A Novel Scheme for Malicious Nodes Detection in Cloud Markets Based on Fuzzy Logic Technique

https://doi.org/10.3991/ijim.v16i03.27933

Ayoub Alsarhan[1(✉)], Abdel-Rahman Al-Ghuwairi[2], Esra'a Alshdaifat[1], Hasan Idhaim[3], Omar alkhawaldeh[4]
[1]Department of Information Technology, Hashemite University, Zarqa, Jordan
[2]Software Engineering Department, Hashemite University, Zarqa, Jordan
[3]Computer Information Systems Department, Hashemite University, Zarqa, Jordan
[4]Hashemite University, Zarqa, Jordan
ayoubm@hu.edu.jo

Abstract—Cloud security vulnerabilities have recently become more prevalent around the world, posing a threat to cloud service providers' (CSPs) ability to respond to client demands. In cloud market, the requests are announced by the client nodes to their CSP. A malicious node can alter a client's request, resulting in the next cloud market collapse, decreased reliability, and data leaking. To identify malicious nodes in the cloud market, a novel fuzzy multiple criterion decision making scheme is suggested. Authentication test, trust level, traffic size, and node activity levels are all taken into consideration simultaneously as the major criteria for identifying malicious nodes. For each node, the CSP uses fuzzy Integral to generate a composite value based on these criteria. The malicious node is then removed from the cloud market using this composite value. The simulation results demonstrated the potential of the proposed method to prevent nodes in the cloud market from running malware or software that can be used to degrade quality of service by exhausting resources in the cloud market.

Keywords—intrusion detection, cloud market, multiple criterion decision making, security mechanism, fuzzy Integral, malicious nodes

1 Introduction

Cloud-based IT services have seen a substantial increase in the twenty-first century [1, 2]. Due to the sensitivity of clients' information in cloud markets, these markets always have high-security requirements. While external attacks typically are limited in cloud market, internal attacks are often much more difficult to prevent and detect. For internal attacks, the attackers have legitimate access to clients' information on a regular basis, and may know how that information is used. Attackers may sell secret information to other clients in cloud market.

In cloud market, many services are provided to clients by remote servers through the wireless network [24]. Clients get rich computing resources using cloud technology [24]. New security techniques, such as firewalls, virus detection systems, and

cryptographic protocols for secure communication in cloud market, are continually being developed by researchers and businesses to discourage attackers. However, the proposed security techniques are still inadequate to secure the cloud environment as the number of attacks is growing significantly [3]. Furthermore, these strategies were solely focused on the cloud market environment and the market's clients, and they did not consider assessing new clients for security issues prior joining the market. Therefore, obtaining security-related information on the new client node will assist the CSP in determining whether or not to allow the client to join the market. Any customer that uses a hacking tool or runs software that could be used to launch attacks in the cloud market should be blocked by CSP. Furthermore, CSP should use individual node reputation to decide whether or not a node requesting to join the market poses a threat to the cloud market or not.

The availability of cloud market should be guaranteed to gain the trust of clients. CSP should strike a balance between delivering highly available and reliable services and the security measures they employ in order to gain client confidence, satisfaction, and avoid revenue losses.

Researchers and organizations are constantly developing new security mechanisms for deterring attackers in cloud market, such as firewalls, virus detection systems and cryptographic protocols for secure communication. When developing a system with security mechanisms, researchers usually apply static analysis techniques in order to verify that the system behavior adheres to specified security properties. Static analysis techniques range from static verification (e.g. model checking) to techniques measuring the efficiency of algorithms (e.g. encryption). However, cloud environment is distributed system where many clients request services in cloud market dynamically. Unfortunately, when adding new clients to the cloud, many security risks arise. To address this, a new scheme has been proposed in this article with the goal to mitigate the effect of attacks on the performance of cloud market. A fuzzy based security scheme has been proposed for selecting secure clients in the cloud market, in which various important criteria for malicious nodes detection in cloud market are being considered, and every node in the cloud market is assigned a rank using our scheme. The nodes are then authorized to join the cloud market based on their rating.

This paper is organized as follows. We show related works in in Section 2. Next, we describe the system model in Section 3. We formulate the problem in Section 4 and then we describe our scheme. We evaluate the performance of the proposed scheme in Section 5. Finally, the paper is concluded and future research directions are given.

2 Related work

Because the services are delivered to clients via the internet, the cloud market is subject to attacks. In cloud market, CSP serves clients everywhere around the world and charges accordingly [4]. Thus, the cloud becomes increasingly important as consumers come to rely on it, and businesses can now simply hire Cloud services. Gaps in Cloud market are defined in [5] as a lack of confidence between clients and CSP, when clients fear security policies that are hidden from them. CSPs , on the other hand, are concerned that clients would exploit their services and launch attacks in cloud market.

In [6], authors proposed new secure and privacy-preserving distributed deep learning (SPDDL) for fog-cloud computing. SPDDL delivers a superior security, efficiency, and functionality tradeoff. Furthermore, SPDDL can ensure the unforgeability of users' identities in the face of external threats.

Authors introduced in [7] new structure for detecting and preventing denial of service (DoS) attacks and other malicious activity at the network layer by integrating a network intrusion detection system (NIDS) into the Cloud architecture. Intrusion detection system (IDS) is accomplished by monitoring network traffic while maintaining service quality and performance. In IDS, machine learning algorithms use raw data for intrusion detection. However, accessing clients' data may create potential security and privacy risks. To solve this problem, authors proposed in [8] new deep neural network algorithms for data encryption. The proposed solution enables IDS to access clients' data without revealing users' sensitive data. Authors in [9] introduced the SOTA model (Service-Oriented Traceback Architectural) to mitigate two cloud technology threats: HTTP Denial of Service and XML Denial of Service. In this model, a back propagation neural network is employed to track down and detect the source of these attacks.

New IDS was proposed in [10] for identifying attacks in a virtualized cloud under changing environment. The proposed IDS monitors and quantifies the effect of resource adjustments using data collected from the cloud environment. By analyzing objective and subjective trust sources, Bayesian inference was employed in [11] to enable CSP to create credible trust relationships with guest Virtual Machines (VMs). Furthermore, a trust-based maximin game was designed between DoS attackers to minimize the cloud system's detection and hypervisor trying to maximize this minimization under limited budget of resources.

In [12], authors studied an attack scenario where malicious tenants use cloud resources to launch DoS attack targeting data center in cloud market. New approach was proposed for intrusion detection. The approach takes into account the status of virtual machine including CPU usage and network usage. Furthermore, information entropy is applied to monitor the status of virtual machines for detecting attacks in cloud environment. In [13]. authors proposed new Collaborative Network Intrusion Detection System (C-NIDS). C-NIDS monitors network traffic for intrusion detection. C-NIDS uses Support Vector Machine (SVM) to detect network anomaly. Authors proposed new trust model based on virtual machines in [14]. Fuzzy theory was used to calculate the trust value of cloud service providers. In order to protect sensitive information of IoT devices, a new detection system was proposed in [25]. Deep learning was adopted in this system for securing sensitive data of IoT devices. Authors in [26] proposed a new technique for ensuring that a new node that requests to join the cloud does not constitute a threat to the cloud environment. The proposed scheme method checks if a node is running malware or software that could be used to launch an attack before allowing it to join the cloud. Authors discussed in [27] the risks to the cloud environment, as well as proposed detection solutions for malware in the cloud. Furthermore, they suggested a new multi-detection method for preventing malware from spreading in cloud environments.

Fuzzy evaluation engine is proposed in [15] to compute trust value for each resource in cloud market. The proposed evaluation model considered four service measurement indexes: availability, success rate, turnaround efficiency and feedback about a resource.

A new technique based on a genetic algorithm (GA) was proposed in [16] to deal with data integrity and privacy concerns in cloud market. Keys for encryption and decryption are generated using GA. These keys are integrated with a cryptographic algorithm to ensure privacy and integrity of data. Authors proposed new intelligent system with genetic algorithm in [17] to cope with cloud security. A list of users (trusted or un-trusted depend on behavior) is created using the services that provided by cloud. GA was used in [18] for protecting user's data in cloud market. Homomorphic encryption algorithm was used to support the operations in the encrypted domain. GA was customized in [19] for improving data encryption. The generated key is based on altering the population size, number of generations, and mutation rate.

Existing intrusion detection schemes in the cloud market are commonly limited in their ability to identify a wide range of threats. For instance, most of the aforementioned methods can only identify small set of attacks. In this work, we develop general intrusion detection framework to detect a variety of attacks in the cloud market.

3 System model

We model cloud market as network that has M clients divided into K clusters based on their geographical locations as in [20]. The details of the clustering process, as well as its constraints, are outside the scope of this article. In addition to manage the market, the CSP controls network traffic. Clients use networked client devices including desktop computers, laptops, and cellphones to access CSP's resources. To communicate with the CSP, requests are sent via web browser.

To ensure the security of market transactions, we deploy both public-key infrastructure and symmetric-key encryption. To allow clients to participate in the cloud market, CSP employs public-key cryptography. For all trusted clients to encrypt and decode messages, CSP broadcasts the symmetric key.

Let $M = \{m_1, m_2,\ldots, m_n\}$ be the set of clients on which the security test of the clients is performed. Let $P = \{P(m_1), P(m_2),\ldots, P(m_n)\}$ is the set of profiles for clients, where $P(m_i)$, corresponds to the profile of i^{th} client. A client's profile is a collection of attributes that include node ID, event time and location, and time since the last event in the market. CSP uses the proposed security scheme in the cloud market to detect malicious nodes and remove them from the market by informing trustworthy nodes to discard any messages from malicious nodes.

4 Intrusion detection using fuzzy logic

Independent assumption is not realistic in intrusion detection problem in cloud market due to some inherent among the attributes of an intrusion. For this problem, it is possible to utilize fuzzy logic in non-linear circumstances without assuming that one attribute of intrusion is independent of another. The hierarchical fuzzy integral [21, 22] is proposed to detect intrusion in this work.

4.1 Fuzzy measure

Fuzzy measure G_y is defined on $P(X)$ of a finite set X satisfying the following properties [21–23]:

- G_y: $P(X) \rightarrow [0, 1]$
- $G_y(\emptyset) = 0$, $G_y(X) = 1$.
- If $A, B \in P(X)$, $A \subset B$, *then* $G_y(A) \leq G_y(B)$.
- If $H_n \in P(X), \forall 1 \leq n < \infty$ and a sequence $\{H_n\}$ is monotone where $\lim_{n\to\infty} G_y(H_n) = G_y(\lim_{n\to\infty} H_n)$.

In [23], λ-Fuzzy is presented with the following properties:

- $\forall A, B \in P(X), A \cap B = \emptyset$.
- $G_\lambda(A \cup B) = G_\lambda(A) + G_\lambda(B) + \lambda G_\lambda(A) G_\lambda(B), \lambda \in (-1, \infty)$

Definition 1: Assume G_y be a fuzzy measure on X and F be a measurable function where $G_y(X) \rightarrow [0, 1]$. The Sugeno fuzzy integral [23] can be written as follows:

$$z(F) = \int_F F dG_y = max_{i=1}^{n} min(F(x_i), G_y(F_i)) \tag{1}$$

$$F_1 = \{x_1\}, F_2 = \{x_1, x_2\}, \ldots, F_n = \{x_1, x_2, \ldots, x_n\} \tag{2}$$

In this problem formulation, F denotes the performance of a given attribute for the alternatives, while G_y denotes the attribute's weighting grade. The total evaluation for each alternative is given by a fuzzy integral of F with regard to G_y [21–23].

4.2 Intrusion detection using fuzzy integral

In order to achieve high accuracy for detecting intrusion, we take the authentication level for each node, trust level, traffic size, and node activity level as the main criteria.

Authentication test. The main concern of the proposed scheme is to ensure that a new client that requests to join the market does not represent a risk to the cloud environment. The scheme works at two levels: the CSP's level and the client's level. To be accepted into the cloud market, the new node must pass through the suggested two layers of authentication. The certificate for each client is validated through the CSP. The CSP issues certificate for each client in the market. Each certificate includes: logic identifier, MAC address, and a pair of its public/private keys. Since each node's certificate is issued by the CSP, each client in the market contacts the CSP to validate other certificates for others nodes. Each client may get all node information from the CSP for authentication purposes, with the exception of the node's private key, which is not shared with any other node in the market. To avoid launching attacks in the market, CSP should scan new nodes for hacking tools and viruses before allowing it to join the cloud market.

The new node has to send its CSP's certificate to CSP, which only accepts nodes from a predefined CSPs' list in the market. If the certificate was issued by a known CSP, the CSP issues the node a new certificate. The client then sends the new certificate along

with a message containing the node ID and MAC address, which is encrypted with the public key. The following is the definition of authentication's evaluation value:

$$A_t = \begin{cases} 1, & authenticat\ node \\ 0, & o.w \end{cases} \tag{3}$$

Trust level based on multiple criteria decision making. When it comes to calculate the trust level for each node, there are a lot of parameters to consider. These parameters include: response time, throughput, availability, and success rate. The CSP keeps track of the values of these attributes in the performance logs. These values are updated after each event in the market, ensuring that the most recent parameter values are always current. For each node, the overall length of time it takes to react to a service request is known as response time. The response time for i^{th} node is computed as follows:

$$T_i = \frac{T_r - T_{min}}{T_{max} - T_{min}} \tag{4}$$

where the T_{max} and T_{min} are the maximum and minimum response time in the neighborhood, respectively; and T_r is the response time of a regular node. The higher the T_i value, the more likely the node is to be an intruder.

Throughput for the node is the percentage of messages successfully transmitted via a communication medium. Throughput for i^{th} node is calculated as follows:

$$H_i = \frac{H_r - H_{min}}{H_{max} - H_{min}} \tag{5}$$

where the H_{max} and H_{min} are the maximum and minimum throughput in the neighborhood, respectively; and H_r is the throughput of a regular node. The lowest the H_i value, the more likely the node is to be an intruder.

The length of time a system works at full functionality during the time it is required to do so is referred to as availability. In this situation, the CSP would strive to resolve the issue so that the node could continue to function. Frequent node failure, on the other hand, indicates that this node is acting maliciously. Therefore, the evaluation value of availability can be computed as follows:

$$V_i = 1 - \frac{V_0 - V_{min}}{V_{max} - V_{min}} \tag{6}$$

where V_0 denotes the total failure times of i^{th} node; V_{max} and V_{min} are the maximum and minimum restart number received from neighbors, respectively. The percentage of requests that are fulfilled successfully is known as the success rate. Therefore, the evaluation value of success rate can be evaluated as follows:

$$S_i = 1 - \frac{S_0 - S_{min}}{S_{max} - S_{min}} \tag{7}$$

where S_0 denotes the total number of requests successfully completed by i^{th} node; S_{max} and S_{min} are the maximum and minimum number of requests that executed successfully from neighbors, respectively.

Traffic size. When an intruder node in cloud market pump more traffic into the network than it can handle, subsequent nodes face high contention rates, rendering cloud resources unavailable to clients. However, when one of the later nodes fails to relay a packet after several tries, the link is declared broken, and the routing scheme starts looking for a new path. In the cloud market, no packets can be forwarded until a new route is discovered. As a result, the number of packets lost increases, and the throughput declines dramatically.

The more pumping of unnecessary traffic in the network, the more waiting time in the network and the more degrading of quality of service. In our work, $D^K(d_i)$ denotes the distance between i^{th} node's and its K^{th} nearest data rate of neighbor. All rates are ranked based on their $D^K(d_i)$ distances, which leads to the following definition of outliers:

$$O_i = \begin{cases} 1, & i \in O \\ 0, & o.w \end{cases} \tag{8}$$

where O denotes the set of greatest distances.

Node activity level. We distinguish three types of clients in the cloud market based on their behavior: normal, passive, and hyperactive. Some nodes exchange data with their neighbors more frequently than others, and they interact with others more frequently. The client is interested in increase of his/her benefits. Therefore, more efforts do, the more benefit for a client. A client makes his/her decision for increasing the benefit. The decision depends on the conditions of the market conditions.

The activity of a node is determined by the number of events in which it participates. Relaying packets and generating new messages are examples of these events. Node activity can be computed as follows:

$$L_i = \Sigma_{T \to \infty} a_t \tag{9}$$

where a_t denotes the event at time t, and T is the time horizon. The evaluation value of node activity level can be evaluated as follows:

$$N_i = \frac{L_i - L_{min}}{L_{max} - L_{min}} \tag{10}$$

where L_i denotes the node activity level of i^{th} node; L_{max} and L_{min} are the maximum and minimum number of activities in which neighbors participate, respectively.

4.3 Intrusion detection scheme based on multiple criteria decision making

The trust value of a node is determined using a variety of criteria, including authentication level for each node, trust level, traffic amount, and node activity level. CSP measures and maintains the values of these attributes. The evaluation matrix E_i for i^{th} node is given below:

$$E_{i,j} = \begin{bmatrix} e11 & e12 & e13\ldots\ldots\ldots \\ e21 & e22 & e23\ldots\ldots\ldots \\ em1 & em2 & em3\ldots\ldots\ldots \end{bmatrix} \quad (11)$$

where *eij* represents the value of i^{th} event in market for j^{th} attribute. Analytical Hierarchy Process (AHP) is used to assign the wight for attributes. The pair-wise matrix for each attribute is constructed to compute the wight. This matrix is used to find the comparative priority of each attribute over the other. The following algorithm is used to calculate the weights and node trust:

Algorithm 1: Intrusion Detection

Inputs:

E_i: *The evaluation matrix for i^{th} node.*
N: *number of attributes for event.*
X: *number of events.*

Outputs:

P: *matrix of pairwise comparisons for event attributes.*
w: *matrix of wights for attributes.*
T_n: *Trust value for n^{th} node.*

1. $\boldsymbol{P}$ *= Genrate- pair-wise()*
2. for $\boldsymbol{i} \leftarrow \boldsymbol{0}, \boldsymbol{N}$ *do*
3. *t=1*
4. *for* $\boldsymbol{j} \leftarrow \boldsymbol{0}, \boldsymbol{N}$ *do*
5. *t=t*P[i][j]*
6. *end for*
7. *H[i] = t;*
8. end for
9. Sum = 0
10. for $\boldsymbol{i} \leftarrow \boldsymbol{0}, \boldsymbol{N}$ *do*
11. $RootN[i] = \sqrt[N]{\boldsymbol{H}[\boldsymbol{i}]}$
12. *Sum=Sum+ RootN[i]*
13. end for
14. for $\boldsymbol{i} \leftarrow \boldsymbol{0}, \boldsymbol{N}$ *do// Wight calculation*
15. $W[i] = \frac{\boldsymbol{RootN}[\boldsymbol{i}]}{\boldsymbol{sum}}$
16. end for
17. for $\boldsymbol{i} \leftarrow \boldsymbol{0}, \boldsymbol{X}$ *do// Computing node trust*
18. $\boldsymbol{T_n} = \sum_{\boldsymbol{j}=\boldsymbol{1}}^{\boldsymbol{n}} \boldsymbol{W}[\boldsymbol{i}]\boldsymbol{E}[\boldsymbol{i}][\boldsymbol{j}]$
19. end for
20. if ($\boldsymbol{T_n} > \boldsymbol{0.5}$*) then Accept=Node() else Reject-Node();*
21. End

The market's trust value for each client will be in the range of [0–1]. If the trust value is less than 0.5, the node is removed from the market, and CSP sends a warning message to all nodes telling them to avoid messages from this node. If the value of trust is more than 0.5, the CSP will enable the node to participate in the cloud market.

5 Performance evaluation

We test the suggested security scheme to identify the attacker nodes that degrade network performance in cloud market. Table 1 illustrates the network that was simulated, along with the values that were utilized for the needed parameters. By monitoring node activity and analyzing node data, the results are evaluated to highlight the importance and implications of using our scheme to protect data across cloud market. In the simulations, the following major performance measures are of interest:

(1) Throughput, which is the average rate at which a message is delivered successfully via a communication connection.
(2) Resources utilization, which is the average amount of time the resources in cloud market are used.
(3) Delay

Table 1. Simulation Parameters

Parameter	Value
Number of clients	200
Number of messages per client	Random
Type of interface per node	802.11 b
MAC layer	IEEE 802.11 b
Path loss exponent	4
d_0	5 m
Transmission power	0.1 watt
Packet size	512

Figure 1 shows a comparison of throughput for network in the cloud with the help of our security scheme (secure cloud, SC) and cloud without security mechanism (NSC). It is clear from the figure that the throughput shifts into higher level when SC is applied and the arrival rate for requests increases. Some malicious nodes keep dropping packets and decreases significantly the number of received packets successfully. Furthermore, these malicious nodes block packets from being forwarded. In terms of throughput, our scheme surpasses NSC since it filters out these nodes.

For varying values of request arrival rates in the cloud market, Figure 2 depicts the packet drop ratio analysis between the NSC and suggested SC. Even with high levels of arrival rates, the packet drop ratio for SC gradually reduces as compared to NSC. Since our scheme excludes all malicious nodes, the packet dropping ratio has decreased significantly. Unfortunately, some cloud market nodes refuse to relay packets and discard part of them.

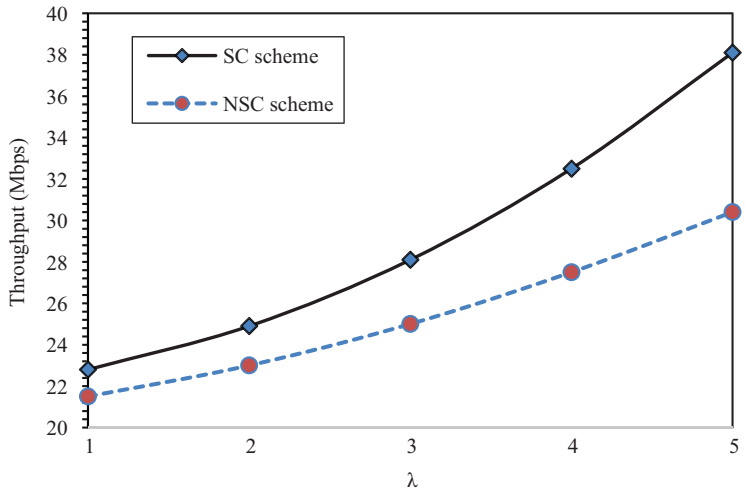

Fig. 1. Throughput under different values of node's arrival rates

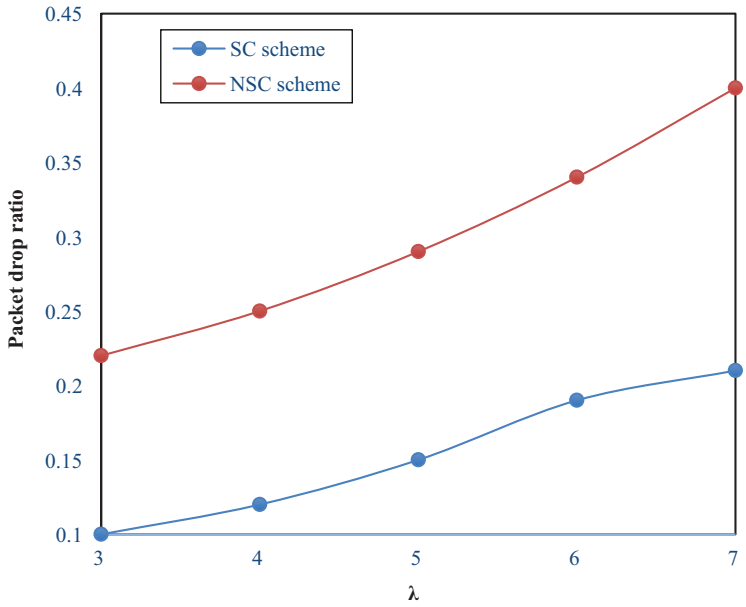

Fig. 2. Paket drop ratio under different values of arrival rates

In Figure 3, we examine resource utilization for both schemes at various task load levels (i.e. arrival rate). The figure clearly shows that when the load increases, the utilization of resources falls. As the number of attacker nodes increases, resource utilization falls dramatically. Some attackers use cloud resources far more frequently than the ordinary client. Furthermore, attackers may continue to transmit malicious traffic until the cloud's resources, such as network resources, processors, and servers, are depleted.

Because our approach keeps these attackers out of the cloud market, it makes better utilization of resources than the NSC scheme.

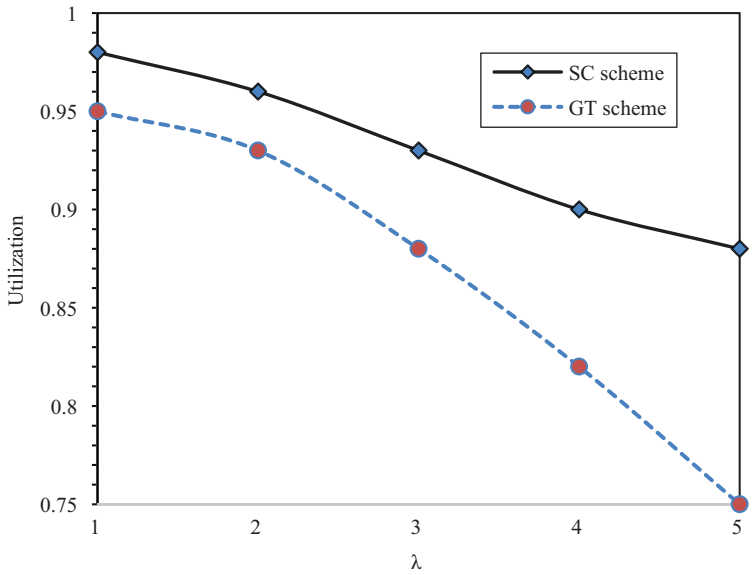

Fig. 3. Utlization under different values of arrival rate

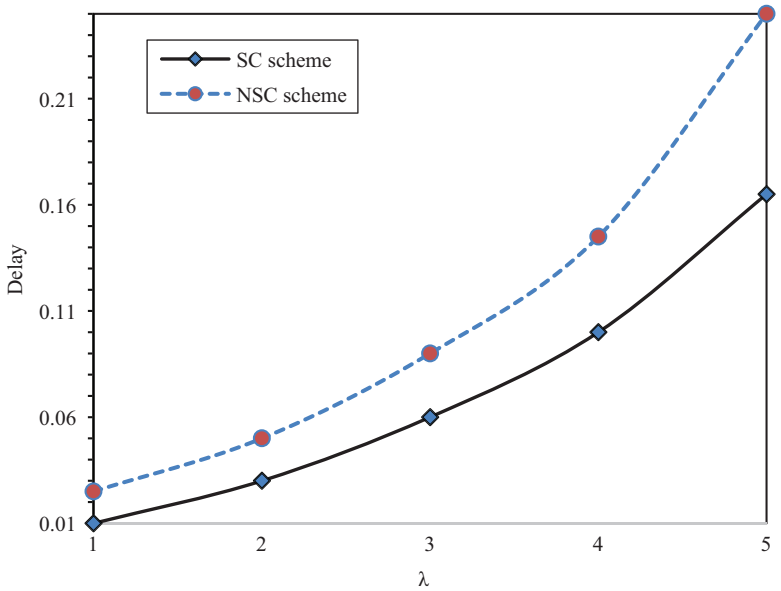

Fig. 4. Delay for different values of arrival rate

We measure the delay in a different level of networks' load to see how attackers affect the quality of service for clients. The latency increases as the value of λ (i.e. network load) increases, as shown in Figure 4. By flooding the cloud market with malicious traffic, some attackers attempt to render all resources in the cloud market inaccessible to clients. As seen in Figure 4, our scheme eliminates these nodes, resulting in a significant reduction in service delay in the cloud market.

6 Conclusion

The security challenge is an important research topic that will have an impact on the operational efficiency of cloud computing industry. This research looked into security concerns when designing an intrusion detection system for the cloud market. Unfortunately, cloud market is vulnerable to a variety of threats. We are concentrating on identifying malicious nodes and eliminating them from the cloud market by developing a new fuzzy-based security scheme. The key contribution of this work is that the proposed approach treated the security challenge in the cloud market as a multi-source information fusion problem, with the criteria depicted as evidence by taking into account both the subjective and objective weights of these criteria.

The fuzzy integral was used to combine the most important criteria that can influence intrusion detection in the cloud market into a single one. Furthermore, the new strategy offers data fusion at CSP, which can effectively eliminate redundant data and minimize traffic in the cloud market. The simulation results showed that the suggested security scheme greatly increase throughput while also improving service quality. We intend to deploy the proposed scheme in the real world and analyze it against a variety of attack types in the future.

7 References

[1] Ali, M., Khan, S. U., & Vasilakos, A. V. (2015). Security in cloud computing: Opportunities and challenges. Information Sciences, 305(2015), 357–383. https://doi.org/10.1016/j.ins.2015.01.025

[2] Khan, S., Parkinson, S., & Qin, Y. (2017). Fog computing security: A review of current applications and security solutions. Journal of Cloud Computing, 6(1), 1–22. https://doi.org/10.1186/s13677-017-0090-3

[3] Kouatli, I. (2016). Managing cloud computing environment: Gaining customer trust with security and ethical management. Procedia Computer Science, 91(2016), 412–421. https://doi.org/10.1016/j.procs.2016.07.110

[4] A. Alsarhan, A. Itradat, A. Y. Al-Dubai, A. Y. Zomaya, and G. Min, "Adaptive resource allocation and provisioning in multi-service cloud environments," in IEEE Transactions on Parallel and Distributed Systems, vol. 29, no. 1, pp. 31–42, 1 Jan. 2018, doi: https://doi.org/10.1109/TPDS.2017.2748578

[5] M. T. Khorshed, A. B. M. S. Ali, and S. A. Wasimi, "Trust issues that create threats for cyber attacks in cloud computing," in Proc. IEEE 17th Int. Conf. Parallel Distrib, pp. 900–905, Dec. 2011, doi: https://doi.org/10.1109/ICPADS.2011.156

[6] Y. Li, H. Li, G. Xu, T. Xiang, X. Huang, and R. Lu, "Toward secure and privacy-preserving distributed deep learning in fog-cloud computing," Internet of Things Journal IEEE, vol. 7, no. 12, pp. 11460–11472, 2020. https://doi.org/10.1109/JIOT.2020.3012480

[7] C. Modi, D. Patel, B. Borisanya, A. Patel, and M. Rajarajan, "A novel framework for intrusion detection in cloud," in Proc. 5th Int. Conf. Secur. Inf. Netw. (SIN), pp. 67–74, 2012, doi: https://doi.org/10.1145/2388576.2388585

[8] E. Hesamifard, H. Takabi, M. Ghasemi, and C. Jones, "Privacy-preserving machine learning in cloud," in Proc. Cloud Comput. Secur. Workshop, pp. 39–43, 2017, doi: https://doi.org/10.1145/3140649.3140655

[9] Chonka, A., Xiang, Y., Zhou, W., & Bonti, A. (2011). Cloud security defence to protect cloud computing against HTTP-DoS and XML-DoS attacksa. Journal of Network and Computer Applications, 34(4), 1097–1107. https://doi.org/10.1016/j.jnca.2010.06.004

[10] Abusitta, A., Bellaiche, M., & Dagenais, M. (2018). An SVM-based framework for detecting DoS attacks in virtualized clouds under changing environment. Journal of Cloud Computing 7, 9. https://doi.org/10.1186/s13677-018-0109-4

[11] O. A. Wahab, J. Bentahar, H. Otrok, and A. Mourad, "Optimal load distribution for the detection of VM-based DDoS attacks in the cloud," in IEEE Transactions on Services Computing, vol. 13, no. 1, pp. 114–129, 1 Jan.–Feb. 2020, doi: https://doi.org/10.1109/TSC.2017.2694426

[12] Cao, J., Yu, B., Dong, F., Zhu, X., & Xu, S. (2015). Entropy-based denial-of-service attack detection in cloud data center. Concurrency and Computation: Practice and Experience, 27(18), 5623–5639. https://doi.org/10.1002/cpe.3590

[13] Al Haddad, Z., Hanoune, M., & Mamouni, A. (2016). A collaborative network intrusion detection system (C-NIDS) in cloud computing. International Journal of Communication Networks and Information Security, 8(3), 130. https://doi.org/10.1109/CloudTech.2016.7847708

[14] Gu, L., Wang, C., Zhang, Y., Zhong, J., & Ni, Z. (2014). Trust model in cloud computing environment based on fuzzy theory. International Journal of Computers Communications & Control, 9(5), 570–583. https://doi.org/10.15837/ijccc.2014.5.1276

[15] Priya, G., & Jaisankar, N. (2019). A fuzzy based trust evaluation model for service selection in cloud environment. International Journal of Grid and High Performance Computing (IJGHPC), 11(4), 13–27. https://doi.org/10.4018/IJGHPC.2019100102

[16] Tahir, M., Sardaraz, M., Mehmood, Z. et al. (2021). CryptoGA: a cryptosystem based on genetic algorithm for cloud data security. Cluster Computing 24, 739–752. https://doi.org/10.1007/s10586-020-03157-4

[17] Al-Shaikhly, M. H., El-Bakry, H. M., & Saleh, A. A. (2018). Cloud security using Markov chain and genetic algorithm. International Journal of Electronics and Information Engineering, 8(2), 96–106.

[18] Jiang, L., & Fu, Z. (2020). Privacy-preserving genetic algorithm outsourcing in cloud computing. Journal of Cybersecurity, 2(1), 49. https://doi.org/10.32604/jcs.2020.09308

[19] Arshad, M. J., Umair, M., Munawar, S., Naveed, N., & Naeem, H. (2020). Improving cloud data encryption using customized genetic algorithm. International Journal of Intelligent Systems & Applications, 12(6). https://doi.org/10.5815/ijisa.2020.06.04

[20] A. Alsarhan and A. Agarwal, "Cluster-based spectrum management using cognitive radios in wireless mesh network," 2009 Proceedings of 18th International Conference on Computer Communications and Networks, pp. 1–6, 2009, doi: https://doi.org/10.1109/ICCCN.2009.5235261

[21] Tzeng, G. H., OuYang, Y. P., Lin, C. T., & Chen, C. B. (2005). Hierarchical MADM with fuzzy integral for evaluating enterprise intranet web sites. Information Sciences, 169(3–4), 409–426. https://doi.org/10.1016/j.ins.2004.07.001

[22] Sugeno, M. (1974). Theory of fuzzy integrals and its applications. Ph.D. Dissertation, Tokyo Institute of Technology, Tokyo, Japan.

[23] A. Alsarhan, Y. Kilani, A. Al-Dubai, A. Y. Zomaya, and A. Hussain, "Novel fuzzy and game theory based clustering and decision making for VANETs," in IEEE Transactions on Vehicular Technology, vol. 69, no. 2, pp. 1568–1581, Feb. 2020, doi: https://doi.org/10.1109/TVT.2019.2956228

[24] A. Alsarhan, A. Itradat, A. Y. Al-Dubai, A. Y. Zomaya, and G. Min, "Adaptive resource allocation and provisioning in multi-service cloud environments," in IEEE Transactions on Parallel and Distributed Systems, vol. 29, no. 1, pp. 31–42, 1 Jan. 2018, doi: https://doi.org/10.1109/TPDS.2017.2748578

[25] M. Ali Alheeti, K., Alsukayti, I., & Alreshoodi, M. (2021). Intelligent botnet detection approach in modern applications. International Journal of Interactive Mobile Technologies (iJIM), 15(16), 113–126. https://doi.org/10.3991/ijim.v15i16.24199

[26] Aljammal, A. H., Bani-Salameh, H., Alsarhan, A., Kharabsheh, M., & Obiedat, M. (2017). Node verification to join the cloud environment using third party verification server. International Journal of Interactive Mobile Technologies (iJIM), 11(4), 55–65. https://doi.org/10.3991/ijim.v11i4.6501

[27] El-Khouly, M. M., & Abou El-Seoud, S. (2017). Malware detection in cloud environment (MDCE). International Journal of Interactive Mobile Technologies (iJIM), 11(2), 139–145. https://doi.org/10.3991/ijim.v11i2.6575

8 Authors

Ayoub Alsarhan received the B.E. degree in computer science from the Yarmouk University, Jordan, in 1997, the M.Sc. degree in computer science from Al-Bayt University, Jordan, in 2001, and the Ph.D. degree in electrical and computer engineering from Concordia University, Canada, in 2011. He is currently Professor with the Department of Information Technology, Hashemite University, Zarqa, Jordan. His research interests include cognitive networks, parallel processing, cloud computing, machine learning, and real-time multimedia communication over the Internet.

Abdel-Rahman Al-Ghuwairi received his Ph.D. in Computer Science from the New Mexico State University, USA in 2013, M.Sc. in Computer Science from the University of Jordan in 2006, and B.Sc. in Computer Science from the M'utah University in 1990. He is currently an Associate Professor at the Software Engineering Department of the Hashemite University, Zarqa, Jordan. His research interests include software engineering, cloud computing, service level agreements, requirement engineering, cloud computing database, information retrieval, big data, and database systems.

Esra'a Alshdaifat received his Ph.D. in Computer Science from the University of Liverpool, United Kingdom in 2015, M.Sc. in Computer Science from the Yarmouk University in 2008, and B.Sc. in Computer Science from the Hashemite University in 2006. He is currently an Associate Professor at the Department of Information Technology of the Hashemite University, Zarqa, Jordan. Her research interests include Machine Learning and Information Retrieval, Knowledge Discovery in Data (KDD), Natural Language Processing and Data mining.

Hasan Idhaim received his M.Sc. in Computer Science from the New Mexico Highlands University in 2008, and B.Sc. in Computer Science from the Yarmouk University in 1985. He is currently Lecturer at the Computer Information System Department of the Hashemite University, Zarqa, Jordan. His research interests include Database Design & Tuning, E Commerce Application, and Industrial Data analysis.

Omar alkhawaldeh received his M.Sc. in Computer Information System Hashemite University, and B.Sc. in Software Engineering from the Hashemite University. His research interests include cognitive networks, parallel processing, cloud computing, machine learning, and real-time multimedia communication over the Internet.

Article submitted 2021-10-29. Resubmitted 2021-12-19. Final acceptance 2021-12-22. Final version published as submitted by the authors.