Privacy Preserving Cyber Threat Information Sharing and Learning for Cyber Defense

Shahriar Badsha
Department of Computer Science and Engineering
University of Nevada, Reno
Reno, NV, USA
sbadsha@unr.edu

Iman Vakilinia
Department of Computer Science and Engineering
University of Nevada, Reno
Reno, NV, USA
ivakilinia@unr.edu

Shamik Sengupta
Department of Computer Science and Engineering
University of Nevada, Reno
Reno, NV, USA
ssengupta@unr.edu

***Abstract*—To secure cyber infrastructure against intentional and potentially malicious threats, a growing collaborative effort between cybersecurity professionals and researchers from institutions, private industries, academia, and government agencies has engaged in exploiting and designing a variety of cyber defense systems. Cybersecurity researchers and designers aim to maintain the confidentiality, integrity, and availability of information and information management systems through various cyber defense systems that protect computers and networks from hackers who may want to steal financial, medical, or other identity-based information. The *Cooperative Cyber-defense* has been recognized as an essential strategy to fight against cyberattacks. Cybersecurity information sharing among various organizations and leveraging the aggregated cyber information to build proactive cyber defense system is nontrivial for organizations. However, building such cyber defense system is challenged by two issues: (1) organizations are reluctant to share their private information to others (2) even when they agree on a solution where information can be shared in privacy preserving manner, the obfuscated cyber threat information has to be processed to build the trained model for future prediction of any new or unknown cyber incident. To address these issues, in this paper, we propose a privacy preserving protocol where organizations can share their private information as an encrypted form with others and they can learn the information for future prediction without disclosing any private information. More specifically we propose a privacy preserving decision tree algorithm, where each organization can build and learn the decision tree based on overall organizations' training spam/ham email data without disclosing any private information of any party. Once the building of a decision tree is done, the organizations can predict if any new email is spam or ham locally.**

Keywords- Privacy, Cyber Threat Information Sharing, Machine Learning, Prediction

I. Introduction

Conventional vs. Cooperative Cyber-defense: The revolution of Information and Communication Technologies (ICT) has brought economic prosperity in recent years. However, securing the cyberspace from malicious attackers has been a critical concern. Due to increasing rate of cyber crimes and complexity of cyber-threats, the organizations face difficulty in effectively tackling cybersecurity issues alone. Though an organization's sole security investigation may lead to developing potential cyber-defense solutions, this reactive approach may not help in better understanding the cybersecurity landscape and take proactive measures to reduce future exploits. The recently proposed cyber-threat information (CTI) sharing scheme is envisioned to help the organizations in enhancing their security standpoints. In addition to organizations' own internal efforts, such sharing could complement their cybersecurity handling tactics and benefit in various means such as: (1) fostering cyber situational awareness, (2) developing proactive defense mechanisms, (3) clarity in understanding the threat landscape, malicious actors, security loopholes etc. Thus, organizations collaborations could decrease the time of threat detection, while increasing the accuracy of detection [1], [2] at the same time.

The mechanisms for timely sharing actionable cybersecurity information such as detection signatures or vulnerabilities are paramount for enabling the cooperative cyberdefence [3]. There are also other approaches which aim to facilitate automatic sharing such as TAXII [4]. The Cybersecurity Information Sharing (CIS) has been encouraged worldwide by the governments through a number of legal initiatives [5]. For instance, the CIS act has proposed a structure managed by governments which will gather as well as distribute cybersecurity threat information. The CIS partnership which is similar to the earlier effort. lunched in 2013 in UK, is a joint industry-government initiative to share the cybersecurity related information. Additionally, in 2015 the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) published an international standard to provide guidance in the sensitive information exchange. This standard also serves for the implementation of information security management within information sharing communities.

The Scope of Collaborative Security: According to [6], the collaborative security has been defined as " instead of centrally managed policies, the organizations or nodes may share and gather cybersecurity related knowledge from other organizations or nodes to make the security related decisions". The aim of this collaboration is to make the decisions more effectively as the more information are available the more accurate decision can be made using that information. So the "collaborative security" is the joint effort among different security systems by sharing various security related information to make more reasonable and effective decisions. The collaborative security system has been applied in many security related domains such as anti spam, anti malware, intrusion detection etc. The application of collaborative security may range from desktop to mobile environment.

This research is supported by the National Science Foundation (NSF), USA, Award #1739032

978-1-7281-0554-3/19/$31.00 ©2019 IEEE

A. Motivation

Due to the availability of the large amount of data in cyber infrastructure and the increasing number of cyber criminals attempting to gain unauthorized access to the data, different machine learning based techniques are necessary to build proactive cyber defense system which will help to defend against any intrusion before harming the system by making right decision or prediction of unusual behaviour or anomalies. This can be achieved by learning the existing dataset where there is information about the various intrusions or attacks as well as their responses. Once the system finishes learning from the trained dataset, it is able to detect if new intrusion happens.

The number of security incidents worldwide is increasing and the security community relies on the ability to detect and to react to such threats. Historically, information security is a continuous cycle where vulnerabilities are discovered, exploited by malicious actors, and patched by the information security community. As new vulnerabilities and exploits are observed, signatures or patterns indicating malicious activity are created. These signatures are used by Intrusion Detection Systems (IDS) to detect malicious activity in networks. The IDS create alarms for human analysts for which to decide on what action to be taken. Unfortunately, many of these alarms are False Positive (FP), that is wrongly raised alarms. Applying Machine Learning (ML) approaches to event classification can provide great benefits to the daily operation of a Security Operation Center (SOC) [2].

Any organization can get benefits if it can receive various types of CTI data from other organizations. By sharing CTI the organization can learn to be aware of varieties of threats and can prepare itself before actually getting compromised by those threats. Therefore, more the organizations share their information with each other, more they learn about the existing threats and the way to deal with them. This is how they can increase their accuracy of prediction of any activities being malicious or not.

However, collaboration through threat intelligence exchange still has certain challenges: (1) the possibility of information exploitation as the sharing organizations may not trust other participants, (2) concerns of privacy of sensitive information which may get exposed to attackers or other competitors, (3) organizations' reputation might get negatively affected if the vulnerability information can identify an organization.

Therefore, we formulate the problem statement as how to develop a learning mechanism of CTI in privacy preserving manner so that no private information is leaked to any party and at the same time organization can make decisions from the learned model from the new activity.

B. Contributions

The main contributions of this paper can be summarized as follows.

- We propose a privacy preserving collaborative cyber threat information sharing and learning framework where multiple organization can share and learn each others' information in a privacy preserving manner to detect future malicious incidents.
- We propose a privacy preserving learning algorithm based on decision tree where organizations can learn the global decision tree which includes other organizations' information without revealing their local data. Based on this global tree the organizations can make predictions or classifications on newly arrived threat information. As an example of application, we test the framework to learn the decision tree on spam email dataset in privacy preserving manner which can be used to detect if any new email is spam or ham without disclosing any private information.

II. Related Work

The existing work shows a notable attempts to explore the paradigm of collaborative security and review associated methods. However, the scopes of such attempts are often restricted to specific domains, which lack systematic analysis and classification. The coalitional approaches for cybersecurity information sharing and the underlying privacy challenges have been studied in [7]–[11]. A framework for privacy preservation of cybersecurity information sharing has been proposed by [9]. This scheme uses group signature to hide the identities of the organizations. However, this scheme does not protect the participants' information. [7] has modeled the privacy issue in cybersecurity information sharing as a game between organizations and attackers. Although such a model helps the organizations to decide their sharing strategy, it does not provide any practical solution to protect the underlying information. The [6] presented a collection of collaborative security related research. The main issue is that they lack detailed and insightful analysis with summarization. The [12] presented common building blocks of collaborative intrusion detection system which specifically includes the information sharing as well as system security. They also presented the privacy preservation scheme during sharing any security related information. The challenges in collaborative intrusion detection system were shown in [13]. They surveyed various coordinated attacks that traditional intrusion detection systems cannot detect. Zhou et al. introduced a new kind of intrusion detection system through a collaborative lens. Besides above research works, there early research which looked into specific aspects of collaborative security, however, they did not consider the entirety of the topic. The [14] surveyed multiple categories of collective anomalies, and present key challenges for each category. They also investigated a series of methods to handle these collective anomalies as well as a thorough comparison between these methods. The [15], for example, discussed one particular field in collaborative intrusion detection systems which is alert correlation. Their research surveyed a considerable number of applied approaches of alert correlation and presented the strengths and weaknesses respectively. Caruana and Li [16] also conducted a survey of spam filtering approaches, specifically those dealing with collaboration, and provided a summary of the practical applications.

III. Prelimenaries

A. Homomorphic Encryption

In our privacy preserving protocol, we use ElGamal cryptosystem [17] to leverage their homomorphic properties while performing the computations. The ElGamal encryption scheme is a probabilistic public key encryption algorithm which is composed of key generation, encryption and decryption. Specifically we have used the distributed version [18] of ElGamal cryptosystem which supports both homomorphic addition and multiplication. It has been very effective in some of existing researches of privacy preserving applications [19]–[23] along with other similar types of homomorphic encryption based applications [24], [25]. The ElGamal cryptosystem and its distributed variant work as follows. We assume there are n users in the system. Each i-th user has its own public key y_i and secret key x_i. The distributed ElGamal cryptosystem consists of the following algorithms.

Key generation: A common public key is used in the distributed ElGamal cryptosystem as follows:

$$PK = \prod_{i=1}^{n} y_i = g^{x_1+\cdots+x_n} \tag{1}$$

Encryption: To encrypt a plaintext message $m \in G$: an integer r is randomly chosen from $\mathbb{Z}_q^*$; then the ciphertext computation becomes: $c_1 = g^r$ and $c_2 = g^m \cdot PK^r$.

The encrypted message is $E(m) = (c_1, c_2)$.

Decryption: A common decryption key is not computed. Each user computes and broadcasts a partially decrypted value, and the final plaintext is revealed by combining all partially decrypted values. For the ciphertext (c_1, c_2), decryption proceeds as follows:

- Each i-th user computes ${c_1}^{x_i}$;
- All users broadcast commitment of computed values $H({c_1}^{x_i})$;
- Each i-th user broadcasts ${c_1}^{x_i}$ and checks if each $c_1^{x_i}$ matches with $H(c_1^{x_i})$;
- Each user computes $\dfrac{c_2}{\prod_{i=1}^{n} {c_1}^{x_i}} = \dfrac{c_2}{c_1^{x_1+\cdots+x_n}} = g^m$.

Finally, m can be revealed by computing a discrete logarithm.

Homomorphic Property ElGamal encryption has an inherited homomorphic property [26], which allows multiplication and exponentiation to be performed on a set of ciphertexts without decrypting them, such as addition homomorphic computation

$$\begin{aligned} E(m_1) \times E(m_2) &= (g^{r_1}, g^{m_1} \cdot pk^{r_1}) \times (g^{r_2}, g^{m_2} \cdot pk^{r_2}) \\ &= (g^{r_1+r_2}, g^{m_1+m_2} \cdot pk^{r_1+r_2}) \\ &= E(m_1 + m_2) \end{aligned} \tag{2}$$

and multiplication homomorphic computation:

$$\begin{aligned} E(m_1)^{m_2} &= (g^{r_1}, g^{m_1} \cdot pk^{r_1})^{m_2} \\ &= (g^{r_1 \cdot m_2}, g^{m_1 \cdot m_2} \cdot pk^{r_1 \cdot m_2}) \\ &= E(m_1 \cdot m_2) \end{aligned} \tag{3}$$

B. Decision Tree

Decision tree algorithm is a renowned classification algorithm. In a decision tree, the nodes are representing the object attributes and categories. The edges are the possible outcome of a decision, and each leaf node is assigned a category. To classify an object into a category, the algorithm begins from the root of the decision tree and the object is checked for the corresponding attribute at each internal node. Then, the algorithm goes down the tree along the edge corresponding to the objects value for that attribute. This traversal of a tree proceeds till a leaf node is reached. Thus, an object's category is decided based on its path from the root to a leaf of the decision tree.

The ID3 algorithm receives a set of samples and generates a decision tree in a top-down manner. It begins at the root and specifies the best attribute which makes the optimal classification of the objects. Then, an edge is created for every possible value of the attribute. Recursively, this process generates the other nodes and their corresponding edges. Once all of the attributes are examined, then the tree construction is finished. The information theory is used to decide the attribute that gives the best prediction at each internal node. The attribute that decreases the entropy of the category information the most is selected as the best attribute. Assume that there are k categories $c_1, ..., c_k$, and a set T of objects whose categories are known. Let $T(c_i)$ be the set of objects with category c_i. Then the information needed to classify an object in T is:

$$E(T) = \sum_{i=1}^{k} \left(-\frac{|T(c_i)|}{|T|} \cdot log\frac{|T(c_i)|}{|T|}\right) \tag{4}$$

Lets consider the objects contain n attributes $A_1 ... A_n$. To access the prediction quality of any attribute A we need to calculate the information needed to classify an object in T given its value for attribute A. Assume that A can have p values $a_1 a_p$. Then the information of T given A is:

$$E(T|A) = \sum_{i=1}^{p} \left(\frac{|T(a_i)|}{|T|} \cdot E(T(a_i))\right) \tag{5}$$

The information gain of attribute A can be written as

$$G(A) = E(T) - E(T|A) \tag{6}$$

The best attribute A is then the one that has the maximum gain, i.e., minimum $E(T|A)$, among all considered attributes.

IV. System Model

In our proposed model, we assume there are total t organizations $\{O_1, O_2, ..., O_t\}$ who collaborate with each other by sharing their aggregated information to learn the decision tree in privacy preserving manner. They encrypt their own information, and send the ciphertexts to a Central Server (CS) which has enough computation power to perform homomorphic operations. The CS finds the encrypted results and sends it back to the organizations without learning any private information of the organizations. We assume the parties in our system are semi-honest but curious, which means they follow

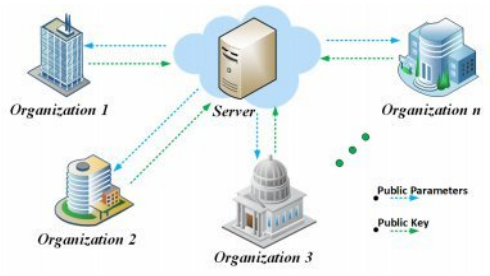

Fig. 1. System model for key generation

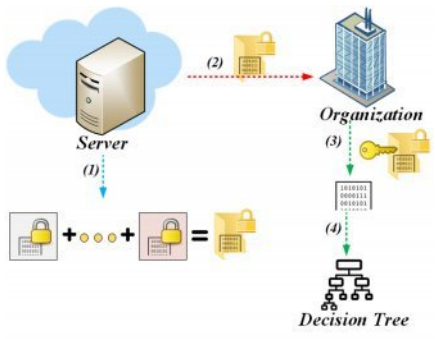

Fig. 2. Server performs homomorphic operations to get the encrypted aggregated results. This encrypted result is sent to the corresponding organization which then can decrypt using certain protocol and use the decrypted result to build the decision tree.

the protocol but try to learn as much as information they can. Figure 1. shows the system model for key generation where the organizations generate their own key pairs and send the public keys to the CS. Then the CS aggregates the individual public keys to get the master public key which is broadcast to all organizations. This master public key is used to encrypt individual private information. Upon receiving the encrypted information, the CS then performs homomorphic operations to get aggregated encrypted results which then are shared among the organizations. The organizations then collaborate to decrypt the results with the help of CS using their own secret keys without revealing any private information to CS or other organizations. Finally based on the decrypted results the organizations can build the decision tree which can be used to predict any future events related to cyber threat such as email spam detection. The main architecture of our proposed model is shown in Figure 2.

V. Privacy Preserving Collaborative Decision Tree

In this section we present privacy preserving ID3 based decision tree algorithm. Note that, the ID3 algorithm requires to compute logarithm and computing a logarithm privately is in general a complex task and requires specialized protocols to be applicable in practice [27]. Instead of computing a logarithm securely we choose to go a different well known splitting measure to avoid secure computation of logarithms. Our protocols will be based on the Gini index, which is another common splitting measure that can be implemented using simple arithmetic only. The Gini index measures the probability of incorrectly classifying transactions in T if classification is done randomly according to the distribution of the class values in T [28], and is given by

$$E(T) = 1 - \sum_i \left(\frac{|T(a_i)|}{|T|}\right)^2 \tag{7}$$

$$\begin{aligned} E(T|A) &= \sum_{i=1}^{p}\left(\frac{|T(a_i)|}{|T|} \cdot E(T(a_i))\right) \\ &= \frac{1}{|T|}\sum_{i=1}^{p}\left(|T(a_i)| \cdot \left(1 - \sum_j^k \left(\frac{|T(a_i.c_j)|}{|T(a_i)|}\right)^2\right)\right) \\ &= \frac{1}{|T|}\left(\sum_{i=1}^{p}|T(a_i)| - \sum_{i=1}^{p}\sum_j^k \left(|T(a_i.c_j)|\right)^2\right) \end{aligned} \tag{8}$$

A. Initialization

Each organization O_k generates its own secret and public key pairs (x_k, y_k) where x_k and $y_k = g^{x_k}$ (g is a generator which is public parameter) represent its secret and public key respectively. Then the O_k sends the y_k to CS. The CS generates master public key as follows:

$$PK = \prod_k^t y_k = g^{\sum x_k} \tag{9}$$

Then the CS broadcasts the PK to all organizations to encrypt their own aggregated results which they dont want to share in plaintext.

B. Protocol

Step 1: Each organization o_k has to compute $|T(a_i)|$ and $(|T(a_i, c_j)|)^2$ from their own dataset and share with other organizations to get aggregated results. Therefore, all organizations jointly need to compute as follows:

$$A = \sum_k^t \sum_{i_k=1}^{p_k} |T(a_{i_k})| \tag{10}$$

and

$$B = \sum_k^t \sum_{i_k=1}^{p_k} \sum_{j_k}^{q_k} \left(|T(a_{i_k}.c_{j_k})|\right)^2 \tag{11}$$

Step 2: The o_k encrypts its own $\sum_{i=1}^{p}|T(a_i)|$ and $\sum_{i=1}^{p}\sum_j^k \left(|T(a_i.c_j)|\right)^2$ as

$$\begin{aligned} C_{11}^k, C_{21}^k &= g^{r_k}, g^{\sum_{i_k=1}^{p_k}|T(a_{i_k})|} \cdot PK^{r_k} \\ C_{12}^k, C_{22}^k &= g^{r_k}, g^{-\sum_{i_k=1}^{p_k}\sum_{j_k}^{q_k}\left(|T(a_{i_k}.c_{j_k})|\right)^2} \cdot PK^{r_k} \end{aligned} \tag{12}$$

Step 3: The organizations sends the ciphertexts to CS. Then the CS performs homomorphic additions as,

$$\begin{aligned} C_{11}, C_{12} &= \prod_k^t (g^{r_k}, g^{\sum_{i_k=1}^{p_k}|T(a_{i_k})|} \cdot PK^{r_k}) \\ &= g^{\sum_k^t r_k}, g^{\sum_k^t \sum_{i_k=1}^{p_k}|T(a_{i_k})|} \cdot PK^{r_k} \end{aligned} \tag{13}$$

$$C_{21}, C_{22} = \prod_{k}^{t} (g^{r_k}, g^{\sum_{i_k=1}^{p_k} \sum_{j_k}^{q_k} \left(|T(a_{i_k}.c_{j_k})|\right)^2} \cdot PK^{r_k})$$
$$= g^{\sum_k^t r_k}, g^{-\sum_k^t \sum_{i_k=1}^{p_k} \sum_{j_k}^{q_k} \left(|T(a_{i_k}.c_{j_k})|\right)^2} \cdot PK^{r_k} \quad (14)$$

$$C_{31}, C_{32} = (C_{21}, C_{22}) \cdot (C_{21}, C_{22})$$
$$= g^{\sum_k^t 2r_k}, g^{\sum_k^t \sum_{i_k=1}^{p_k} |T(a_{i_k})| - \sum_k^t \sum_{i_k=1}^{p_k} \sum_{j_k}^{q_k} \left(|T(a_{i_k}.c_{j_k})|\right)^2}$$
$$\cdot PK^{2\sum r_k} \quad (15)$$

For each attribute A the CS calculates the ciphertexts using equation 15 and gets the minimum one using encrypted comparison protocol. Then the attribute with minimum value is selected as root node. To find the minimum attribute we use the process described in [29] where two private values for example a_1 and a_2 can be compared with the involvement of t organizations without revealing any of the private values.

Step 4: After performing homomorphic additions, the CS broadcasts C_{31} and C_{32} to all organizations for preparing the decision tree. Then each o_k hides their own secret keys into the ciphertexts as $C_{31}^{x_k}$ and shares it to other organizations. Therefore each organization o_k has $C_{31}^{x_1}, ..., C_{31}^{x_t}$ which are used to decrypt the the cphertexts (C_{31}, C_{32}). Finally each o_k decrypts the ciphertexts as

$$D(C_{31}, C_{32}) = \frac{C_{32}}{\prod_k^t (C_{31})^{x_k}}$$
$$= g^{\sum_k^t \sum_{i_k=1}^{p_k} |T(a_{i_k})| - \sum_k^t \sum_{i_k=1}^{p_k} \sum_{j_k}^{q_k} |\left(|T(a_{i_k}.c_{j_k})|\right)^2} \quad (16)$$

To get the exponent, the organization performs discrete log as $log_g^{D(C_{31}, C_{32})}$.

Theorem 1. *If the organizations and the CS follow the protocol we have that* $D(C_{31}, C_{32}) = g^{\sum_k^t \sum_{i_k=1}^{p_k} |T(a_{i_k})| - \sum_k^t \sum_{i_k=1}^{p_k} \sum_{j_k}^{q_k} \left(|T(a_{i_k}.c_{j_k})|\right)^2}$

Proof. From equation 16 we have,

$$D(C_{31}, C_{32}) = \frac{C_{32}}{\prod_k^t (C_{31})^{x_k}}$$
$$= \frac{g^{\sum_k^t \sum_{i_k=1}^{p_k} |T(a_{i_k})| - \sum_k^t \sum_{i_k=1}^{p_k} \sum_{j_k}^{q_k} \left(|T(a_{i_k}.c_{j_k})|\right)^2} PK^{2\sum r_k}}{\prod_k^t (g^{\sum_k^t 2r_k})^{x_k}}$$
$$= g^{\sum_k^t \sum_{i_k=1}^{p_k} |T(a_{i_k})| - \sum_k^t \sum_{i_k=1}^{p_k} \sum_{j_k}^{q_k} \left(|T(a_{i_k}.c_{j_k})|\right)^2} \quad (17)$$

□

VI. Privacy Analysis

This section provides the privacy analysis of our proposed protocol for each stage. More specifically in this section, we discuss what information the participants are able to learn and how far they can achieve to harm the system if they are malicious. We have followed the structure

Theorem 2. *Proposed protocol is private and no entity is able to learn any private information from the protocol.*

Proof. At the initialization stage, each organization generates its own secret and public key pair, then share only the public key with CS to produce a master public key according to equation 9. Note that in this process none of the organizations reveals any private information except their own public key with CS. In the main protocol part, each organization o_k encrypts their own information as $E(\sum_{i=1}^{p} |T(a_i)|)$ and $E(\sum_{i=1}^{p} \sum_{j}^{k} \left(|T(a_i.c_j)|\right)^2)$ using the master public key PK and broadcasts the ciphertexts to CS as shown in equation 12. The CS performs homomorphic addition on these ciphertexts over all organizations as shown in equation 13 and 14 to get $E(\sum_k^t \sum_{i_k=1}^{p_k} |T(a_{i_k})|)$ and $E(-\sum_k^t \sum_{i_k=1}^{p_k} \sum_{j_k}^{q_k} \left(|T(a_{i_k}.c_{j_k})|\right)^2)$. Finally the CS performs another homomorphic addition to get $E(\sum_k^t \sum_{i_k=1}^{p_k} |T(a_{i_k})| - \sum_k^t \sum_{i_k=1}^{p_k} \sum_{j_k}^{q_k} \left(|T(a_{i_k}.c_{j_k})|\right)^2)$. During these process the CS learns nothing except the ciphetexts and their resultant ciphertexts after performing the homomorphic operations. This resultant ciphertext is broadcast to all organizations where the organizations collaborate with each other to decrypt the results. According to step 4 the organizations collaborate with CS by raising the ciphertest C_{31} to the power of their own secret key x_k and the the organization o_k uses equation 16 to decrypt the results. During this process none of the secret keys is disclosed to any other party and the organizations perform the decryption locally.

□

VII. Performance Analysis

Our experimental analysis is divided into two main sections. First, we discuss the complexity of the proposed protocols in terms of computation and communication costs. Then based on the complexities, we run experiments and show the actual time and bandwidth taken by the protocol on our specific settings. In the complexity analysis, we analyze the encryption/ decryption and the data transmission costs for the single execution of the protocol. We consider one encryption in ElGamal cryptosystem is equivalent to $2m_e$ where m_e represents modular exponentiation. Let, one homomorphic addition, homomorphic multiplication and discrete logarithm for decryption represented as h_a, h_m and $\sqrt{T}m_e$ respectively where T represents the size of the plaintext. We consider the units of computation and communication costs as seconds and bits respectively. To test the performance of our protocol we experimented on spam email classification using the proposed privacy preserving decision tree learning. More specifically,

we gather the dataset of spam/ham emails. Then we randomly split the dataset into 20 different subsets which represent 20 different organizations who are supposed to share their private information with others in privacy preserving manner and learn the decision tree over all dataset from other organizations without knowing any private information of others. In this experiment we used spam assassin dataset [1]. Then we consider each email as bag of words. For each subset of dataset in each organization, we create the decision table as the words as features and the value of the features are either 1 or 0 depending on the presence of that particular word in the email. Each organization holding the subset of dataset will perform the intermediate computations and share the results as encrypted form with the server. Upon receiving the encrypted aggregated results over all organiations' dataset from the server, the organization can generate the decision tree by decrypting the results locally. In below we discuss the computation and communication complexities in which the time and bandwidth are represented as seconds and bits respectively.

A. Computation Complexity

According to the protocol, we have two main sections which are initialization and privacy preserving collaborative learning protocol. In the initialization protocol, the server collects all the public keys and produce master public key for all organizations. In this computation server perform k multiplication over the public parameter as shown in equation 9. Therefore the computation complexity of initialization is k seconds. In the main protocol of section VI-B, each organization performs two encryptions as shown in equation 14 with the complexity of $2m_e$ seconds. According to step 3, the server performs three different homomorphic additions where two of them are for t organizations. Therefore, the complexity becomes $2h_a \times t + h_a$ seconds. Then the server finds the minimum results among the n ciphertexts which results the complexity of n seconds. After the final resultant ciphertexts are broadcast to the organizations, they decrypt the results locally, as described in step 4 which gives the complexity of $m_e + t \times m_e$, since each organization hides their secret keys x_k by raising the ciphertext to the power of x_k (as shown in step 4) and performing t modular exponentiation. Finally, the organization performs discrete logarithm which takes $\sqrt{T} \times m_e$ seconds. Therefore, the total complexity of decryption becomes $O(m_e + t \times m_e + \sqrt{T} \times m_e)$ seconds. Finally, the total computation costs of each organization and server become $k + 2m_e + m_e + t \times m_e + \sqrt{T} \times m_e$ and $2h_a \times t + h_a + n$ seconds.

B. Communication Complexity

For communication complexity we assume that the ciphertexts exchanged between the parties are of l bits. At the initialization phase the server receives the public keys from all organizations. Therefore it takes $t \times l$ bits and l bits for server and each organization respectively to receive the public parameters for server from the organizations. In the main protocol, each organization sends two pairs of ciphertexts to the server as shown in step 3 and the communication costs becomes $4l$ bits for each organization. The server receives from t organization which results in $4l \times t$ bits to receive the ciphertexts. After performing homomorphic operations the server broadcasts a pair of ciphertext which are received by each organizations of $2l$ bits. During the decryption process, the server and each organization exchange another set of ciphertexts (shown in step 4). In this process the bandwidth requirements becomes $2l$ bits for each organizations and $2l \times t$ bits for the server. Therefore the total data require for each organization and server are $7l$ and $7t \times l$ bits respectively.

[1] http://spamassassin.apache.org/publiccorpus/.

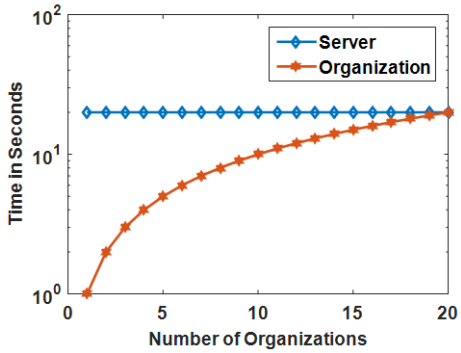

Fig. 3. Computation cost of server and each organization interms of increasing total number of organizations.

C. Efficiency

Table 1 shows the performance results of our proposed protocol to build the decision tree for server and each organization. We consider there are 20 organizations, therefore $t = 20$. We also consider that there are $n = 20$ different ciphertexts among which the server has to run the comparison protocol to find the minimum value without learning the actual value. In our experimental setup one modular exponentiation and one homomorphic addition take 2×10^{-5} and 5.7×10^{-5} seconds respectively. Based on this setup, in our protocol, the server and each organization take 20 seconds to finish the protocol or to learn the build the decision tree locally in each organization's side. Once the decision tree building is complete the organization can use the tree to predict if any new email is spam or ham. Since the prediction can be done locally without applying any privacy preserving protocol, the efficiency of prediction protocol is out of our scope. Figure 1 shows the scalability of our proposed protocol with increasing the number of organizations. It shows that for each organization, the computation time increases if the number of organization increases. However in our experiment, the computation time of server didnt increase since the number of ciphertexts for comparison protocol was fixed which was 20.

VIII. Conclusion

We proposed a privacy preserving protocol to learn the decision tree algorithm which can be applied in proactive cyber

TABLE I
COMPUTATION COMPLEXITY AND TIME ($t = 20$)

	Computation Complexity	Communication Complexity
Server	$2h_a \times t + h_a + n$	$7l$
Organization	$k + 2m_e + m_e + t \times m_e + \sqrt{T} \times m_e$	$7t \times l$
	Cost in Seconds	Cost in MB
Server	20	0.0008
Organization	20	0.01

defense by classifying if an email in an organization is spam or ham. Although this is a simple approach to learn the decision tree over a suitable dataset but the challenges remain in learning the tree in privacy preserving manner without disclosing any private information while sharing them with other parties. Our proposed protocol can address these challenges and learn the decision tree over email dataset without disclosing any private information of organization. This protocol can also be applied to build the deicison trees from other types of cyber threat datasets as long as the features are of numerical values (such as phishing data) as the homomorphic encryptions can not be applied to fractional numbers. The proposed model can be of a great use for proactive cyber defense system as it can learn the dataset in privacy preserving manner by building the tree. In future, we are interested to build collaborative and proactive cyber defense system using unsupervised machine learning algorithms in privacy preserving manner. The privacy analysis and experimental results of proposed protocol show that it is private as well as practical.

REFERENCES

[1] S. Katti, B. Krishnamurthy, and D. Katabi, "Collaborating against common enemies," in *Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*. USENIX Association, 2005, pp. 34–34.

[2] M. E. Locasto, J. J. Parekh, A. D. Keromytis, and S. J. Stolfo, "Towards collaborative security and p2p intrusion detection," in *Information Assurance Workshop, 2005. IAW'05. Proceedings from the Sixth Annual IEEE SMC*. IEEE, 2005, pp. 333–339.

[3] D. E. Denning, "Framework and principles for active cyber defense," *Computers & Security*, vol. 40, pp. 108–113, 2014.

[4] R. A. Martin, "Making security measurable and manageable," in *Military Communications Conference, 2008. MILCOM 2008. IEEE*. IEEE, 2008, pp. 1–9.

[5] F. Skopik, G. Settanni, and R. Fiedler, "A problem shared is a problem halved: A survey on the dimensions of collective cyber defense through security information sharing," *Computers & Security*, vol. 60, pp. 154–176, 2016.

[6] J.-M. Seigneur, *Collaborative Computer Security and Trust Management*. IGI Global, 2009.

[7] I. Vakilinia, D. K. Tosh, and S. Sengupta, "3-way game model for privacy-preserving cybersecurity information exchange framework," in *Military Communications Conference (MILCOM), MILCOM 2017-2017 IEEE*. IEEE, 2017, pp. 829–834.

[8] I. Vakilinia and S. Sengupta, "A coalitional game theory approach for cybersecurity information sharing," in *Military Communications Conference, MILCOM 2017-2017 IEEE*. IEEE, 2017, pp. 237–242.

[9] I. Vakilinia, D. K. Tosh, and S. Sengupta, "Privacy-preserving cybersecurity information exchange mechanism," in *Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2017 International Symposium on*. IEEE, 2017.

[10] I. Vakilinia and S. Sengupta, "A coalitional cyber-insurance framework for a common platform," *IEEE Transactions on Information Forensics and Security*, 2018.

[11] I. Vakilinia, S. Cheung, and S. Sengupta, "Sharing susceptible passwords as cyber threat intelligence feed," in *Military Communications Conference (MILCOM), MILCOM 2018-2018 IEEE*. IEEE, 2018.

[12] R. Bye, S. A. Camtepe, and S. Albayrak, "Collaborative intrusion detection framework: Characteristics, adversarial opportunities and countermeasures." in *CollSec*, 2010.

[13] C. V. Zhou, C. Leckie, and S. Karunasekera, "A survey of coordinated attacks and collaborative intrusion detection," *Computers & Security*, vol. 29, no. 1, pp. 124–140, 2010.

[14] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, p. 15, 2009.

[15] H. T. Elshoush and I. M. Osman, "Alert correlation in collaborative intelligent intrusion detection systemsa survey," *Applied Soft Computing*, vol. 11, no. 7, pp. 4349–4365, 2011.

[16] G. Caruana and M. Li, "A survey of emerging approaches to spam filtering," *ACM Computing Surveys (CSUR)*, vol. 44, no. 2, p. 9, 2012.

[17] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE transactions on information theory*, vol. 31, no. 4, pp. 469–472, 1985.

[18] F. Brandt, "Efficient cryptographic protocol design based on distributed el gamal encryption," in *International Conference on Information Security and Cryptology*. Springer, 2005, pp. 32–47.

[19] A. Kelarev, X. Yi, S. Badsha, X. Yang, L. Rylands, and J. Seberry, "A multistage protocol for aggregated queries in distributed cloud databases with privacy protection," *Future Generation Computer Systems*, vol. 90, pp. 368–380, 2019.

[20] S. Badsha, X. Yi, and I. Khalil, "A practical privacy-preserving recommender system," *Data Science and Engineering*, vol. 1, no. 3, pp. 161–177, 2016.

[21] S. Badsha, X. Yi, I. Khalil, D. Liu, S. Nepal, E. Bertino, and K.-Y. Lam, "Privacy preserving location-aware personalized web service recommendations," *IEEE Transactions on Services Computing*, 2018.

[22] S. Badsha, I. Khalil, X. Yi, and M. Atiquzzaman, "Designing privacy-preserving protocols for content sharing and aggregation in content centric networking," *IEEE Access*, vol. 6, pp. 42 119–42 130, 2018.

[23] S. Badsha, X. Yi, I. Khalil, D. Liu, S. Nepal, and K.-Y. Lam, "Privacy preserving user based web service recommendations," *IEEE Access*, vol. 6, pp. 56 647–56 657, 2018.

[24] S. Badsha, X. Yi, I. Khalil, D. Liu, S. Nepal, and E. Bertino, "Privacy preserving location recommendations," in *International Conference on Web Information Systems Engineering*. Springer, 2017, pp. 502–516.

[25] S. Badsha, X. Yi, I. Khalil, and E. Bertino, "Privacy preserving user-based recommender system," in *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2017, pp. 1074–1083.

[26] X. Yi, R. Paulet, and E. Bertino, *Homomorphic encryption and applications*. Springer, 2014, vol. 3.

[27] S. de Hoogh, B. Schoenmakers, P. Chen, and H. op den Akker, "Practical secure decision tree learning in a teletreatment application," in *International Conference on Financial Cryptography and Data Security*. Springer, 2014, pp. 179–194.

[28] L. E. Raileanu and K. Stoffel, "Theoretical comparison between the gini index and information gain criteria," *Annals of Mathematics and Artificial Intelligence*, vol. 41, no. 1, pp. 77–93, 2004.

[29] F. Kerschbaum, D. Biswas, and S. de Hoogh, "Performance comparison of secure comparison protocols," in *20th International Workshop on Database and Expert Systems Application*. IEEE, 2009, pp. 133–136.