A Privacy-Preserving Framework for Collaborative Intrusion Detection Networks Through Fog Computing

Yu Wang[1], Lin Xie[3], Wenjuan Li[2,3], Weizhi Meng[2(✉)], and Jin Li[1]

[1] School of Computer Science, Guangzhou University, Guangzhou, China
yuwang@gzhu.edu.cn
[2] Department of Applied Mathematics and Computer Science, Technical University of Denmark, Lyngby, Denmark
weme@dtu.dk
[3] Department of Computer Science, City University of Hong Kong, Kowloon Tong, Hong Kong

Abstract. Nowadays, cyber threats (e.g., intrusions) are distributed across various networks with the dispersed networking resources. Intrusion detection systems (IDSs) have already become an essential solution to defend against a large amount of attacks. With the development of cloud computing, a modern IDS is able to implement more complicated detection algorithms by offloading the expensive operations such as the process of signature matching to the cloud (i.e., utilizing computing resources from the cloud). However, during the detection process, no party wants to disclose their own data especially sensitive information to others for privacy concerns, even to the cloud side. For this sake, privacy-preserving technology has been applied to IDSs, while it still lacks of proper solutions for a collaborative intrusion detection network (CIDN) due to geographical distribution. A CIDN enables a set of dispersed IDS nodes to exchange required information. With the advent of fog computing, in this paper, we propose a privacy-preserving framework for collaborative networks based on fog devices. Our study shows that the proposed framework can help reduce the workload on cloud's side.

Keywords: Collaborate network · Privacy preserving · Intrusion detection · Cloud environment · Fog computing

1 Introduction

Cyber threats (e.g., virus, denial-of-service (DoS) attack) are a big issue for current computer networks. To defend against various cyber attacks, intrusion detection systems (IDSs) have been widely deployed in organizations. Based on the detection approaches, an IDS can be roughly classified as signature-based

W. Meng—The author was previously known as Yuxin Meng.

S. Wen et al. (Eds.): CSS 2017, LNCS 10581, pp. 267–279, 2017.
https://doi.org/10.1007/978-3-319-69471-9_20

IDS and anomaly-based IDS [31]. The former (e.g., Snort [28]) detects an intrusion through comparing the packet payload with stored signatures (i.e., which describes a known attack or exploit), while the latter uses a pre-defined threshold and identifies an anomaly by comparing current profile with normal profile (i.e., which describes normal status of a network or system). Additionally, an IDS can be categorized as network-based IDS (NIDS) and host-based IDS (HIDS) in terms of the deployed locations.

With the increasing traffic volumes, it is hard for a traditional IDS to handle large incoming traffic. For example, the processing burden of a signature-based IDS is at least linear to the size of an input string [4]. With the development of cloud environment, an IDS can reduce its burden by offloading expensive operations to such computing infrastructures. For instance, Alharkan and Martin [1] proposed *IDSaaS* in Amazon EC2 cloud, which could monitor and record malicious network behaviors between virtual machines and users within a Virtual Private Cloud. Yassin et al. [33] proposed *CBIDS*, a Cloud-based Intrusion Detection Service Framework (CBIDS) to monitor different layers' traffic and detect unexpected activities from different points of a network.

In practical usage, an IDS should upload its traffic to the cloud side for inspection, which leads to threaten users' privacy. For example, cloud service provider may passively monitor users' log information to improve some of their services, but users may not want their log information to be monitored. As a result, privacy-preserving technology is widely applied to current IDSs. As an example, Park et al. [26] proposed *PPIDS*, a privacy preserving method for IDSs by applying cryptographic approaches to log files without a trusted third party (TTP). This system can encrypt the audit log file and identify intrusions over encrypted data.

Motivations. However, current cloud-based IDS is not suitable for a distributed IDS infrastructure due to its geographical distribution. Collaborative intrusion detection networks (CIDNs) enable a set of IDS nodes to collect and exchange information with each other [32]. If all data is uploaded to a cloud server for computation, it would consume considerable communication and computing resources, which makes a negative impact on the quality of service (QoS) (i.e., dealing with many redundant data). To further 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/nodes), which can support for mobility, heterogeneity, interoperability and pre-processing.

Contributions. As fog computing can provide 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. In this paper, we thus propose a privacy-preserving framework for CIDNs based on fog devices. The contributions of our work 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 CIDNs based on fog computing. The fog computing can provide storage, computing and networking services between an IDS and a cloud. With the equipped resource, fog devices could loose the workload of a cloud server.
– As a study, we apply Rabin fingerprint algorithm to our proposed framework, and evaluate our approach in a simulated environment. The experimental results show that our framework can help reduce the workload of a central server on the cloud.

Organization. The rest of this paper is organized as follows. In Sect. 2, we introduce the background of collaborative intrusion detection networks. Section 3 describes our proposed privacy-preserving framework and Sect. 4 shows a study and performance results. Section 5 introduces related work and Sect. 6 concludes our paper.

2 Background of CIDNs

This section briefly introduces the background of collaborative intrusion detection networks (CIDNs). As a CIDN is vulnerable to insider attacks, trust computation and evaluation is essential within such network [18,20,25]. This section takes challenge-based CIDNs as an example, describing its major components and explaining how it works.

Major components. In addition to a detection engine, each node in a CIDN usually 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. Regarding challenge-based CIDNs, the trustworthiness of target nodes is mainly computed by evaluating the received feedback. Each node can send out either normal requests or challenges for alert 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.
– *Collaboration component.* This component is mainly responsible for assisting a node to evaluate the trustworthiness of others by sending out *normal requests* or *challenges*, and receiving the relevant *feedback*. If a tested IDS node receives a request or challenge, this component will help send back its feedback. As shown in Fig. 1, if node A sends a *request/challenge* to node B, then node B will send back relevant feedback.
– *P2P communication.* This component is responsible for connecting with other IDS nodes and providing network organization, management and communication among IDS nodes.

Network Interactions. In a CIDN, each IDS node can choose its partners or collaborators based on its own policies and experience. These IDS nodes can be

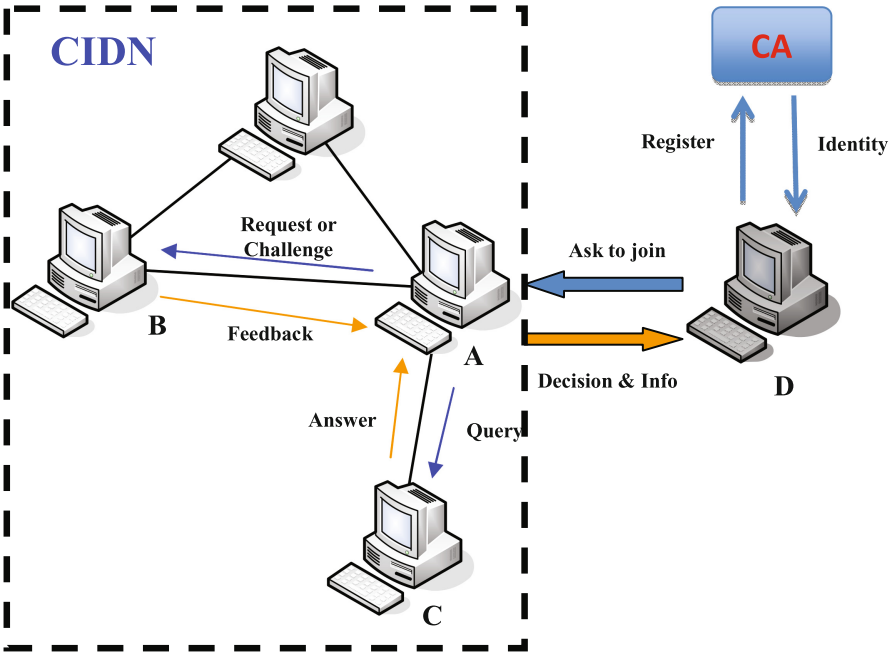

Fig. 1. The high-level architecture of a typical challenge-based CIDN.

associated if they have a collaborative relationship. Each node can maintain a list of their collaborated nodes, called *partner list* (or *acquaintance list*). Such list is customizable and stores information of other nodes (e.g., public keys and their current trust values). Before a node asks for joining the network, it has to register to a trusted certificate authority (*CA*) and obtain its unique proof of identity (e.g., a public key and a private key). As shown in Fig. 1, if node *D* wants to join the network, it needs to send an application to a network node, say node *A*. Then, node *A* makes a decision and sends back an initial *partner list*, if node *D* is accepted.

CIDNs allow IDS nodes exchanging required messages in-between to enhance their performance. 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, 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 alarm ranking information as their feedback. Alarm aggregation is an important feature for CIDNs, which can help improve the detection performance, and it usually considers the feedback from trusted nodes.

3 Our Approach

This section introduces the concept of fog computing and details our proposed privacy-preserving framework for collaborative intrusion detection.

3.1 Fog Computing

Fog computing is proposed by Cisco, which aims to help ease the burden of the IoT server and safeguard the QoS [3]. As cloud computing does not need the enterprise and the end user to know specification or many details, it bliss 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 resource. With these resources, fog devices can process the collected data locally, in order to ease the burden of the cloud side (e.g., a central server). For example, the fog devices can perform some specific operations on the received data and send the results to the central server. In this case, the volume of data sent to the server could be reduced to a large extend.

3.2 Our Proposed Framework

Due to the features of fog computing, it is suitable for distributed intrusion detection architectures. Figure 2 depicts our proposed privacy-preserving framework for CIDNs based on fog devices. There are totally three layers:

- *CIDN layer.* This is the normal collaborative network layer, where different IDS nodes can improve their detection performance by exchanging required information with each other. Some expensive operations (e.g., signature matching) and sensitive information (e.g., logs) could be offloaded to the cloud side (cloud layer).
- *Cloud layer.* The cloud environment can provide sufficient computation resources for the 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, depending on the geographical locations.
- *Fog layer.* The fog layer often embodies software modules and embedded operating systems. This layer is able 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 thus to provide quick responses to unusual behaviors.

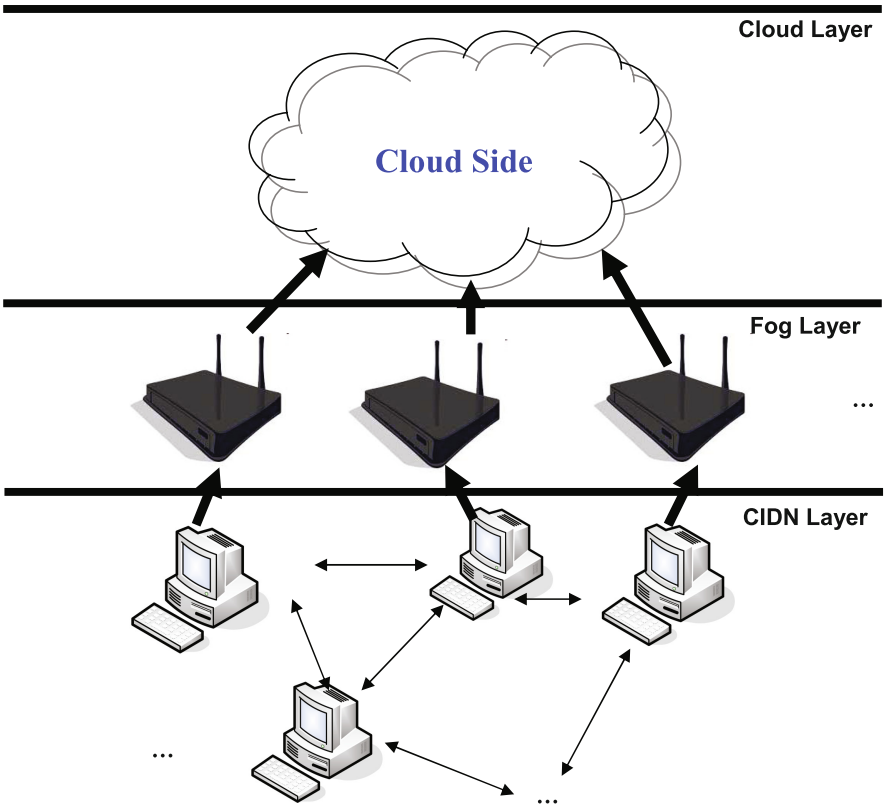

Fig. 2. Our privacy-preserving framework for CIDNs using fog devices.

4 Study and Evaluation

In this section, we detail the threat model, present a study of applying Rabin fingerprint algorithm to our framework, and illustrates performance results.

4.1 Threat Model

This work adopts the *curious-but-honest model* for a cloud provider [5]. That is, the cloud provider is trustful to follow the agreed protocol and perform intrusion inspection (i.e., analyzing network traffic for an organization); however, the cloud provider attempts to monitor, store, and learn the information about the sensitive (or private) data from the examined traffic, or attempt to discover anything they are interested in.

4.2 Fingerprint Computation for Signature-Based IDS

To facilitate comparison with [17], we conduct a study by applying *Rabin fingerprint algorithm* [27] to our framework. Rabin fingerprints can be computed

using polynomial modulus operations with fast XOR, shift and table look-up operations. It has two merits: (1) one way; and (2) fast computation. For a signature-based IDSs, these fingerprints can be applied to the process of signature matching for 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:

$$f(x) = m_k + m_{k-1}x + m_{k-2}x^2 + ... + m_1x^{k-1} mod \;\; p(x) \tag{1}$$

Based on Eq. (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 [17] 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 still 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:

$$f'(x) = f(x) \oplus s \;\; (0 < l_s < |f(x)|) \tag{2}$$

4.3 Performance Results

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).

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's side is shown in Fig. 3. It is observed that 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's side.

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

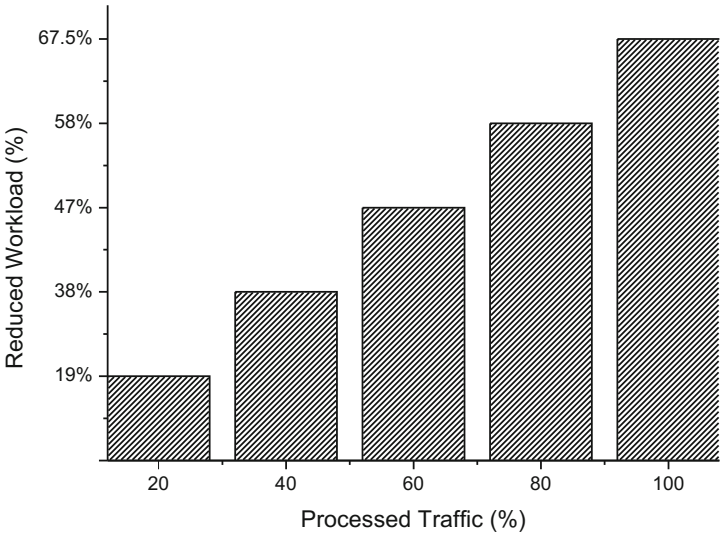

Fig. 3. Performance result of reduced workload vs. processed traffic on fog devices.

Security discussion. In current IDS scenario, we consider signatures are sensitive as well, and we can use Eq. (2) to perturb the fingerprints. According to [17], 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 [27].

5 Related Work

This section introduces related work about distributed intrusion detection systems, challenge-based CIDNs and privacy-preserving IDSs.

Distributed trust-based intrusion detection. Collaborative intrusion detection networks (CIDNs) [32] can enable an IDS node to achieve better detection performance by collecting and communicating information with other IDS nodes.

Li *et al.* [11] identified that most distributed intrusion detection systems (DIDS) might rely on centralized fusion, or distributed fusion with unscalable communication mechanisms. Based on this, they proposed a DIDS according to 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). To detect insider attacks, Duma *et al.* [6] proposed a P2P-based overlay for intrusion

detection (Overlay IDS) that mitigated the insider threat by using a trust-aware engine for correlating alerts and an adaptive scheme for managing trust. The trust-aware correlation engine is capable of filtering out warnings sent by untrusted or low quality peers, while the adaptive trust management scheme uses past experiences of peers to predict their trustworthiness.

Similarly, Shaikh *et al.* [30] 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.* [9] 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.

Challenge-based intrusion detection. Challenge-based mechanism is a special way of computing trust for IDSs, where the trustworthiness of a node depends on the received answers to the challenges. Fung *et al.* [7] proposed a HIDS collaboration framework that enables each HIDS to evaluate the trustworthiness of others based on its own experience by means of a forgetting factor. The forgetting factor can give more emphasis on the recent experience of the peer. Then, they improved their trust management model by using a Dirichlet-based model to measure the level of trustworthiness among IDS nodes according to their mutual experience [8]. This model had strong scalability properties and was robust against common insider threats. Experimental results demonstrated that the new model could improve robustness and efficiency.

To further enhance the performance of CIDNs, Li *et al.* [12] identified that different IDSs may have distinct levels of sensitivity in detecting particular types of threats based on their own signatures and profiles. They thus defined a concept of *intrusion sensitivity* and explored its feasibility 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 [13], and proposed a machine learning-based approach in automatically allocating the values of *intrusion sensitivity* [16].

On the other hand, Li *et al.* [14] 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 under PMFA, malicious nodes can send malicious responses to normal requests while maintaining their trust values. A special On-Off attack (called SOOA) was also developed by them, which could keep responding normally to one node while acting abnormally to another node [15]. As a result, there is still a need to enhance the security of CIDN frameworks [24], i.e., considering behavior profile [29]. Other related studies on improving IDSs can be referred to alert reduction [19], alert verification [22,23] and EFM [21].

IDS and privacy-preserving techniques. A number of privacy-preserving schemes are developed for protecting data privacy during data sharing and intrusion detection. For example, Park *et al.* [26] proposed *PPIDS*, a privacy preserving approach for an IDS through 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.* [2] identified and discussed some privacy issues. For example, when several organizations decided to collaborate in identifying intrusive activities, every organization resource manager was requested to send the events log to a central unit. As a result, such central unit was supposed to act as a trusted entity. Indeed, 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.

Zhou *et al.* [35] proposed a framework to detect Sybil attacks, while preserving the privacy of users in vehicular ad hoc networks. The framework could distribute the responsibility of detecting Sybil attacks to semi-trusted third parties. Kerschbaum and Oertel [10] 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.* [34] 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.

6 Conclusion

Intrusion detection systems are an important solution to defend against cyber attacks. With the help of cloud computing, a modern IDS is able to deploy advanced detection algorithms by offloading the expensive operations like the process of signature matching to the cloud side. However, during the detection, no party wants to disclose their own data especially sensitive data to others. For this sake, privacy-preserving intrusion detection technology has received much attention, while most current approaches are not suitable for collaborative intrusion detection networks (CIDNs) due to its geographical distribution. With the advent of fog computing, in this paper, we propose a privacy-preserving framework for CIDNs based on fog devices. In our study, we apply Rabin fingerprint algorithm to our framework, and found that our approach can greatly reduce the workload of the central server on cloud's side.

This is an early study in this direction, and there are many topics for our future work. One is to apply our proposed framework to a real network environment and investigate the detection performance. It is also an interesting topic

to analyze accuracy, privacy and efficiency of the proposed framework for an anomaly-based IDS.

Acknowledgment. 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).

References

1. 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)
2. Benali, F., Bennani, N., Gianini, G., Cimato, S.: A distributed and privacy-preserving method for network intrusion detection. In: Meersman, R., Dillon, T., Herrero, P. (eds.) OTM 2010. LNCS, vol. 6427, pp. 861–875. Springer, Heidelberg (2010). doi:10.1007/978-3-642-16949-6_13
3. 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)
4. Dreger, H., Feldmann, A., Paxson, V., Sommer, R.: Operational experiences with high-volume network intrusion detection. In: Proceedings of the 11th ACM Conference on Computer and Communications Security (CCS), pp. 2–11 (2004)
5. 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)
6. 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)
7. Fung, C.J., Baysal, O., Zhang, J., Aib, I., Boutaba, R.: Trust management for host-based collaborative intrusion detection. In: Turck, F., Kellerer, W., Kormentzas, G. (eds.) DSOM 2008. LNCS, vol. 5273, pp. 109–122. Springer, Heidelberg (2008). doi:10.1007/978-3-540-87353-2_9
8. 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)
9. 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)
10. Kerschbaum, F., Oertel, N.: Privacy-preserving pattern matching for anomaly detection in RFID anti-counterfeiting. In: Ors Yalcin, S.B. (ed.) RFIDSec 2010. LNCS, vol. 6370, pp. 124–137. Springer, Heidelberg (2010). doi:10.1007/978-3-642-16822-2_12
11. 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)

12. 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)
13. Li, W., Meng, W., Kwok, L.-F.: Design of intrusion sensitivity-based trust management model for collaborative intrusion detection networks. In: Zhou, J., Gal-Oz, N., Zhang, J., Gudes, E. (eds.) IFIPTM 2014. IAICT, vol. 430, pp. 61–76. Springer, Heidelberg (2014). doi:10.1007/978-3-662-43813-8_5
14. Li, W., Meng, W., Kwok, L.-F., Ip, H.H.S.: PMFA: toward passive message fingerprint attacks on challenge-based collaborative intrusion detection networks. In: Chen, J., Piuri, V., Su, C., Yung, M. (eds.) NSS 2016. LNCS, vol. 9955, pp. 433–449. Springer, Cham (2016). doi:10.1007/978-3-319-46298-1_28
15. Li, W., Meng, W., Kwok, L.-F.: SOOA: exploring special on-off attacks on challenge-based collaborative intrusion detection networks. In: Au, M.H.A., Castiglione, A., Choo, K.-K.R., Palmieri, F., Li, K.-C. (eds.) GPC 2017. LNCS, vol. 10232, pp. 402–415. Springer, Cham (2017). doi:10.1007/978-3-319-57186-7_30
16. 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. J. Netw. Comput. Appl. **77**, 135–145 (2017)
17. 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)
18. Meng, Y., Kwok, L.-F., Li, W.: Towards designing packet filter with a trust-based approach using bayesian inference in network intrusion detection. In: Keromytis, A.D., Pietro, R. (eds.) SecureComm 2012. LNICSSITE, vol. 106, pp. 203–221. Springer, Heidelberg (2013). doi:10.1007/978-3-642-36883-7_13
19. Meng, Y., Kwok, L.F., Li, W.: Enhancing false alarm reduction using voted ensemble selection in intrusion detection. Int. J. Comput. Intell. Syst. **6**(4), 626–638 (2013)
20. Meng, Y., Li, W., Kwok, L.: Evaluation of detecting malicious nodes using bayesian model in wireless intrusion detection. In: Lopez, J., Huang, X., Sandhu, R. (eds.) NSS 2013. LNCS, vol. 7873, pp. 40–53. Springer, Heidelberg (2013). doi:10.1007/978-3-642-38631-2_4
21. Meng, W., Li, W., Kwok, L.F.: EFM: enhancing the performance of signature-based network intrusion detection systems using enhanced filter mechanism. Comput. Secur. **43**, 189–204 (2014)
22. Meng, Y., Kwok, L.F.: Adaptive blacklist-based packet filter with a statistic-based approach in network intrusion detection. J. Netw. Comput. Appl. **39**, 83–92 (2014)
23. Meng, W., Li, W., Kwok, L.F.: Design of intelligent KNN-based alarm filter using knowledge-based alert verification in intrusion detection. Secur. Commun. Netw. **8**(18), 3883–3895 (2015)
24. 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)
25. Meng, W., Li, W., Xiang, Y., Choo, K.K.R.: A Bayesian inference-based detection mechanism to defend medical smartphone networks against insider attacks. J. Netw. Comput. Appl. **78**, 162–169 (2017)

26. Park, H.-A., Lee, D.H., Lim, J., Cho, S.H.: PPIDS: privacy preserving intrusion detection system. In: Yang, C.C., Zeng, D., Chau, M., Chang, K., Yang, Q., Cheng, X., Wang, J., Wang, F.-Y., Chen, H. (eds.) PAISI 2007. LNCS, vol. 4430, pp. 269–274. Springer, Heidelberg (2007). doi:10.1007/978-3-540-71549-8_27
27. Rabin, M.O.: Fingerprinting by Random Polynomials. Center for Research in Computing Technology, Harvard University. Technical Report TR-CSE-03-01 (1981)
28. Roesch, M.: Snort: lightweight intrusion detection for networks. In: Proceedings of the 1999 Usenix Lisa Conference, pp. 229–238 (1999)
29. Ruan, X., Wu, Z., Wang, H., Jajodia, S.: Profiling online social behaviors for compromised account detection. IEEE Trans. Inf. Forensics Secur. **11**(1), 176–187 (2016)
30. 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 Trans. Parallel Distrib. Syst. **20**(11), 1698–1712 (2009)
31. Scarfone, K., Mell, P.: Guide to Intrusion Detection and Prevention Systems (IDPS). NIST Special Publication 800–94, February 2007
32. 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)
33. 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)
34. 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)
35. 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)