A fog-based privacy-preserving approach for distributed signature-based intrusion detection

Wang, Yu; Meng, Weizhi; Li, Wenjuan; Li, Jin; Liu, Wai-Xi; Xiang, Yang

Published in:
Journal of Parallel and Distributed Computing

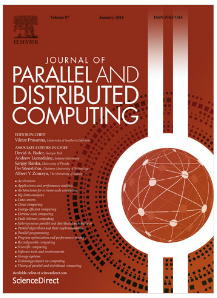
Link to article, DOI:
10.1016/j.jpdc.2018.07.013

Publication date:
2018

Document Version
Peer reviewed version

Link back to DTU Orbit

Citation (APA):
Wang, Y., Meng, W., Li, W., Li, J., Liu, W-X., & Xiang, Y. (2018). A fog-based privacy-preserving approach for distributed signature-based intrusion detection. *Journal of Parallel and Distributed Computing*, *122*, 26-35. https://doi.org/10.1016/j.jpdc.2018.07.013



A fog-based privacy-preserving approach for distributed signature-based intrusion detection

Yu Wang, Weizhi Meng, Wenjuan Li, Jin Li, Wai-Xi Liu, Yang Xiang

PII: S0743-7315(18)30505-7
DOI: https://doi.org/10.1016/j.jpdc.2018.07.013
Reference: YJPDC 3918

To appear in: *J. Parallel Distrib. Comput.*

Received date : 29 January 2018
Revised date : 6 May 2018
Accepted date : 15 July 2018

Please cite this article as: Y. Wang, W. Meng, W. Li, J. Li, W.-X. Liu, Y. Xiang, A fog-based privacy-preserving approach for distributed signature-based intrusion detection, *J. Parallel Distrib. Comput.* (2018), https://doi.org/10.1016/j.jpdc.2018.07.013

Highlights

1. We design a privacy-preserving framework for signature-based intrusion detection in a distributed environment, based on fog devices.
2. In the evaluation, we investigated the performance of our approach in both simulated and real network environments.
3. We compared our approach with similar approaches like PPIDS, and experimental results demonstrated that our approach could help secure data, reduce the workload on the cloud's side and offer less detection delay.

A Fog-based Privacy-Preserving Approach for Distributed Signature-based Intrusion Detection[☆]

Yu Wang[a], Weizhi Meng[☆☆b], Wenjuan Li[b,c], Jin Li[a], Wai-Xi Liu[a], Yang Xiang[d]

[a]*School of Computer Science, Guangzhou University, China*
[b]*Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark*
[c]*Department of Computer Science, City University of Hong Kong, Hong Kong SAR*
[d]*Digital Research and Innovation Capability Platform, Swinburne University of Technology, Australia*
[e]*E-mail address: yuwang@gzhu.edu.cn;weme@dtu.dk*

Abstract

Intrusion detection systems (IDSes) are the frontier of defense against transmissible cyber threats that spread across distributed systems. Modern IDSes overcome the limitation of hardware processing power by offloading computation extensive operations such as signature matching to the cloud. Moreover, in order to prevent the rapid spread of transmissible cyber threats, collaborative intrusion detection schemes are widely deployed to allow distributed IDS nodes to exchange information with each other. However, no party wants to disclose their own data during the detection process, especially sensitive user data to others, even the cloud providers for privacy concerns. In this background, privacy-preserving technology has been researched in the field of intrusion detection, whereas a collaborative intrusion detection network (CIDN) environment still lacks of appropriate solutions due to its geographical distribution. With the advent of fog computing, in this paper, we propose a privacy-preserving framework for signature-based intrusion detection in a distributed network based on fog devices. The results in both simulated and real environments demonstrate that our proposed framework can help reserve the privacy of shared data, reduce the workload on the cloud side, and offer less detection delay as compared to similar approaches.

Keywords:
Collaborate Network, Privacy Preserving, Intrusion Detection, Cloud Environment, Fog Computing.

1. Introduction

Transmissible cyber threats have become a major security challenge in the cyber space, due to their ability to infect and spread rapidly in distributed systems. For instance, the WannaCry ransomware [4], which exploits a publicly known vulnerability in Microsoft Windows SMB, spread through the Internet in May 2017 and infected over 200 thousand computers in more than 150 countries and regions. With the world becoming more connected and more interdependent, greater and greater scale of damages and impacts could be caused by transmissible cyber threats like the WannaCry attack and other types of spreading malware [67]

To build the first line of defense against cyber threats, intrusion detection systems (IDSes) are widely deployed in various distributed systems to monitor a host or a network for the signs of malicious activities and violations of policies [38, 49], such that security breaches can be detected on spot and countermeasures can be triggered immediately to prevent further infections and spread. These systems can be generally classified into signature-based and anomaly-based, according to their detection mechanisms. Signature-based IDSes identify potential attacks by looking for specific patterns that represent known threats, while anomaly-based IDSes detect deviations from established normal behavior profiles, which can be caused by attacks that are previously known or unknown. In addition, based on the monitored targets and deployed locations, an IDS can be categorized as network-based IDS (NIDS) or host-based IDS (HIDS). The former focuses on any threats on network resources and the latter targets on system-level anoma-

[☆] A preliminary version of this paper appears in Proc. of the 9th International Symposium on Cyberspace Safety and Security (CSS), pp. 163-175, 2017 [1].
[☆☆] Corresponding author

lies.

Effective and efficient NIDSes are the key to prevent transmissible cyber threats from spreading across the Internet. However, in the face of the increasing volume of network traffic, the processing power of traditional IDS hardware becomes a bottleneck. Therefore, offloading computation extensive operations of IDSes to the cloud become a trend [42]. For instance, Alharkan and Martin [3] proposed *IDSaaS* for Amazon EC2 cloud, which could monitor and record malicious network behaviors between virtual machines and users within a virtual private cloud. Yassin et al. [53] introduced *CBIDS*, a cloud-based intrusion detection service framework for monitoring network traffic in different layers and detect unexpected activities from different points of a network.

For most cloud-based IDS frameworks, the systems have to upload the packet payload contents and log files to the cloud for inspection, which puts sensitive data of the users in the protected network at risk. For example, with access to the uploaded traffic data and logs, the cloud service provider can easily derive the history of browsing activities of the protected users, which leads to a leakage of user privacy [66]. To address this problem, privacy-preserving technology has been widely studied aiming to protect user privacy in cloud-based IDS. Park et al. [44] proposed *PPIDS*, a privacy-preserving detection approach by applying cryptographic approaches to log files without a trusted third party (TTP). Their system could encrypt the log files and identify intrusions based on encrypted data.

Motivations. It is difficult for standalone IDSes to defend against transmissible cyber threats that spread rapidly across network domains. Distributed intrusion detection systems (DIDSes) or collaborative intrusion detection networks (CIDNs) are an important solution to enhance the detection performance of separate IDSes, where the key is to allow IDS nodes to exchange threat intelligence information with each other [52]. However, uploading all the data to the cloud for processing and analysis would consume considerable communication and computing resources, causing a negative impact on the quality of service (QoS) (i.e., dealing with redundant data [65]). In order to mitigate this issue, fog computing is a paradigm extending cloud computing and its services to the edge of the network (i.e., proximity to end-users or nodes), which can support for mobility, heterogeneity, interoperability and pre-processing [7].

Contributions. Because fog computing provides a computing and storage platform physically closer to the end nodes and users, provisioning a new breed of applications and services with the cloud layer, it well complements the application of cloud computing, which could be proper for a DIDS or CIDN. Further, signature matching is an expensive operation for intrusion detection, which may have a high demand to offload the workload. In this paper, we propose a novel privacy-preserving framework for signature-based IDSes in a distributed environment, based on fog devices. Our contributions can be summarized as below:

- We introduce the background of collaborative intrusion detection environments including its major components and propose a privacy-preserving framework for distributed and collaborative signature-based intrusion detection based on fog devices. The fog computing can provide storage, computing and networking services between an IDS and a cloud. With the provided resources, fog devices could help decrease the workload of a cloud server.

- To protect privacy, we apply Rabin fingerprint algorithm to our proposed framework, and evaluate our approach in a simulated and a real environment, respectively. The experimental results demonstrate that our framework can help secure the data, reduce the workload of a central server on the cloud, and achieve less detection delay as compared to similar approaches like PPIDS.

Organization. The rest of this paper is organized as follows. Section 2 reviews related work on distributed IDSs and privacy-preserving technology. In Section 3, we introduce the background of collaborative intrusion detection networks and the main components. Section 4 describes our proposed privacy-preserving framework and the Rabin fingerprint algorithm for signature-based IDSs. Section 5 evaluates the framework in both simulated and real environments. Finally, Section 6 concludes our work.

2. Related Work

This section introduces related work regarding distributed IDSs, challenge-based CIDNs and the application of privacy-preserving technology in IDSs.

Distributed trust-based intrusion detection. Collaborative intrusion detection networks (CIDNs) [52] can enable an IDS node to achieve better detection accuracy by collecting and exchanging information with other IDS nodes. Li *et al.* [16] identified that most distributed IDSs were depending on centralized fusion, or distributed fusion with unscalable communication mechanisms. Based on this observation, they proposed a DIDS

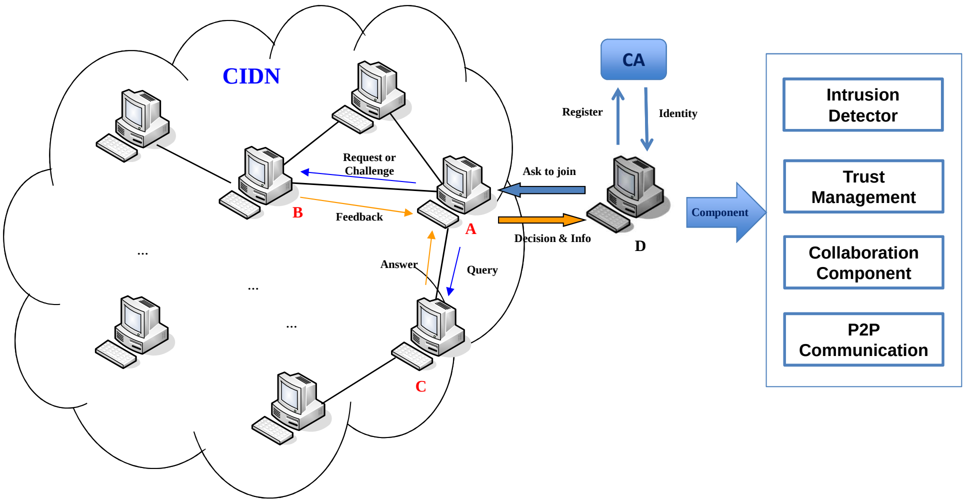


Figure 1: The high-level architecture of a typical challenge-based CIDN.

by considering the emerging decentralized location and routing infrastructure. Their approach assumes that all peers are trusted which is vulnerable to insider attacks (i.e., betrayal attacks where some nodes suddenly become malicious). Khan and Herrmann [15] developed some IDS mechanisms for IoT Networks like small devices, which applied a trust management mechanism for managing reputation of others in a processing and energy-friendly way.

Insider attacks are one of the major threat to a distributed architecture. To address this issue, Duma *et al.* [9] proposed a P2P-based overlay for intrusion detection (Overlay IDS) that could identify insider threat by using a trust-aware engine for correlating alarms from different nodes and an adaptive scheme for managing reputation levels of others. The trust-aware correlation engine is capable of filtering out alarms sent by untrusted or low quality peers, and the adaptive trust management scheme could evaluate and predict a node's trust via their past experiences.

For group trust, Shaikh *et al.* [48] proposed a Group-based Trust Management Scheme (GTMS), which evaluated the trust of a group of sensor nodes for two topologies: intragroup topology and intergroup topology. Guo *et al.* [12] described a trust management framework to generate trust values based on Grey theory and Fuzzy sets. They computed trust values by using relation factors and weights of neighbor nodes, not just by simply taking an average value. A recent survey on trust evaluation can refer to [13, 59].

Challenge-based CIDN. Challenge-based mechanism is a special way of computing reputation levels for IDS nodes, based on the satisfaction level between the received answers and the challenges. Fung *et al.* [10] proposed a HIDS-based collaboration framework that enabled each HIDS to communicate with others and evaluate the trustworthiness of others based on its own experience. They also employ a forgetting factor to give more emphasis on the recent experience of a node. Then, they improved their approach with a Dirichlet-based model in order to measure the reputation levels among IDS nodes according to their mutual experience [11]. Experimental results demonstrated that the new model had strong scalability properties and was robust against common insider threats.

To further improve the detection performance of CIDNs, Li *et al.* [17] figured out that IDSs could have different levels of sensitivity in detecting particular types of attacks based on their own resources like signatures and normal profiles. They then proposed a concept, called *intrusion sensitivity* and explored its impact on evaluating the trust of an IDS node. They further designed a trust management model based on *intrusion sensitivity* to improve the robustness of CIDNs [18], and proposed a machine learning-based approach in automatically allocating the values of *intrusion sensitivity* [22]. In the evaluation, their compared three supervised classifiers in assigning sensitivity values. Experimental results demonstrated that the *intrusion sensitivity* can enhance the detection accuracy of malicious nodes.

On the other hand, Li *et al.* [19] proposed a novel type of collusion attack, called passive message fingerprint attack (PMFA), which can collect messages and identify normal requests in a passive way. In the evaluation, their results demonstrated that malicious nodes under PMFA could identify normal messages and send malicious answers to only normal requests to maintain their trust values. A special On-Off attack (called SOOA) was also developed by them, which could keep behaving normally to one node but send malicious answers to another node [20, 21]. In this case, there is still a need to enhance the security of CIDN frameworks [35], i.e., considering behavior profile [37, 47]. Some other related studies on IDS improvement can refer to alert reduction [29, 31], alert verification [33, 34], blokchain technology [39] and overall filtration mechanism [32].

Privacy-preserving technique and its applications in intrusion detection. Many privacy-preserving schemes have been developed to protect data privacy for intrusion detection during data sharing. For example, Park *et al.* [44] proposed *PPIDS*, a privacy preserving approach for an IDS by applying cryptographic methods to log files without a trusted third party (TTP). Thanks to the use of cryptographic methods, *PPIDS* could prevent users'log information from being monitored and misused. In addition, their approach could provide anonymity (encryption of ID), pseudonymity (encryption of quasi-identifier such as IP address), confidentiality of data, and unobservability. One major issue is that *PPIDS* could lower the performance due to encryptions when log information was stored in SQL table and it could not provide perfect unlinkability.

Regarding the integration of a trusted third party, Benali *et al.* [5] identified and discussed some privacy issues. For example, when several organizations decided to collaborate in detecting intrusive activities, each manger handling organization resources was requested to send the events log to a central unit, which was supposed to act as a trusted entity. In practice, when the analyzer received the event from the participant, a large amount of private information regarding resources and IP addresses would be communicated. In addition, it could be embarrassing for a participant to be pointed out by the third party as a particular weak participant. Niksefat et al. [43] designed ZIDS, a client-server solution for private detection of intrusions. The system consists of an IDS server including sensitive signatures for zero-day attacks and IDS clients for handling sensitive data. They reduced the problem of privacy-preserving intrusion detection to an instance of secure two-party oblivious deterministic finite automata evaluation (ODFA). Their approach was proved to not leak any sensitive information about the nature of the sparsity in the private DFA.

Zhou *et al.* [58] proposed a framework to detect Sybil attacks, while preserving the privacy of users in vehicular ad hoc networks. Their framework could distribute the responsibility of detecting Sybil attacks to semi-trusted third parties. Kerschbaum and Oertel [14] presented a provably secure pattern matching algorithm that could be used for distributed anomaly detection. Their algorithm implemented pattern matching that could be used as the building block for anomaly detection. The experiments indicated that their algorithm was acceptable in RFID anti-counterfeiting. Later, Zhang *et al.* [55] designed a 'semi-centralized' architecture, which used secure multiparty computation (SMC) protocol to conduct a privacy-preserving Principal Component Analysis (PCA), and maintain its scalability and accuracy for anomaly detection. In the evaluation, they showed that none of the participant could learn the private information of other participants during the computation progress. The applications of privacy-preserving technology in other domains can refer to [2, 23, 24, 25, 26, 41, 51, 54, 56, 57, 60, 61, 62, 63, 64].

3. Background on CIDNs

This section introduces the background of collaborative intrusion detection networks (CIDNs) including the major components and node interactions, and describe how a challenge-based CIDN works. Due to the distributed nature, a CIDN is often vulnerable to insider attacks. Thus, trust management is required to protect such network against malicious nodes [40]. Challenge-based trust mechanism is one promising solution for CIDNs, which can evaluate the reputation levels of IDS nodes based on the difference between challenges and received feedback.

Major components. Besides a detection engine that is used for examining traffic, each IDS node often contains several components including *trust management component*, *collaboration component* and *P2P communication*.

- *Trust management component.* This component aims to evaluate the trustworthiness of other nodes. Various trust management approaches can be applied here. Regarding the challenge-based trust mechanism, a node's reputation level can be computed by evaluating the difference between challenges and received answers. Each node can send out either normal requests or challenges for alarm ranking (consultation). To protect challenges, it is worth noting that challenges should be sent out in a random manner and in a way that makes them difficult to be distinguished from a normal alarm ranking request [10].

- *Collaboration component.* This component is responsible for measuring the trustworthiness of IDS nodes by sending out *normal requests* or *challenges*, as well as receiving the corresponding *feedback*. If an IDS node receives a request or challenge, this component can help send back its feedback. Taking Fig. 1 as an example, if node *A* sends a *request or challenge* to node *B*, then node *B* has to send back relevant feedback.
- *P2P communication.* This component is responsible for connecting with other IDS nodes and providing network organization, management and communication among various IDS nodes.

Network Interactions. An IDS node within CIDNs can choose its own collaborators according to its adopted policies and experience. Several IDS nodes can be associated if they have a collaborative relationship. Each node can maintain a list of their collaborated nodes, called *partner list* (or *acquaintance list*). This list contains important information of other nodes, including public keys and their current reputation levels. If a node is willing to join the network, it has to register to a trusted certificate authority (*CA*) and obtain a unique pair of identity like a public key and a private key. As shown in Fig. 1, if node *D* applies for joining the network, it has to contact an insider node, say node *A*. Then, node *A* makes a decision and sends back an initial *partner list*, if node *D* is accepted.

To improve the detection performance, CIDNs allow a set of IDS nodes exchanging necessary messages with each other. There are two major types of messages for interactions.

- *Challenges.* A challenge contains a set of IDS alarms asking for labeling their severity. A testing node can send a challenge to other tested nodes and obtain the relevant feedback. As the testing node knows the severity of the alarms in advance, it can use the received feedback to derive a trust value (e.g., satisfaction level) for the tested node.
- *Normal requests.* A normal request is sent by a node for alarm aggregation. Other IDS nodes should send back a list of alarm ranking as the feedback. Alarm aggregation is an important feature for distributed or collaborative intrusion detection in detecting some complicated threat like DoS. Due to the importance, alarm aggregation process usually only considers the feedback from trusted nodes.

4. Our Approach

This section introduces the concept of fog computing, describes our proposed privacy-preserving framework for distributed and collaborative intrusion detection including the adopted threat model and Rabin fingerprint algorithm for signature matching.

4.1. Fog Computing

Fog computing is proposed by Cisco, which aims to help ease the burden of the IoT server and safeguard the quality-of-service (QoS) [6]. As cloud computing does not need the enterprise and the end user to know specification or many details, it becomes a problem for latency-sensitive applications, which require nodes in the vicinity to meet their delay requirements. For this sake, fog computing is proposed, which enables a new set of applications and services. There is a fruitful interplay between the cloud layer and the fog layer, particularly in the aspects of data management and analytic.

The main idea of fog computing is to provide storage, computing and various networking services between the environmental devices and the cloud side. For this sake, fog devices are often close to end devices, and provide a certain amount of storage and computation resources. With such deployment, fog devices can process the collected data locally, in order to ease the burden of cloud side (e.g., the workload of a central server). For example, the fog devices can perform some specific operations on the received data locally and then forward the results to the central server in the cloud. Therefore, the volume of data sent to the server could be reduced to a large extend.

4.2. Our Proposed Framework

According to the features of fog computing, it can help reduce the delay of intrusion examination, which is suitable for distributed and collaborative intrusion detection. Focusing on rule-based detection and CIDNs, we propose a privacy-preserving framework based on fog devices in Fig. 2, which contains a total of three layers:

- *CIDN layer.* This layer allows various IDS nodes to improve their detection performance by exchanging required information with each other. As signature matching is an expensive process for

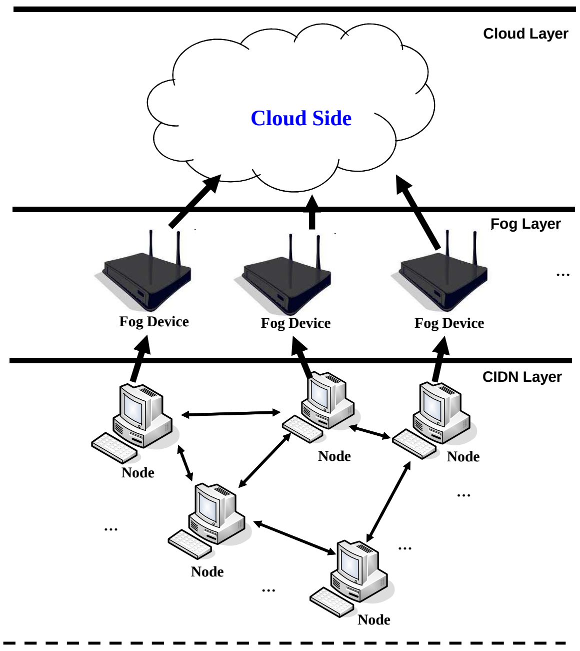


Figure 2: Our privacy-preserving framework for CIDNs using fog devices.

rule-based detection. Thus, those expensive operations like signature matching and sensitive information (e.g., logs) could be offloaded to the cloud side (cloud layer).

- *Cloud layer.* The cloud environment aims to provide sufficient computation resources for expensive operations run in IDSs (CIDN layer), so that data owners can ease the computational burden. However, cloud side cannot ensure an instant reply or return of the computational results, due to the geographical locations. In this work, this issue is expected to be addressed by using fog devices.

- *Fog layer.* This layer attempts to embody software modules and embedded operating systems, which allows to analyze gathered data obtained from the CIDN layer and thus make decisions locally. Local decision making is an important way to reduce latency and provide quick responses to unusual behaviors.

4.3. Threat Model

Due to the wide adoption, this work employs the *curious-but-honest model* for a cloud provider [8]. That is, the cloud provider is trustful to follow the agreed protocol and perform intrusion inspection (i.e., analyzing network traffic for anomalies); however, the cloud provider would attempt to monitor, store, and learn the sensitive (or private) data from the examined traffic, or attempt to discover anything they have an interest.

4.4. Fingerprint Computation for Signature-based Distributed IDSs

Similar to [27], as a study, we select *Rabin fingerprint algorithm* [45] to our proposed framework. More specifically, Rabin fingerprints can be computed using polynomial modulus operations with fast XOR, shift and table look-up operations, resulting in two major merits: 1) one way; 2) and fast computation. In other words, a prime p by random selection can be used to compute the residue of a long-length string. A real-time string matching is a typical application of *Rabin fingerprint algorithm.*

For a signature-based IDSs, these fingerprints can be used during signature matching to meet the real-time requirement. More formally, for a binary string, given a sliding window and an irreducible polynomial $p(x)$, the fingerprint of each k-bit gram can be computed as below:

$$f(x) = m_k + m_{k-1}x + m_{k-2}x^2 + ... + m_1 x^{k-1} mod \ \ p(x) \quad (1)$$

Based on Equation (1), we can generate fingerprints for both IDS signatures and transmitted network packets, and the cloud side can raise an alarm if any packet fingerprint matches the signature fingerprints. However, our previous work [27] indicated that the above straightforward approach has a privacy concern if there is a match between two fingerprints from signatures and packet payloads. For example, the cloud provider can still learn some useful information (i.e., which part of a signature did match), as the signatures may be known.

To resolve this issue, we can perturb fingerprints before sending them to the cloud provider. Note that for the exact matching, it is hard to completely prevent the cloud provider from successfully launching brute-force attacks, but we can reduce the possibility of cracking. As a study, we employ a simple approach; that is, the data owner can select a secret s with a length of l_s and use this secret to perturb the original fingerprints. This approach enables the data owner to decide the length of l_s so that the cloud provider still needs to guess the secret and its length. The equation can be presented as below:

$$f'(x) = f(x) \oplus s \ \ (0 < l_s < |f(x)|) \qquad (2)$$

Security discussion. In current IDS scenario, we consider signatures are sensitive as well. Since we can use Equation (2) to perturb the fingerprints. According to [27], assume that there are matches between signature fingerprints and payload fingerprints. Given the secret length l_s (assuming that the length of l_s is random with uniformly distribution) and the fingerprint length l_p, thus, the cloud provider has no more than $\frac{1}{2^{l_s}\times 2^{l_p}}$ $(0 < l_s < l_p)$ probability of inferring the sensitive information. If there is no match, then the cloud provider should brute force to reverse the Rabin fingerprinting calculation. This brute-force attack is difficult for a polynomial-time adversary [45].

For irreducible polynomials, any degree k is acceptable, but it is more convenience when k is prime, e.g., $k = 17, 19, \ldots$. Based on the previous work on *Rabin fingerprint algorithm* [45], we can have the following knowledge.

Lemma 1. Let k be prime, then the number of irreducible polynomials $p(x) = m_k + m_{k-1}x^{k-1} + m_{k-2}x^{k-2} + \ldots + m_0 \in Z_2[x]$, is $(2_k - 2)/k$.

Proof. Let $GF(2^k) = E$ be the Galois field with 2^k elements. Every irreducible polynomial $p(x) \in Z_2[x]$ of degree k has exactly k roots in E, and since $1 < k$ these roots are in $E - Z_2$.

Lemma 2. For a prime degree k, it is known that a random selection can have an equal probability of an irreducible polynomial $p(t) \in Z_2[x]$ of degree k.

Proof. Let $p_1(t), \ldots, p_d(t)$, $d = (2^k - 2)/k$ be an enumeration of all the different irreducible polynomials of degree k. As described above, $GF(2^k) - Z_2 = S_1 \cup S_2 \cup S_3 \cup \ldots \cup S_D$ where S_i presents the k roots of $p_i(t)$. The random selection $\gamma \in GF(2^k) - Z_2$ has equal probability of falling within each of S_j, because all these options have the same number of elements. In this case, under a probability of $1/d$, we have $\gamma \in S_i$ when $p(t) = p_i(t)$.

Signature (String) matching. Let $\alpha = x_1, x_2, \ldots, x_n$, $\beta = y_1, y_2, \ldots, y_m$, $x_i, y_i \in 0, 1$ denote bit strings. The purpose of string matching is to identify one or more indices i such that $\alpha = y_i y_{i+1} \ldots y_{i+n-1}$. If a match is found for any i, it means that the signature (or rule) α matches the *ith* substring of length n of text β.

Let k be the smallest prime such that $k > log_2(nm\varepsilon^{-1})$. In all practical applications, assume $a(t) = x_1t^{n-1} + \ldots + x_n$, $a_i(t) = y_it^{n-1} + \ldots + y_{i+n-1}$, $1 \leq i \leq m-n+1$. Then, a match for index i is equivalent to $a(t) = a_i(t)$.

Theorem 3. For a set of signatures $\alpha = x_1, x_2, \ldots, x_n$ and a text $\beta = y_1, y_2, \ldots, y_m$, if $k > log_2(nm\varepsilon^{-1})$, then the probability of generating errors is smaller than ε, and all actual matches can be identified.

Proof. Denote $\bar{g}(t)$ as the residue module $p(t)$ of the polynomial $g(t)$. If $\bar{a}(t) = \bar{a}_1(t)$, then the output is 'index 1 is matched'. Thus, all actual matches can be found since $a(t) = a_i(t)$ implies $\bar{a}(t) = \bar{a}_i(t)$ for each $p(t)$.

If there is an error, i.e., the output for some i is 'index i is matched', then we will have $\bar{a}(t) = \bar{a}_i(t)$ and $a(t) \neq a_i(t)$. Thus, $p(t)$ divides $H(t) = \Pi_{a(t)\neq a_i(t)}(a(t) \neq a_i(t))$, shortly $p(t)|H(t)$. Conversely, if for an irreducible polynomial $p(t)|H(t)$, then $p(t)$ have to divide some factors $a(t) = a_i(t)$ of $H(t)$, However, as we have $\bar{a}(t) = \bar{a}_i(t)$ and $a(t) \neq a_i(t)$, the output is wrong when such $p(t)$ is used.

Let $p_1(t), \ldots, p_\varphi(t)$ be a list of irreducible polynomials of degree k that divide $H(t)$. Then their product $P(t)$ can divide $H(t)$ as well. As $H(t)$ is a product of at most $m-n$ factors, we can have $\varphi \leq nm/k$. For each pair of (α,β), at most nm/k irreducible polynomial $p(t)$ of degree k can produce an error.

It is known that the total number of all irreducible polynomials of degree k is $N = (2^k - 2)/k$. Since $k > log_2(nm\varepsilon^{-1})$, we can have $N > nm\varepsilon^{-1}$. For a random selection $p(t)$, the probability of an error output equals the number φ divided by the number of N of all irreducible polynomials of degree k. Then, we can have the the probability of generating errors is smaller than $(nm/k)/(nm\varepsilon^{-1}/k) = \varepsilon$.

5. Evaluation

In this section, we evaluate the performance of our approach in both simulated and real environments, as compared to other similar approaches like PPIDS [44].

5.1. Evaluation in a Simulated Environment

To investigate the performance, we simulated a cloud environment based on iCanCloud[1], which can simulate instance types provided by Amazon. The simulated CIDN consists of 10 nodes. The implementation of Rabin fingerprint is based on cyclic redundancy code and all grams are in 8-byte. The fingerprints are in 128-bit with 129-bit irreducible polynomials, and we set the length (l_s) of the secret s to 64-bit (half length of the fingerprints).

Workload. Fog devices can help perform signature matching for the transmitted traffic from the CIDN layer to the cloud layer, and send the alarms / records to the cloud side. The reduced workload of the central server on cloud side is shown in Fig. 3. It is observed that

[1] `http://icancloudsim.org/Home.html`.

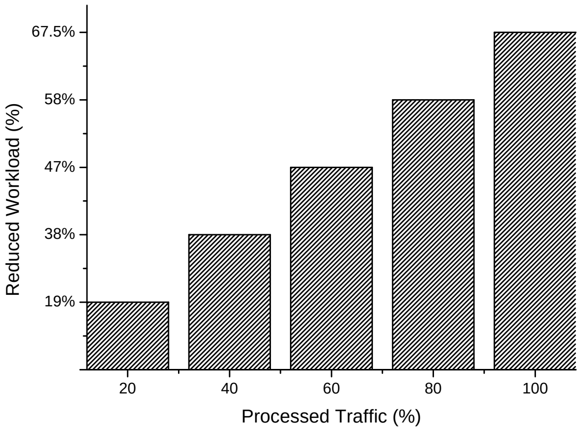


Figure 3: Performance result of reduced workload vs. processed traffic on fog devices.

Table 1: Simulation parameters in the experiment.

Parameters	Value	Description
λ	0.9	Forgetting factor
ε_l	10/day	Low request frequency
ε_h	20/day	High request frequency
r	0.8	Trust threshold
T_s	0.5	Trust value for newcomers
m	10	Lower limit of received feedback
d	0.3	Severity of punishment

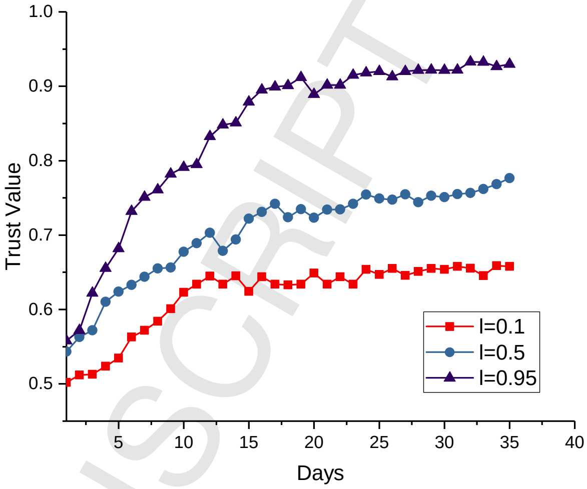


Figure 4: Convergence of nodes' trust values regarding three expertise levels.

with more traffic processed by fog devices, the workload of the central server (in the cloud environment) can be greatly reduced. It is worth noting that the central server still needs to aggregate IDS alarms and correlate information. Overall, our results demonstrate that our proposed framework can help reduce the burden of the central server on cloud side.

Detection delay. Delay always exists for a cloud-based IDSs, as the central server on the cloud has to collect information from different intrusion detectors. In this work, we take an early privacy-preserving intrusion detection approach, called PPIDS [44], in the comparison. More specifically, PPIDS can encrypt the audit log file and detect intrusions by means of a secure computation method without a trusted third party (TTP). The arithmetic operations over encrypted data was accomplished through privacy homomorphism. For implement, their approach was also applied to a signature-based IDSs.

The settings of collaborative intrusion detection is summarized in Table 1. The CIDN environment consists of 25 nodes and each node installed Snort [50] as intrusion detector. Various IDS nodes can communicate necessary information with each other. The initial trust value of every node is set as $T_s = 0.5$.

Node Trust Evaluation. To evaluate the reputation levels of others, an IDS node has to send a challenge under a rate. The trustworthiness of a node can be calculated by identifying the difference between the expected answers and the received feedback. The trustworthiness of a node i according to node j can be computed as below:

$$T_i^j = (w_s \frac{\sum_{k=0}^{n} F_k^{j,i} \lambda^{tk}}{\sum_{k=0}^{n} \lambda^{tk}} - T_s)(1-x)^d + T_s \tag{3}$$

where $F_k^{j,i} \in [0,1]$ describes the satisfaction level of a feedback k, n is the total number of received feedbacks, λ is the *forgetting factor* that gives more weight to recent answers, and w_s is the *significant weight*, which relies on the number of received feedbacks. If the number of received feedbacks is below a minimum threshold m, then $w_s = \frac{\sum_{k=0}^{n} \lambda^{tk}}{m}$; otherwise $w_s = 1$; x is the percentage of "don't know" replies for a time interval; d is a positive incentive parameter to control the punishment severity of "don't know" replies. More details can be referred to [10, 11].

Node Expertise. We adopted three expertise levels for an IDS node: low (0.1), medium (0.5) and high (0.95). Then the expertise can be represented as below:

$$\begin{aligned} f(p'|\alpha,\beta) &= \frac{1}{B(\alpha,\beta)} p'^{\alpha-1}(1-p')^{\beta-1} \\ B(\alpha,\beta) &= \int_0^1 t^{\alpha-1}(1-t)^{\beta-1} dt \end{aligned} \tag{4}$$

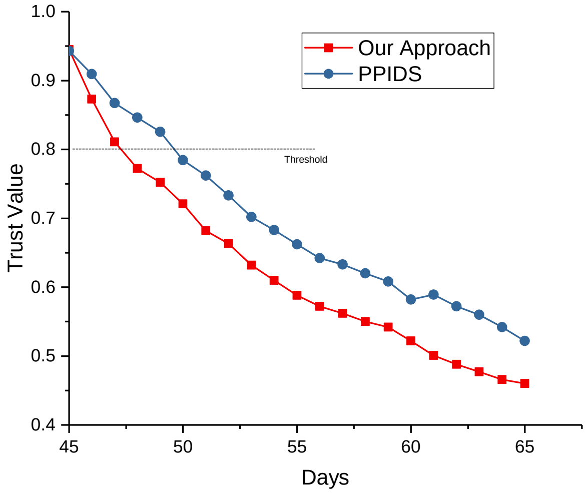


Figure 5: Trust value of malicious node for our approach and PPIDS in the simulated environment.

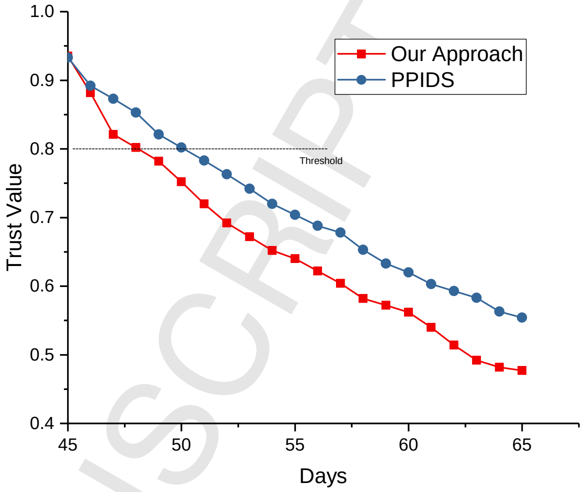


Figure 6: Trust value of malicious node for our approach and PPIDS in the real network environment.

where $p'(\in [0,1])$ describes the probability of an IDS in examining an intrusion, and $f(p'|\alpha,\beta)$ means the probability that an IDS node at the expertise level l responds in p' to an intrusion examination of difficulty level $d(\in [0,1])$. α and β can be modeled as below:

$$\begin{aligned} \alpha &= 1 + \frac{l(1-d)}{d(1-l)}r \\ \beta &= 1 + \frac{l(1-d)}{d(1-l)}(1-r) \end{aligned} \tag{5}$$

where $r \in \{0,1\}$ is the expected result of the detection and $d(\in [0,1])$ is the difficulty level.

Figure 4 shows the convergence of trust values for different expert nodes. It is found that nodes with higher expertise can achieve bigger trust values, which is inline with the observations in former work [10, 11]. The reputation levels become stable after around 25-30 days.

To measure the detection performance of our approach and PPIDS, we randomly chosen one expert node (I = 0.95) to perform an attack from Day 45. The trust value of malicious node is depicted in Figure 5. It is found that under both approaches, the trust value of malicious node would decrease below the threshold, but our approach could reduce the reputation faster than PPIDS. This is because the computation of Rabin fingerprint is faster than the privacy homomorphism used in PPIDS.

5.2. Evaluation in a Real Environment

In this experiment, we collaborated with an IT service provider to validate the performance of our approach in a real network environment. The network deployment is similar to Figure 2 and the implemented CIDN contains 32 nodes with Snort. The basic settings of CIDN can refer to Table 1, while the threshold is the same as 0.8.

Due to some privacy issues, the IT administrator helped implement our approach. The trust values of IDS nodes could become stable after 24 days. Then we randomly selected three high expert nodes to send malicious packets from Day 45, which would activate Snort alarms. The average trust values of malicious nodes under are shown in Figure 6.

Similar to the results obtained in the simulated environment, it is visible that our approach could decrease the trust value of malicious nodes faster than PPIDS. The IT administrator from the participating organization confirmed that the computation time of our approach is faster than PPIDS. Figure 7 depicts the reduced workload of the central server on the cloud side. Similarly, with more traffic processed by fog devices, the workload of the central server in the cloud could be greatly decreased.

Overall, these results demonstrate that our approach can help protect data privacy and reduce the workload on the cloud side. Further, it is validated that our approach may result in less detection delay than other similar approaches like PPIDS. A fast detection is very important to prevent further loss when an intrusion occurred.

5.3. Discussion

In this work, we evaluated our approach in both a simulated and a real network environment, by combin-

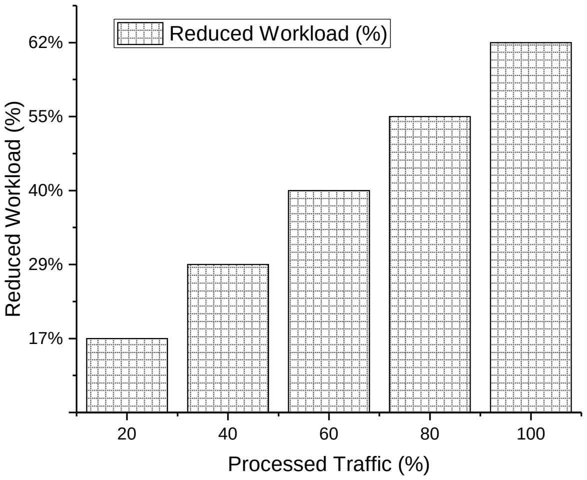


Figure 7: Performance: reduced workload vs. processed traffic on fog devices.

ing CIDN with fog devices. The results showed that our proposed privacy-preserving framework is suitable for signature-based intrusion detection in a distributed environment. However, some challenges may still affect the performance when offload the expensive process of signature matching to the cloud side.

- *Privacy and efficiency.* In any cloud-based mechanism, the data owner has to pre-process all signatures and examined packets. If we consider two parties without a trusted third party, then it is very hard to offload the packet encryption to the cloud provider. While this process may increase the delay locally. This is an open challenge in the field of cloud-based intrusion detection.

- *Detection delay.* For a cloud-based IDS, detection delay is a common challenge, especially when the geographical distribution is broad. Our results demonstrate that fog computing technology is a promising solution to reduce such detection delay; however, some kind of delay still exists. How to design an appropriate encryption methods to further decrease the detection delay is another interesting issue in this area.

- *Adversary model.* In the area of intrusion detection, two adversary model can be considered: *weak adversary model* and *strong adversary model*. The first model assumes that the signature or strings used in the signature matching is non-sensitive, so that users may only require to protect texts (e.g., packet payloads) other than the signature-strings. On the other hand, the second model assumes that the signatures and strings are sensitive data, in case that some organizations may have their self-developed signatures. In this case, a cloud providers can only identify a match, but should not know which plaintext signature is matched. How to address the privacy preserving issue under these two models are one of our future work.

- *Privacy-preserving IDS schemes.* In this paper, we mainly compared our approach with PPIDS [44], which can encrypt the audit log file and detect intrusions by means of a secure computation method without a trusted third party (TTP). PPIDS is the most relevant work based on goals and threat model. There are some other privacy-preserving IDS schemes available, but it is not easy to compare the performance among different schemes. In future, we plan to design a more general scenario and consider more schemes in the evaluation.

6. Conclusion

IDSes have been widely deployed in the Internet to defend against transmissible cyber threats that spread across network domains. To overcome the limitation of processing power in traditional IDS hardware, modern IDSes employ advanced detection algorithms by offloading the process of signature matching to the cloud. However, during the detection, no party wants to disclose their own data especially sensitive data to others. For this sake, privacy-preserving technology has been studied for IDSes, whereas most existing approaches are not suitable for distributed and collaborative intrusion detection due to the geographical distribution. With the advent of fog computing, in this work, we propose a privacy-preserving framework using Rabin fingerprint algorithm for collaborative signature-based intrusion detection by means of fog devices. In the evaluation, we evaluated our approach in both simulated and real network environments. Experimental results demonstrate that our approach can help protect the privacy of data, greatly reduce the workload of the central server on the cloud side, and achieve less detection delay as compared similar approaches like PPIDS.

This is an early study and there are many topics for our future work. For example, it is an interesting topic to design a more lightweight encryption scheme to further decrease the detection delay in distributed environment, and apply our approach to anomaly detection.

Acknowledgments.

This work was partially supported by National Natural Science Foundation of China (No. 61472091), Natural Science Foundation of Guangdong Province for Distinguished Young Scholars (2014A030306020), Science and Technology Planning Project of Guangdong Province, China (2015B010129015) and the Innovation Team Project of Guangdong Universities (No. 2015KCXTD014).

Reference

[1] Wang, Y., Xie, L., Li, W., Meng, W., Li, J.: A Privacy-Preserving Framework for Collaborative Intrusion Detection Networks through Fog Computing. In: Proceedings of The 9th International Symposium on Cyberspace Safety and Security (CSS 2017), pp. 267-279 (2017)

[2] Alabdulatif, A., Kumarage, H., Khalil, I., Yi, X.: Privacy-preserving anomaly detection in cloud with lightweight homomorphic encryption. J. Comput. Syst. Sci. 90, pp. 28-45 (2017)

[3] Alharkan, T., Martin, P.: IDSaaS: Intrusion Detection System as a Service in Public Clouds. In: Proceedings of the 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), pp. 686-687 (2012)

[4] Australia Cyber Threat Report 2017. `https://www.acsc.gov.au/publications/ACSC_Threat_Report_2017.pdf`

[5] Benali, F., Bennani, N., Gianini, G., Cimato, S.: A Distributed and Privacy-Preserving Method for Network Intrusion Detection. In: Proceedings of the 2010 International Conference on On the Move to Meaningful Internet systems (OTM), pp. 861-875 (2010)

[6] Bonomi, F., Milito, R., Zhu, J., Addepalli, S.: Fog computing and its role in the internet of things. In: Proceedings of the first edition of the MCC workshop on Mobile cloud computing (MCC), pp. 13-16 (2012)

[7] Chen, S., Lu, R., Zhang, J.: A flexible privacy-preserving framework for singular value decomposition under internet of things environment. In: Proceedings of International Conference on Trust Management (IFIPTM), pp. 21-37 (2017)

[8] di Vimercati, S.D.C., Foresti, S., Jajodia, S., Paraboschi, S., Samarati, P.: Over-Encryption: Management of Access Control Evolution on Outsourced Data. In: Proceedings of the 33rd International Conference on Very Large Data Bases (VLDB), pp. 123-134 (2007)

[9] Duma, C., Karresand, M., Shahmehri, N., Caronni, G.: A Trust-Aware, P2P-Based Overlay for Intrusion Detection. In: Proccedings of DEXA Workshop, pp. 692-697 (2006)

[10] Fung, C.J., Baysal, O., Zhang, J., Aib, I., Boutaba, R.: Trust Management for Host-Based Collaborative Intrusion Detection. In: Proceedings of DSOM, LNCS 5273, pp. 109-122 (2008)

[11] Fung, C.J., Zhang, J., Aib, I., Boutaba, R.: Robust and scalable trust management for collaborative intrusion detection. In: Proceedings of the 11th IFIP/IEEE International Conference on Symposium on Integrated Network Management (IM), pp. 33-40 (2009)

[12] Guo, J., Marshall, A., Zhou, B.: A New Trust Management Framework for Detecting Malicious and Selfish Behaviour for Mobile Ad Hoc Networks. In: Proceedings of the 10th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 142-149 (2011)

[13] Jiang, W., Wang, G., Bhuiyan, M.Z.A., Wu, J.: Understanding Graph-based Trust Evaluation in Online Social Networks: Methodologies and Challenges, ACM Computing Surveys 49(1), Article 10 (2016)

[14] Kerschbaum, F., Oertel, N.: Privacy-Preserving Pattern Matching for Anomaly Detection in RFID Anti-Counterfeiting. In: Proceedings of the 6th International Conference on Radio Frequency Identification: Security and Privacy Issues (RFIDSec), pp. 124-137 (2010)

[15] Khan, Z.A., Herrmann, P.: A Trust Based Distributed Intrusion Detection Mechanism for Internet of Things. In: Proceedings of the IEEE 31st International Conference on Advanced Information Networking and Applications (AINA), pp. 1169-1176 (2017)

[16] Li, Z., Chen, Y., Beach, A.: Towards Scalable and Robust Distributed Intrusion Alert Fusion with Good Load Balancing. In: Proceedings of the 2006 SIGCOMM Workshop on Large-Scale Attack Defense (LSAD), pp. 115-122, (2006)

[17] Li, W., Meng, Y., Kwok, L.F.: Enhancing Trust Evaluation Using Intrusion Sensitivity in Collaborative Intrusion Detection Networks: Feasibility and Challenges. In: Proceedings of the 9th International Conference on Computational Intelligence and Security (CIS), pp. 518-522 (2013)

[18] Li, W., Meng, W., Kwok, L.F.: Design of Intrusion Sensitivity-Based Trust Management Model for Collaborative Intrusion Detection Networks. In: Proceedings of the 8th IFIP WG 11.11 International Conference on Trust Management (IFIPTM), pp. 61-76 (2014)

[19] Li, W., Meng, Y., Kwok, L.F., Ip, H.H.S.: PMFA: Toward Passive Message Fingerprint Attacks on Challenge-based Collaborative Intrusion Detection Networks. In: Proceedings of the 10th International Conference on Network and System Security (NSS), pp. 433-449 (2016)

[20] Li, W., Meng, Y., Kwok, L.F.: SOOA: Exploring Special On-Off Attacks on Challenge-based Collaborative Intrusion Detection Networks. In: Proceedings of the 12th International Conference on Green, Pervasive and Cloud Computing (GPC 2017), pp. 402-415 (2017)

[21] Li, W., Meng, Y., Kwok, L.F.: Investigating the Influence of Special On-Off Attacks on Challenge-based Collaborative Intrusion Detection Networks. Future Internet, vol. 10, no. 1, pp. 1-16 (2018)

[22] Li, W., Meng, Y., Kwok, L.F., Ip, H.H.S.: Enhancing Collaborative Intrusion Detection Networks Against Insider Attacks Using Supervised Intrusion Sensitivity-Based Trust Management Model. Journal of Network and Computer Applications 77, pp. 135-145 (2017)

[23] Li, P., Li, J., Huang, Z., Gao, C.Z., Chen, W.B., Chen, K.: Privacy-preserving outsourced classification in cloud computing. Cluster Computing, 2017.

[24] Liu, L., Zhu, H., Huang, Z., Xie, D.: Minimal Privacy Authorization in Web Services Collaboration. Computer Standards & Interfaces, 33(3), pp. 332-343 (2011)

[25] Liu, Q., Wang, G., Li, F., Yang, S., Wu, J.: Preserving Privacy with Probabilistic Indistinguishability in Weighted Social Networks, IEEE Transactions on Parallel and Distributed Systems 28(5), pp. 1417-1429 (2017)

[26] Luo, E., Liu, Q., Abawajy, J.H., Wang, G.: Privacy-Preserving Multi-Hop Profile-Matching Protocol for Proximity Mobile Social Networks, Future Generation Computer Systems 68, pp. 222-223 (2017)

[27] Meng, Y., Li, W., Kwok, L.F., Xiang, Y.: Towards Designing Privacy-Preserving Signature-based IDS As a Service: A Study and Practice. In: Proceedings of the 5th IEEE International Conference on Intelligent Networking and Collaborative

Systems (INCoS), pp. 181-188 (2013)

[28] Meng, Y., Kwok, L.F., Li, W.: Towards Designing Packet Filter with a Trust-Based Approach Using Bayesian Inference in Network Intrusion Detection. In: Proceedings of SecureComm 2012, pp. 203-221, 2012.

[29] Meng, Y., Kwok, L.F., Li, W.: Enhancing False Alarm Reduction Using Voted Ensemble Selection in Intrusion Detection. International Journal of Computational Intelligence Systems 6(4), pp. 626-638 (2013)

[30] Meng, Y., Li, W., Kwok, L.F.: Evaluation of Detecting Malicious Nodes Using Bayesian Model in Wireless Intrusion Detection. In: Proceedings of NSS, pp. 40-53 (2013)

[31] Meng, Y., Kwok, L.F.: Adaptive Non-Critical Alarm Reduction Using Hash-based Contextual Signatures in Intrusion Detection. Computer Communications, vol. 38, pp. 50-59 (2014)

[32] Meng, W., Li, W., Kwok, L.F.: EFM: Enhancing the Performance of Signature-based Network Intrusion Detection Systems Using Enhanced Filter Mechanism. Computers & Security 43, pp. 189-204 (2014)

[33] Meng, Y., Kwok, L.F.: Adaptive Blacklist-based Packet Filter with A Statistic-based Approach in Network Intrusion Detection. Journal of Network and Computer Applications 39, pp. 83-92 (2014)

[34] Meng, W., Li, W., Kwok, L.F.: Design of Intelligent KNN-based Alarm Filter Using Knowledge-based Alert Verification in Intrusion Detection. Security and Communication Networks 8(18), pp. 3883-3895 (2015)

[35] Meng, W., Luo, X., Li, W., Li, Y.: Design and Evaluation of Advanced Collusion Attacks on Collaborative Intrusion Detection Networks in Practice. In: Proceedings of the 15th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 1061-1068 (2016)

[36] Meng, W., Li, W., Xiang, Y., Choo, K.K.R.: A Bayesian Inference-based Detection Mechanism to Defend Medical Smartphone Networks Against Insider Attacks. Journal of Network and Computer Applications 78, pp. 162-169 (2017)

[37] Meng, W., Li, W., Wang, Y., Au, M.N.: Detecting Malicious Nodes in Medical Smartphone Networks through Euclidean Distance-based Behavioral Profiling. In: Proceedings of the 9th International Symposium on Cyberspace Safety and Security (CSS), pp. 163-175 (2017)

[38] Meng, W., Jiang, L., Wang, Y., Li, J., Zhang, J., Xiang, Y.: JFCGuard: Detecting Juice Filming Charging Attack via Processor Usage Analysis on Smartphones. Computers & Security (2018)

[39] Meng, W., Tischhauser, E.W., Wang, Q., Wang, Y., Han, J.: When Intrusion Detection Meets Blockchain Technology: A Review. IEEE Access, vol. 6, no. 1, pp. 10179-10188 (2018)

[40] Meng, W., Li, W., Su, C., Zhou, J., Lu, R.: Enhancing Trust Management for Wireless Intrusion Detection via Traffic Sampling in the Era of Big Data. IEEE Access, vol. 6, no. 1, pp. 7234-7243 (2018)

[41] Meng, W., Choo, K.K.R., Furnell, S., Vasilakos, A.V., Probst, C.W.: Towards Bayesian-based Trust Management for Insider Attacks in Healthcare Software-Defined Networks. IEEE Transactions on Network and Service Management, In Press, IEEE (2018)

[42] Mishra, P., Pilli, E.S., Varadharajan, V., Tupakula, U.K.: Intrusion detection techniques in cloud environment: A survey. Journal of Network and Computer Applications 77, pp. 18-47 (2017)

[43] Niksefat, S., Sadeghiyan, B., Mohassel, P., Sadeghian, S.S.: ZIDS: A Privacy-Preserving Intrusion Detection System Using Secure Two-Party Computation Protocols. Computer Journal 57(4), pp. 494-509 (2014)

[44] Park, H.A., Lee, D.H., Lim, J., Cho, S.H.: PPIDS: Privacy Preserving Intrusion Detection System. In: Proceedings of PAISI 2007, pp. 269-274 (2007)

[45] Rabin, M.O.: Fingerprinting by Random Polynomials. Center for Research in Computing Technology, Harvard University. Technical Report TR-CSE-03-01 (1981)

[46] Roesch, M.: Snort: Lightweight Intrusion Detection for Networks. In: Proceedings of the 1999 Usenix Lisa Conference, pp. 229-238 (1999)

[47] Ruan, X., Wu, Z., Wang, H., Jajodia, S.: Profiling Online Social Behaviors for Compromised Account Detection. IEEE Transactions on Information Forensics and Security 11(1), pp. 176-187 (2016)

[48] Shaikh, R.A., Jameel, H., d'Auriol, B.J., Lee, H., Lee, S., Song, Y.J.: Group-based Trust Management Scheme for Clustered Wireless Sensor Networks. IEEE Transactions on Parallel and Distributed Systems 20(11), pp. 1698-1712 (2009)

[49] Scarfone, K., Mell, P.: Guide to Intrusion Detection and Prevention Systems (IDPS). NIST Special Publication 800-94 (Feb 2007)

[50] Snort: An Open Source Network Intrusion Prevention and Detection System (IDS/IPS). Homepage. Available online: `http://www.snort.org/`.

[51] Wang, Y., Cai, Z., Tong, X., Gao, Y., Yin, G.: Truthful incentive mechanism with location privacy-preserving for mobile crowdsourcing systems. Computer Networks, 135, pp. 32-43 (2018)

[52] Wu, Y.-S., Foo, B., Mei, Y., Bagchi, S.: Collaborative Intrusion Detection System (CIDS): A Framework for Accurate and Efficient IDS. In: Proceedings of the 2003 Annual Computer Security Applications Conference (ACSAC), pp. 234-244 (2003)

[53] Yassin, W., Udzir, N.I., Muda, Z., Abdullah, A., Abdullah, M.T.: A Cloud-based Intrusion Detection Service framework. In: Proceedings of the 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic, pp. 213-218 (2012)

[54] Ye, A., Chen, Q., Xu, L., Wu, W.: The flexible and privacy-preserving proximity detection in mobile social network. Future Generation Comp. Syst. 79, pp. 271-283 (2018)

[55] Zhang, P., Huang, X., Sun, X., Wang, H., Ma, Y.: Privacy-Preserving Anomaly Detection across Multi-Domain Networks. In: Proceedings of the 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), pp. 1066-1070 (2012)

[56] Zhang, J., Xiang, Y., Wang, Y., Zhou, W., Xiang, Y., Guan, Y.: Network traffic classification using correlation information. IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 1, pp. 104-117 (2013)

[57] Zhang, T., Zhu, Q.: Distributed Privacy-Preserving Collaborative Intrusion Detection Systems for VANETs. IEEE Trans. Signal and Information Processing over Networks 4(1), pp. 148-161 (2018)

[58] Zhou, T., Choudhury, R.R., Ning, P., Chakrabarty, K.: Privacy-Preserving Detection of Sybil Attacks in Vehicular Ad Hoc Networks. In: Proceedings of the Fourth Annual International Conference on Mobile and Ubiquitous Systems: Networking & Services (MobiQuitous), pp. 1-8 (2007)

[59] Chen, S., Wang, G., Yan, G., Xie, D.: Multi-dimensional fuzzy trust evaluation for mobile social networks based on dynamic community structures. Concurrency and Computation: Practice and Experience 29(7), e3901. DOI:10.1002/cpe.3901.

[60] Li, P., Li, J., Huang, Z., Li, T., Gao, C.Z., Yiu, S.M., Chen, K.: Multi-key privacy-preserving deep learning in cloud computing. Future Generation Computer Systems 74, 7685 (2017).

[61] Li, J., Zhang, Y., Chen, X., Xiang, Y.: Secure attribute-based data sharing for resource-limited users in cloud computing. Computers Security 72, 1 12 (2018).

[62] Gao, C.z., Cheng, Q., Li, X., Xia, S.b.: Cloud-assisted privacy-preserving profile-matching scheme under multiple keys in mobile social network. Cluster Computing (2018). DOI 10.1007/s10586-017-1649-y.

[63] Gao, C.z., Cheng, Q., He, P., Susilo, W., Li, J.: Privacy-preserving naive bayes classifiers secure against the substitution-then-comparison attack. Information Sciences 444, 72 88 (2018).

[64] Tao Peng, Qin Liu, Dacheng Meng, Guojun Wang. Collaborative Trajectory Privacy Preserving Scheme in Location-Based Services. Information Sciences, 387(2017):165-179.

[65] Cai, J., Wang, Y., Liu, Y., Luo, J.Z., Wei, W., Xu, X.: Enhancing network capacity by weakening community structure in scale-free network. Future Generation Computer Systems (2017). DOI:10.1016/j.future.2017.08.014.

[66] Liu, Q., Wang, G., Liu, X., Peng, T., Wu, J.: Achieving reliable and secure services in cloud computing environments. Computers Electrical Engineering 59, 153 164 (2017).

[67] Li, J., Sun, L., Yan, Q., Li, Z., Srisa-an, W., Ye, H.: Significant permission identification for machine learning based android malware detection. IEEE Transactions on Industrial Informatics pp. 11 (2018). DOI: 10.1109/TII.2017.2789219.

***Author Biography & Photograph**

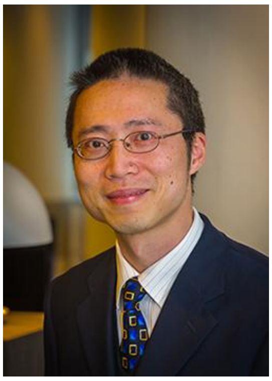
Yu Wang received his Ph.D. degree in computer science from Deakin University, Victoria, Australia. He is currently an associate professor with the School of Computer Science, Guangzhou University, China. His research interests include network traffic analysis, mobile networks, social networks, and cyber security.

Weizhi Meng is currently an assistant professor in the Department of Applied Mathematics and Computer Science, Technical University of Denmark (DTU), Kongens Lyngby, Denmark. He obtained his Ph.D. degree in Computer Science from the City University of Hong Kong (CityU), Hong Kong in 2013. He won the Outstanding Academic Performance Award during his doctoral study, and is a recipient of the Hong Kong Institution of Engineers (HKIE) Outstanding Paper Award for Young Engineers/Researchers in both 2014 and 2017. He was known as Yuxin Meng and prior to joining DTU, he worked as a research scientist in Infocomm Security (ICS) Department, Institute for Infocomm Research, Singapore. His primary research interests include intrusion detection, mobile security, biometric authentication, HCI security, cloud security, trust computation, and vulnerability analysis. He is a member of ACM and IEEE.

Wenjuan Li is currently a Ph.D. student in the Department of Computer Science, City University of Hong Kong (CityU) and is holding an exchange position at Technical University of Denmark (DTU), Prior to this, she worked as a Research Assistant in CityU from 2013 to 2014, and was previously a Lecturer in the Department of Computer Science, Zhaoqing Foreign Language College, China. She was a Winner of Cyber Quiz and Computer Security Competition, Final Round of Kaspersky Lab "Cyber Security for the Next Generation" Conference in 2014. Her research interests include network management and security, collaborative intrusion detection, spam detection, trust computing, web technology and E-commerce technology.

Jin Li received his B.S. degree (2002) in Mathematics from Southwest University and the Ph.D. degree in Information Security from Sun Yat-sen University in 2007. Currently, he is a professor at Guangzhou University. He has been selected as one of the science and technology new stars in Guangdong province. His research interests include security in cloud computing and applied cryptography. He has published over 80 research papers in refereed international conferences and journals, and has served as the Program Chair or Program Committee Member in many international conferences.

Wai-Xi Liu received the Ph.D. degree in communication and information system from Sun Yat-Sen University, China, in 2013. He is currently an associate professor in the Department of Electronic and Information Engineering, Guangzhou University, China. His research interests are in future network, data mining and Network coding. He has published more than 20 papers.



Yang Xiang received his PhD in Computer Science from Deakin University, Australia. He is the Dean of Digital Research & Innovation Capability Platform, Swinburne University of Technology, Australia. His research interests include cyber security, which covers network and system security, data analytics, distributed systems, and networking. In particular, he is currently leading his team developing active defense systems against large-scale distributed network attacks. He has published more than 200 research papers in many international journals and conferences. He is the Coordinator, Asia for IEEE Computer Society Technical Committee on Distributed Processing (TCDP). He is a Senior Member of the IEEE.