pi = (A, B, C)$ securely computes $f(X, Y)$ under the active behavioral model if, and only if, for every real model protocol $\overline{\Pi} = (\overline{A}, \overline{B}, \overline{C})$ that is an admissible deviation of Π, and for any distribution $P_{X,Y}$ on inputs (X, Y) the algorithms $(\overline{A}, \overline{B}, \overline{C})$ respectively produce outputs (U, V, W), such that the following conditions are satisfied:

- *(Correctness)* If $\overline{\Pi} = \Pi$, then

$$\Pr[(U, V, W) = (\emptyset, \emptyset, f(X, Y))] = 1. \quad (2)$$

- *(Security against Alice)* If $(B, C) = (\overline{B}, \overline{C})$, then $\exists\ \overline{X}$:

$$I(U, \overline{X}; Y|X) = 0, \quad (3)$$
$$\Pr[(V, W) = (\emptyset, f(\overline{X}, Y))] = 1. \quad (4)$$

- *(Security against Bob)* If $(A, C) = (\overline{A}, \overline{C})$, then $\exists\ \overline{Y}$:

$$I(V, \overline{Y}; X|Y) = 0, \quad (5)$$
$$\Pr[(U, W) = (\emptyset, f(X, \overline{Y}))] = 1. \quad (6)$$

- *(Security against Charlie)* If $(A, B) = (\overline{A}, \overline{B})$ then

$$I(W; X, Y|f(X, Y)) = 0, \quad (7)$$
$$\Pr[(U, V) = (\emptyset, \emptyset)] = 1. \quad (8)$$

Proof: We sketch the proof here and refer to [8] for details.

To show that the conditions are necessary, we must show that if the protocol Π securely computes $f(X, Y)$, then the information-theoretic conditions must hold. Since the protocol Π is secure, for any admissible deviation $\overline{\Pi}$, there must exist an ideal model protocol $\overline{\Pi}_I$ (that is an admissible deviation from the honest ideal model protocol Π_I, where the same players are honest), such that (1) holds. Due to the structure of the ideal model protocols, we can easily verify that the information-theoretic conditions (2-8) hold when $(\overline{X}, \overline{Y}, U, V, W)$ are replaced with $(\overline{X}_I, \overline{Y}_I, U_I, V_I, W_I)$. Since (1) holds, it follows that (2), (4), (6), and (8) also hold. Further, due to (1), since $\overline{X}_I$ and $\overline{Y}_I$ exist with respect to $P_{U_I,V_I,W_I|X,Y}$, there must exist $\overline{X}$ and $\overline{Y}$ with respect to $P_{U,V,W|X,Y}$ such that (3), (5), and (7) hold.

For sufficiency we will show that if the conditions are satisfied, then the protocol Π securely computes $f(X, Y)$. We will show this by constructing, for any admissible deviation $\overline{\Pi}$ of Π, a corresponding ideal model protocol $\overline{\Pi}_I$ (that is an admissible deviation from the honest ideal model protocol Π_I where the same players are honest) that satisfies (1). For the case $\overline{\Pi} = \Pi$ (all players are honest) we have $\overline{\Pi}_I = \Pi_I$. Then by (2) and the definition of Π_I, condition (1) holds. For the case $\overline{A} \neq A$ (Alice is dishonest) we can choose her corresponding ideal model algorithm $\overline{A}_I$ to be defined by $P_{U_I,\overline{X}_I|X} := P_{U,\overline{X}|X}$, where $\overline{X}$ is the random variable found to satisfy conditions (3) and (4). From this it follows that condition (1) holds. The reasoning is symmetric for the case where Bob is dishonest. For the case $\overline{C} \neq C$ (Charlie is dishonest), we can choose the corresponding ideal model algorithm $\overline{C}_I$ to be defined by $P_{W_I|f(\overline{X}_I,\overline{Y}_I)} := P_{W|f(X,Y)}$, which, by (7) and (8), leads to condition (1) holding. ■

C. Security Conditions for the Passive Behavioral Model

In the passive behavioral model, all parties correctly follow the protocol, but may still attempt to learn as much new information as they can from the messages that they receive from other parties during the execution of the protocol. A protocol is secure against passive behavior if it produces correct computation results and reveals no more information to any party than what can be inherently inferred from their own input or computation result. Thus, security against passive behavior is a statement about the correctness and the information leakage properties of a protocol. We directly state the information-theoretic conditions for security under the passive behavioral model, which one can similarly derive from a real versus ideal model definition.

Definition 4 (Security Against Passive Behavior): A three-party protocol $\Pi = (A, B, C)$ securely computes $f(X, Y)$ under the passive behavioral model (with no collusions) if after Alice, Bob, and Charlie execute the protocol, the following conditions are satisfied:

- *(Correctness)* $\Pr[(U, V, W) = (\emptyset, \emptyset, f(X, Y))] = 1$.
- *(Privacy against Alice)* $I(M_1; Y, f(X, Y)|X) = 0$, where M_1 denotes the "view" of Alice, consisting of all the local randomness generated, local computations performed, and messages sent and received by Alice.
- *(Privacy against Bob)* $I(M_2; X, f(X, Y)|Y) = 0$, where M_2 denotes the view of Bob.
- *(Privacy against Charlie)* $I(M_3; X, Y|f(X, Y)) = 0$, where M_3 denotes the view of Charlie.

In general, security of a protocol under the active behavioral model *does not* necessarily imply security of a protocol under the passive behavioral model [9]. This may seem counterintuitive at first since possible attacks by active parties are surely expected to subsume the possible "passive attacks". This can be resolved by observing that the definition of security under the active behavioral model compares admissible deviations (active attacks) in the real model to possible active attacks in the ideal model. This comparison to a benchmark involving active attacks in the ideal model potentially results in more permissive privacy conditions than the information leakage conditions required in the passive behavioral model. To illustrate this difference, consider the following two-party example (from [9]): Alice and Bob each have a bit and Bob wishes to compute the Boolean AND of the bits, while Alice computes nothing. A protocol where Alice simply gives Bob her bit and he computes his desired function is clearly insecure under the passive behavioral model since Alice directly reveals her bit, whereas the AND function should only reveal her bit if Bob's bit is one. However, this protocol would be considered secure in the active behavioral model since a deviating Bob could change his input to one to always reveal the value of Alice's bit from the trusted genie, and thus the apparent insecurity is inherent to the computation.

III. A Secure Protocol for Hamming Distance

We now present and analyze a simple finite-field arithmetic-based protocol **HamDist** that securely computes the Hamming

distance for finite-field sequences under both passive and active behavioral models. The security of this protocol will be proved using the information-theoretic conditions for security under (i) the active behavioral model (Theorem 1) and (ii) the passive behavioral model (Definition 4). We assume that Alice and Bob have finite-field sequences $X := X^n$ and $Y := Y^n$, respectively, with $X^n, Y^n \in \mathcal{F}_{p^k}^n$, where $\mathcal{F}_{p^k}$ is the finite-field of prime-power order p^k. Charlie wishes to compute the Hamming distance $f(X^n, Y^n) := \sum_{i=1}^n \mathbf{1}_{\{X_i\}}(Y_i)$. Protocol **HamDist** proceeds as follows:

1) Alice randomly chooses two independent sequences $R^n, Z^n \in \mathcal{F}_{p^k}^n$, where R^n is uniform over all sequences and Z^n is uniform over $(\mathcal{F}_{p^k} \setminus \{0\})^n$. Alice also randomly chooses a permutation π of $\{1, \ldots, n\}$, uniformly and independently of (X^n, Y^n, R^n, Z^n).

2) Alice sends R^n, Z^n and π to Bob.

3) Alice sends $A^n := \pi(Z^n \otimes (X^n \ominus R^n))$ to Charlie, where $\ominus$ and $\otimes$ respectively denote element-wise field subtraction and multiplication, and $\pi(\cdot)$ denotes sequence permutation via π.

4) Bob sends $B^n := \pi(Z^n \otimes (R^n \ominus Y^n))$ to Charlie.

5) Charlie combines the messages from Alice and Bob, via element-wise field addition, and outputs the Hamming weight of the sequence $(A^n \oplus B^n)$.

During the execution of the protocol, if any party fails to send a message or sends an invalid message to another party, a valid default message is assumed by the receiving party. Also, any extraneous messages are simply ignored. For example, in step two, Bob expects to receive two sequences and a permutation from Alice. If Alice omits or sends invalid messages (e.g., R^n or Z^n are not finite-field sequences of the appropriate length, Z^n contains a zero, π is not a valid permutation), Bob would interpret an invalid or missing sequence as, for instance, an all-one sequence, and an invalid or missing permutation as the identity permutation. The specific default message assumed in the case of invalid or missing messages is unimportant and could be replaced by any other valid fixed message.

Theorem 2: Protocol **HamDist** is secure under the active behavioral model.

Proof: We sketch the proof here and refer to [8] for details.

(Correctness) When all parties follow the protocol, Charlie computes $A^n \oplus B^n = \pi(Z^n \otimes (X^n \ominus Y^n))$ which has Hamming weight equal to the Hamming distance between X^n and Y^n.

Since any invalid or missing messages are interpreted by the receiver as some default message, we can assume, without loss of generality, that the arbitrarily modified algorithms send well-formed messages in the prescribed message alphabets.

(Security Against Alice) Since Alice only sends messages, the only attack she can mount is to independently change the sequences and permutation that she sends to Bob and Charlie. However, any changes to those messages effectively just changes her input to $\overline{X}^n = \overline{R}^n \oplus (\overline{\pi}^{-1}(\overline{A}^n) \oslash \overline{Z}^n)$, where $(\overline{R}^n, \overline{Z}^n)$ and $\overline{A}^n$ are the modified sequences sent to Bob and Charlie respectively, $\overline{\pi}^{-1}(\cdot)$ denotes the inverse application of the modified permutation $\overline{\pi}$, and $\oslash$ denotes element-wise field division. The random variable $\overline{X}^n$ can be shown to satisfy the conditions corresponding to security against Alice given by (3) and (4).

(Security Against Bob) Bob receives the random sequences (R^n, Z^n) and permutation π from Alice, which are independent of her input X^n. The only attack that Bob can mount is to change the sequence that he sends to Charlie. However this effectively just changes his input to $\overline{Y}^n = R^n \ominus (\pi^{-1}(\overline{B}^n) \oslash Z^n)$, where $\overline{B}^n$ is the modified sequence that he sends to Charlie. The random variable $\overline{Y}^n$ can be shown to satisfy the conditions corresponding to security against Bob given by (5) and (6).

(Security Against Charlie) Since Charlie only receives messages, the only attack that he can mount is to attempt to infer and output additional information about X^n and Y^n that is not already revealed by the Hamming distance. However, the messages that he receives, A^n from Alice and B^n from Bob, are only sufficient to reveal $A^n \oplus B^n = \pi(Z^n \otimes (X^n \ominus Y^n))$, which reveals no more information about X^n and Y^n than the Hamming distance, since the multiplication of each $(X_i - Y_i)$ by an independent, uniform, and non-zero Z_i conceals the difference, only revealing whether X_i is equal to Y_i. Further, the random permutation conceals the locations where the two sequences differ, preserving only the count. ■

As previously discussed, security of a protocol under the active behavioral model does not necessarily imply security of a protocol under the passive behavioral model [9]. We, however, have the following result.

Theorem 3: Protocol **HamDist** is secure under the passive behavioral model.

Proof: We sketch the proof here and refer to [8] for details.

(Correctness) The protocol is correct by the same argument as for the active behavioral model. *(Privacy against Alice)* The protocol is private against Alice since she does not receive any information from other parties. *(Privacy against Bob)* The protocol is private against Bob since the only message from Alice that he receives is independent of X^n, Y^n, W. *(Privacy against Charlie)* The protocol is private against Charlie by the same argument as for the active behavioral model. ■

IV. Inadequacy of BGW for Quadratic Distance

Under the passive behavioral model (with no collusions), any function can be securely computed amongst three parties using the secure computation methods of [2] that are based on homomorphic polynomial secret sharing [10] and popularly called the **BGW** protocol. However, for three parties, the techniques proposed in [2] for active adversaries require a minimum of four parties. To illustrate the differing notions of security between the active and passive models, we consider the **BGW** protocol for three-party quadratic and Hamming distance computation under the *passive* behavioral model and demonstrate how it is insecure under the *active* behavioral model. The question as to whether there exist protocols that securely compute the quadratic distance under the active behavioral model remains open.

We assume that Alice and Bob respectively have integer sequences $X^n, Y^n \in \mathbb{Z}_s^n$, where $\mathbb{Z}_s := \{0, 1, \ldots, s-1\}$. We embed the set $\mathbb{Z}_s$ in a finite-field $\mathbb{Z}_N$ of prime order $N > n(s-1)^2$ with modulo-N field arithmetic. This ensures that $\mathbb{Z}_N$ is

large enough to simulate the necessary integer arithmetic for computing the quadratic distance $f(X^n, Y^n) = \sum_{i=1}^n (X_i - Y_i)^2$ while avoiding overflow (modulo) effects. Protocol **BGW** for computing the quadratic distance proceeds as follows:

1) Alice randomly chooses $\alpha_1, \ldots, \alpha_n \sim$ iid Unif$(\mathbb{Z}_N)$ independently of (X^n, Y^n). For each $i \in \{1, \ldots, n\}$, Alice creates a polynomial $p_i : \mathbb{Z}_N \to \mathbb{Z}_N$, via $p_i(j) := \alpha_i j + X_i$. Alice sends Bob (party $j = 2$) the values $(p_1(2), \ldots, p_n(2))$, and Charlie (party $j = 3$) the values $(p_1(3), \ldots, p_n(3))$, while retaining $(p_1(1), \ldots, p_n(1))$ for herself (party $j = 1$).

2) Similarly, Bob randomly chooses $\beta_1, \ldots, \beta_n \sim$ iid Unif$(\mathbb{Z}_N)$ independently of (X^n, Y^n), and creates polynomials $q_i(j) := \beta_i j + Y_i$. Bob sends Alice the values $(q_1(1), \ldots, q_n(1))$, and Charlie the values $(q_1(3), \ldots, q_n(3))$, while retaining $(q_1(2), \ldots, q_n(2))$.

3) Alice, Bob, and Charlie each individually compute samples of the polynomial $r : \mathbb{Z}_N \to \mathbb{Z}_N$ defined by $r(j) := \sum_{i=1}^n \left[p_i^2(j) + q_i^2(j) - 2p_i(j)q_i(j)\right]$. Specifically, Alice computes $r(1)$ using $\{p_i(1), q_i(1)\}_{i=1}^n$. Likewise, Bob and Charlie compute $r(2)$ and $r(3)$, respectively.

4) Alice and Bob send $r(1)$ and $r(2)$, respectively, to Charlie.

5) Charlie reconstructs the degree-2 polynomial r via interpolation from $r(1)$, $r(2)$, and $r(3)$. Finally, he obtains:

$$\begin{aligned} r(0) &= \sum_{i=1}^n \left[p_i^2(0) + q_i^2(0) - 2p_i(0)q_i(0)\right] \\ &= \sum_{i=1}^n \left[X_i^2 + Y_i^2 - 2X_iY_i\right] = f(X^n, Y^n). \end{aligned}$$

Since quadratic distance coincides with Hamming distance for binary sequences ($s = 2$), the above protocol can also be used to compute the Hamming distance for binary sequences.

Proposition 1: For quadratic and Hamming distance computation, the **BGW** protocol is secure under the passive behavioral model, but not under the active behavioral model.

Proof: The security of this protocol under the passive behavioral model is well-known (see [11] for a rigorous proof) and one can confirm that it satisfies the conditions of Definition 4. To show insecurity under the active behavioral model, it is sufficient to describe an attack that is able to influence the computation beyond what can be achieved against a trusted genie. For this, we demonstrate examples for both the quadratic and Hamming distance below.

Quadratic Distance ($s > 2$): The range $\mathcal{R}(f)$ of the quadratic distance, is a proper subset of $\mathbb{Z}_{n(s-1)^2}$ since each function value is a sum of n numbers from the set $\{x^2 : x \in \mathbb{Z}_s\}$. The finite-field $\mathbb{Z}_N$ must have prime size $N > n(s-1)^2$ in order to simulate integer arithmetic as finite-field arithmetic. Hence, $\mathcal{R}(f) \subsetneq \mathbb{Z}_N$, whereas $\mathbb{Z}_N \setminus \mathcal{R}(f)$ contains invalid outputs for the function computation. In the ideal model, for any attack by Alice (or symmetrically by Bob), the output of Charlie would still remain in $\mathcal{R}(f)$, since Alice can only affect it by changing her input. However, in the real model, Alice can launch a simple attack, where she randomly chooses the final message $r(1)$ sent to Charlie independently and uniformly over $\mathbb{Z}_N$. This causes Charlie's output to uniformly take values over $\mathbb{Z}_N$, including invalid values, due to the polynomial interpolation in computing his output. For fixed $r(2)$ and $r(3)$, each modified value of $r(1)$ corresponds to a unique interpolation result, since 3 samples uniquely determine a degree-2 polynomial. Due to this one-to-one relationship, a uniform distribution on $r(1)$ induces a uniform distribution on the computation result. Thus, the protocol is insecure as there exists an attack in the real model (against the protocol) that cannot be equivalently mounted in the ideal model. In addition to creating the possibility of an invalid output, the attack also makes the distribution of valid outputs uniform, which cannot occur in an attack against a trusted genie.

Hamming Distance ($s = 2$): Suppose that Alice and Bob have independent sequences of iid Bernoulli$(1/2)$ bits. In the ideal model, for any attack by Alice (or symmetrically by Bob), the exclusive-OR of her string and Bob's is an iid Bernoulli$(1/2)$ sequence since his string is iid Bernoulli$(1/2)$ and independent of Alice's modified input. This means that for any attack by Alice against a trusted genie, Charlie's output is always distributed over $\{0, 1, \ldots, n\}$ as a binomial distribution with mean $n/2$. For the protocol in the real model, if $N = n+1$ is prime, then $\mathbb{Z}_N$ can be used without containing any invalid outputs. However, Alice could launch a simple attack by randomly choosing the final message $r(1)$ sent to Charlie uniformly over $\mathbb{Z}_N$, causing Charlie's output to be uniformly distributed over $\{0, 1, \ldots, n\}$. Thus, the protocol is insecure since there exists an attack in the real model that influences the output in a manner that cannot be replicated by an attack against a trusted genie. ■

REFERENCES

[1] R. Cramer and I. Damgård, "Multiparty computation, an introduction," in *Contemporary Cryptology*, ser. Advanced Courses in Mathematics – CRM Barcelona. Birkhäuser Basel, 2005, pp. 41–87.

[2] M. Ben-Or, S. Goldwasser, and A. Wigderson, "Completeness theorems for non-cryptographic fault-tolerant distributed computation," in *Proceedings of the ACM Symposium on Theory of Computing*, Chicago, IL, 1988, pp. 1–10.

[3] D. Chaum, C. Crépeau, and I. Damgård, "Multi-party unconditionally secure protocols," in *Proceedings of the ACM Symposium on Theory of Computing*, Chicago, IL, 1988, pp. 11–19.

[4] M. Pease, R. Shostak, and L. Lamport, "Reaching agreement in the presence of faults," *Journal of the ACM*, vol. 27, no. 2, pp. 228–234, Apr. 1980.

[5] O. Goldreich, *Foundations of Cryptography*. Cambridge University Press, 2004, vol. II: Basic Applications.

[6] C. Crépeau, G. Savvides, C. Schaffner, and J. Wullschleger, "Information-theoretic conditions for two-party secure function evaluation," in *Advances in Cryptology – EUROCRYPT*, ser. Lecture Notes in Computer Science, vol. 4004. Springer-Verlag, 2006, pp. 538–554.

[7] C. Crépeau and J. Wullschleger, "Statistical security conditions for two-party secure function evaluation," in *Proceedings of the 3rd International Conference on Information Theoretic Security*, ser. Lecture Notes in Computer Science, vol. 5155. Springer-Verlag, 2008, pp. 86–99.

[8] Y. Wang, P. Ishwar, and S. Rane, "Information-theoretically secure three-party computation with one corrupt party," arXiv ePrint Archive, 2012, http://arxiv.org/abs/1206.2669.

[9] J. Wullschleger, "Oblivious-transfer amplification," Ph.D. dissertation, Swiss Federal Institute of Technology, Zürich, 2008.

[10] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 637–647, 1985.

[11] G. Asharov and Y. Lindell, "A full proof of the BGW protocol for perfectly-secure multiparty computation," Cryptology ePrint Archive, 2011, http://eprint.iacr.org/2011/136.