Machines and Algorithms

http://www.knovell.org/mna

Review Article

Enhancing IoT Security through Fog Computing and SDN: Trust-Based Approach

Tehseen Irshad[1*], Tehreem Akhtar[2] and Muhammad Sharif Imam [3]

[1]Department of Computer Science, Bahauddin Zakariya University, Multan, Pakistan

[2]Department of Computer Science, GC University Faisalabad, Multan Sub-Campus, Pakistan

[3]Department of computer science, COMSATS University Islamabad, Sahiwal campus, Pakistan

[*]Corresponding Author. Email: tehseenirshad7370@gmail.com

Received: 06 June 2022; Revised: 18 June 2022; Accepted: 05 August 2022; Published: 17 August 2022

AID: 001-02-000010

Abstract: The Internet of Things (IoT) has revolutionized data collection and processing through seamless communication among smart devices. While cloud computing occasionally struggles to deliver rapid, high-quality services, fog computing emerges as a dynamic alternative, offering swift computation and service provisioning. This research introduces an innovative IoT architecture merging software-defined networking and fog computing. The core features pioneering algorithms managing access control and evaluating trust. These algorithms seamlessly integrate new fog nodes, assigning non-sensitive tasks. Communication channels among fog nodes, coupled with behavior reporting to the Fog Manager node (FMN), enhance transparency and adaptability. The FMN evaluates fog node reliability preemptively, bolstering security by sieving out untrustworthy nodes. Validated via Java implementation in iFogSim, the framework swiftly identifies and mitigates malicious fog node activities, ensuring fog environment security and efficiency. By fusing software-defined networking and fog computing, this research addresses service speed, scalability, and security challenges, contributing to a more secure, adaptable, and efficient IoT future.

Keywords: IoT, Fog Computing, SDN, Cloud, Access control, Weighted Trust Management, Security, Dynamic behaviors.

1 Introduction

A variety of intelligent devices are included into the Internet of Things' (IoT) architecture, which is a linked and networked environment. The number of Internet-connected Internet of Things (IoT) devices surpassed the world's population in 2010, claims Cisco. According to predictions, there will be more than 75 billion active Internet of Things devices by 2025 [1, 2]. Because processing nodes can be added and removed as needed during program execution, cloud computing increases application flexibility. Although cloud computing has proven its efficacy in various contexts, it falls short when it comes to ensuring reliable, low-latency inputs within environments such as intelligent transportation systems, industrial vehicle systems, and healthcare systems [3]. In recent years, the Internet of Things has attracted a lot of interest and has developed into an essential part of our daily life [4]. The expansion is driven by global connectivity of

everyday devices like refrigerators and fans, as well as applications in smart cities. Additionally, advancements in wireless communications and electronic devices contribute to the proliferation of connected devices [5]. Rapid innovation and the benefits of scalable app design for serving a large user base simultaneously are fueling the development of cloud computing. This reduces the need for service providers looking to reduce infrastructure requirements to create huge personal data warehouses [3].

A particularly challenging task within the realm of IoT is edge computing, which involves real-time data processing on devices. Tasks that require greater processing power or storage space are frequently offloaded to the cloud, which could cause service delays. However, challenges arise in managing shared data between end-user devices and the cloud due to a lack of established frameworks [2]. CISCO introduced the fog computing model as an extension of cellular edge computing, providing a viable solution for IoT networks and applications. Fog computing, which involves temporarily storing data in local fog nodes, enhances security compared to traditional cloud computing. By deploying numerous cloud computing resources at the network's edge, fog computing reduces latency, improves Quality of Service (QoS), and benefits end-users [7]. Fog computing complements cloud computing by introducing a geographically dispersed layer of fog nodes, enhancing existing services [1]. Software-defined networks (SDNs), which separate the control plane from the data plane, are frequently used by researchers in fog computing for the delivery of real-time services [1]. Network administration has been transformed with the development of SDN and network function virtualization, which allow for sophisticated resource optimization techniques via centralized network management [5]. However, managing widely distributed fog computing infrastructure using a central SDN plane can lead to reliability and performance concerns. As a result, several researchers have used a distributed SDN control plane method [8].

IoT networks face difficulties with bandwidth, network latency, access control, authentication, and reliability. Despite being a useful feature, device-to-device connectivity poses security and privacy issues when rogue IoT devices share data. These issues can render IoT systems ineffective and even detrimental [2, 7, 9]. In IoT systems, numerous nodes connect to the nearest gateway, which in turn connects to other global regions. IoT nodes need adaptable connectivity, sometimes requiring disconnection from and reconnection to the gateway based on circumstances [10]. The interaction and service exchange among fog nodes play a crucial role in task completion. Evaluating and removing malicious fog nodes poses difficulties for network integrity [7].

Amid the pervasive growth of IoT and challenges in cloud computing's latency-sensitive applications, this research pursues two primary objectives:

- Enhanced IoT Architecture and Integration*:* Explore the fusion of edge, fog, and cloud computing in IoT architecture, analyzing integration factors and drivers for widespread adoption.
- Efficient Fog Computing and Security: Investigate fog computing's local data storage, security advantages, and SDN-based real-time services. Develop innovative trust models and access control frameworks to enhance IoT security while identifying malicious fog nodes.

Existing research addresses direct and indirect trust in fog nodes. Direct trust involves evaluating fog node trust based on personal experience, while indirect trust relies on historical behavior. None of these models, however, address the reliability of the trust evaluator. In our suggested paradigm, we define a unique framework for access control and dynamic weighted trust management and introduce the idea of trust evaluator integrity. Our method uses a Fog Manager Node (FMN) to control access to newly connected fog nodes and a centralized architecture. The FMN assigns non-essential tasks to new devices and facilitates access and task delegation. Additionally, we present a technique for estimating the dependability of recently connected devices. The SDN controller oversees network infrastructure, notifying FMNs of malicious fog nodes to prevent their network entry.

2 Literature Review

The authors present a method to manage defective fog nodes in computer systems using a trusted model and role-based access control [3]. Dynamic nodes, performing mathematical functions, are integrated with static and processing nodes. The Fog Nodes Manager (FNM) supervises nodes, assessing issues and

assigning tasks based on trust levels. The system accommodates nodes entering and leaving over time. Jobs are categorized by FNM into connected nodes, with confidence levels determined using equations considering factors like availability and reliability. The Fog Network's role extends to trusted nodes, augmenting capabilities. The proposed Fog computer platform utilizes System C, with Model A System C for testing, allowing interactions among processing components. The system accommodates varying processing frequencies and registered applications distributed randomly over time.

The authors proposed [2] a trust and reliability framework tailored for IoT networks within Fog Computing. They build to address limitations of a prior framework. The approach employs a trust and reputation model where IoT devices evaluate reliability using error codes, connecting only with devices surpassing a predefined confidence threshold. This method effectively guards against attacks such as negative publicity, on-off, and self-promotion. A testbed is employed to simulate IoT device behavior, assessing scenarios including Bad-Mouthing, On-Off attacks, and self-promotion. The results demonstrate successful defense against attacks, maintaining confidence levels despite varying attacker ratios. Notably, attempts by self-promoting malicious devices are thwarted, showcasing effective trust restoration.

The authors introduce an adopted trust and reputation model for mobile agent systems. Users select service providers based on past experiences of investigators and witnesses [9]. The credibility of witnesses is also evaluated to prevent false reports. The framework incorporates customizable weights for evaluations, incentivizing accurate reporting through discounts and fines. Moreover, the approach addresses the behavior of detached agents. A testbed simulation assesses the model with six auditors and 25 service providers. Users make selections from five providers based on trust and service quality thresholds. The outcomes across 50 test runs demonstrate the model's effectiveness, enhancing security in mobile agent systems.

The authors proposed in [10], the integration of distributed trust management into IoT systems is presented to handle the scale and diversity of IoT devices. A multi-layered architecture is designed, incorporating cloud, identity, gateway, IoT, node, and server frameworks, establishing trust among IoT entities. This architecture guarantees dependable end-to-end IoT data flow. Communication flows involve interactions between endpoints, devices, gateways, and servers, promoting secure and reliable IoT data management.

In [3, 7], the authors introduce a Hidden Markov Model (HMM) based approach to identify rogue fog nodes securely and scalable. The trained HMM effectively detects malicious fog nodes with high accuracy, encompassing instruction, observation, and detection phases. These malicious fog nodes pose a threat to user data privacy, making connections to them hazardous. The authors emphasize the importance of embedding security and privacy considerations throughout the fog computing architecture. The HMM approach, tested using MATLAB R2016a and Eclipse IDE, demonstrates efficient detection of rogue nodes, enhancing fog network security against various attack scenarios.

In this research, the authors delve into the challenges within Intelligent Transportation Systems (ITS), focusing on functional and non-functional aspects, authorization, privacy, interoperability, and more [11]. A four-part ITS model is proposed, incorporating cloud infrastructure, roadside, vehicles, and sensors. This model integrates fog computing to enhance latency, application localization, and direct device-to-consumer connections. The study scrutinizes access control management, including attributes-based access control (ABAC), reference monitors (RM), policy distribution, and offline capabilities. Furthermore, prospective approaches for reference monitor deployment are discussed.

This research introduces a secure routing and handoff mechanism for IoT devices and fog nodes employs trust scores to mitigate attacks [12]. The mechanism calculates trust values and updates a lookup table to enhance fog node reliability. In [13], the focus is on the reliability of fog node authentication for data provider and requester verification. A system that combines fog nodes and IoT devices for access verification is proposed, utilizing Ethereum smart contracts. The model features five key components and a protocol with phases like device registration, mapping, authentication, token creation, and data exchange. Python programming language is used for testing, demonstrating improved performance with increased group tail bits and ensuring secure decentralized storage systems.

The author presents a fog computing method, Ciphertext-policy attribute-based encryption (CP-ABE), supporting outsourcing and attribute change [14]. The method ensures independent determination of data owner and decryption user, minimizing costs for attribute updates. Cloud service providers, fog nodes, data owners, end users, and authorities are involved in the proposed system. The process comprises five stages, using Java Cryptographic Library for efficient encryption and decryption. Computational costs are significantly reduced compared to other models, making it suitable for resource-constrained devices.

In this Paper [15], effective functional encryption (FE) schemes tailored for fog computing are explored. These schemes address continual memory leakage (CML) threats, ensuring privacy and precise access control. Due to potential physical attacks in fog environments, traditional functional encryption might not suffice. The paper advocates a shift from LR-FE to conditional coding, introducing designs like double encryption with leak prevention and a sealed FE encoder. The study underscores the need for advanced cryptographic solutions in fog computing security.

Proposing a security architecture for IoT and fog collaboration in [16], the authors integrate access control and monitoring for secure resource collaboration. The architecture consists of isolated fog computing cells, each controlled by a central Fog Nodes Manager (FNM). A mechanism for scheduling and resource allocation is provided to optimize system performance. The FNM serves as an access manager and resource classifier. An algorithm enhances credibility and rating based on device class. The network architecture, iFogSim, is tested using a Java application, highlighting the superiority of the proposed mechanism, TACRM, in terms of efficient resource management and improved response time compared to cloud server services.

Comparing various Fog security designs based on IoT security criteria in [17], the authors analyze robust authentication methods for IoT devices, emphasizing IoT security objectives. They address the need for a standardized fog architecture to manage trust and privacy, mitigating potential IoT security issues. The strategy aims to enhance visibility for IoT devices and fog nodes, addressing challenges in decentralized design. Authentication technologies are evaluated using factors like security, usability, and productivity, and the study combines qualitative and quantitative data to create a comprehensive diagnostic framework for IoT authentication solutions.

Finally, the authors introduce a cloud-fog control middleware framework to efficiently manage service requests and node management [18]. The approach combines cloud and fog computing, reducing energy consumption and service times. The framework addresses data integration, modification, and security concerns stemming from Cloud-IoT-Fog interaction. Inadequate resource policies and lack of user activity monitoring can lead to attacks and security issues. The authors highlight the necessity of defense mechanisms against attacks like resource misuse and viruses to maintain efficient fog system performance. The proposed model offers anti-fog solutions and resource management techniques suitable for large-scale data processing scenarios.

In summary, each reviewed work contributes unique insights to address challenges within Fog Computing and IoT networks. These approaches provide solutions while paving the way for potential future research directions and improvements in network reliability, security, and efficiency.

3 Proposed Methodology

Our novel fog computing framework, guided by software-defined principles, enhances security in the fog computing environment. It fortifies the processing capabilities of end-users and fog nodes for secure utilization. To achieve this, we introduce a Fog Manager Node (FMN) responsible for overseeing fog nodes, access control, and trust management.

3.1 Software-Defined Network-Based Fog Computing

We will examine the distinct layers comprising our innovative fog computing architecture enhanced by SDN (Software-Defined Networking). These layers encompass the IoT devices stratum, the Fog stratum, the SDN controller tier, and the cloud stratum, each housing distinct processes and functionalities.

3.2 IoT Devices Layer

The Internet of Things (IoT) encompasses various networks that amass and exchange data, encompassing devices like computers, gadgets, vehicles, residences, and other items embedded with sensors, circuits, software, electronics, and network connectivity. Within our envisioned framework, IoT devices can seamlessly integrate into the network and establish communication with the fog manager node and neighboring nodes. This enables mutual interaction and service sharing among these devices. The Fog Manager Node (FMN) assesses the reliability of devices based on their conduct. In cases where a device engages in malicious behavior, it is promptly eliminated from the network.

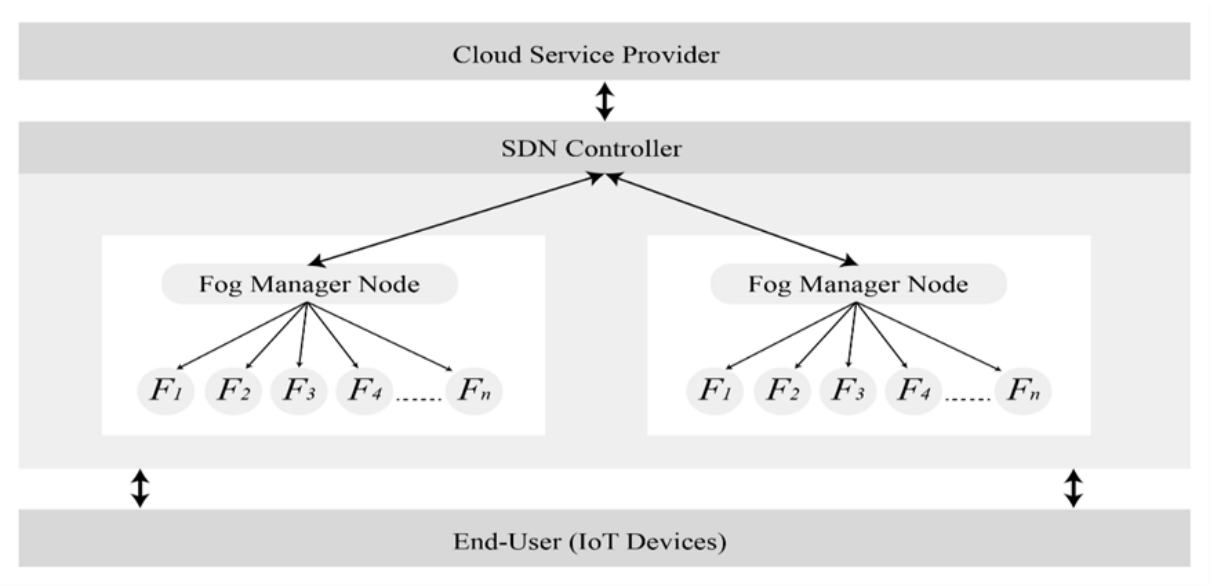

Figure 1: Proposed SDN-Based Fog Computing Framework

3.3 Fog Computing Layer

Edge computing, as described by Cisco, possesses a broader scope compared to fog computing, which primarily involves intelligent doors and sensors. This idea gives commonplace items like motors, pumps, and lights the ability to analyze large amounts of data. Within these objects, known as network edge devices, significant data preparation is intended to be done.

Within the context of our model, fog computing encompasses two distinct types of nodes: parent nodes and child nodes. The parent nodes are the superior nodes, also known as fog management nodes or fog heads, while the child nodes are the inferior nodes, also known as fog nodes or child nodes.

3.4 Fog Nodes

"Cloud nodes" are software applications that run on IoT devices. These nodes interface with other end-user IoT devices using protocols like CoAP and SNMP. The device count mustn't exceed available computing resources. Cloud nodes encompass devices like routers, access points, switches, gateways, firewalls, and dedicated servers. They can integrate SDN equipment, like switches or routers, or link directly to SDN devices. Each cloud node consists of these operational modules:

- Cloud Manager: Initiates requests to the server, which then undertakes the assigned task.
- Monitors: Part of IT service implementation, this module oversees operations.
- Database: Stores incoming requests, updates the node's system status and available resources, and oversees data readiness.

3.5 Fog Manager Node

The FMN plays a central role in our architecture. It verifies the legitimacy of fog nodes attempting to join the network. Once verified, new fog nodes are allocated non-sensitive tasks. FMN continuously monitors new fog nodes, computing trust scores based on interactions. Interactions between fog nodes, particularly fog-to-fog interactions, determine trust scores. FMN identifies and removes malicious fog nodes, promptly sharing their status with other FMNs through Software-Defined Networking (SDN) controllers. This proactive approach enhances network integrity.

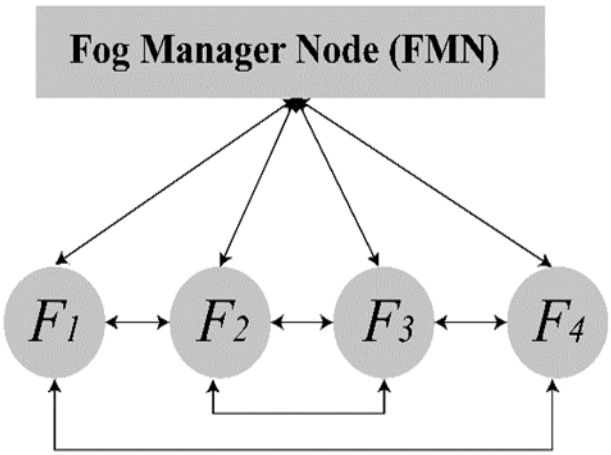

Figure 2: Fog Manager node and Fog Node.

3.6 Software-Defined Network (SDN) Controller

Our architecture combines IoT devices, fog nodes, SDN controllers, and cloud services. IoT devices communicate with FMN and other nodes, integrating seamlessly into the network. Fog computing involves parent and child nodes, with FMN assessing node reliability based on behavior.

3.7 Cloud Service Provider (CSP)

Extraneous data is either uploaded to or stored in the cloud, despite not undergoing computation, as cloud computing primarily handles substantial data calculations and processing. In this context, fog computing cannot match the capabilities of cloud computing. Fog computing enhances computational workflows by minimizing latency. A fog node can connect with the cloud directly as its level of trust rises, allowing data transmission and reception without the need for a fog manager node.

3.8 Access Control Management System

Access control involves FMN verifying new fog nodes before granting access. Malicious Fog Nodes List (MFL) is maintained by FMN based on trustworthiness. Nodes with trust levels exceeding a threshold are trusted, while nodes failing this criterion are considered malicious and added to MFL. FMN denies access to malicious nodes, updates MFL, assigns tasks to legitimate nodes, and informs the network. The inclusion of a node within the MFL is determined through a criterion expressed in Equation 1:

$$\text{Ac\_Control} = \begin{cases} 1 & \text{if } EV'(F_i) \geq \gamma \\ 0 & \text{else} \end{cases} \quad (1)$$

If the FMN detects that the new fog node is malicious, it denies the request and refrains from granting access to the network. If the new fog node doesn't have an ID, the fog head will assign one after verification. The FMN then broadcasts the freshly given fog node ID to all other network nodes. By doing this action, nearby fog nodes (NFN) can exchange services with the newly linked node.

Post this, the FMN designates a non-sensitive task to the new node, ensuring that its interactions do not negatively impact neighboring fog nodes initially. The complete process is outlined in Algorithm 1.

Algorithm 1: Fog node joining into the network and assigning a task.

Input: *FogManagerNode (FMN); NewFogNode (F_n); Maliciousfognode(MF_L)*

Parameters: *FogList (F_L);*

Result: *Give Access control and assign a task to NewFogNode (F_n)*

NewFogNode (F_n) will request *FogManagerNode (FMN)* for joining the system.

If $F_n \epsilon\ MF_L$ **then** ▷ *FMN* will check the ID in MF_L for verification F_n

declined ▷ *FMN* will remove the untrusted node

else

$F_n \leftarrow ID$ ▷ *FMN* will assign ID and task to the new fog node

$F_L \leftarrow ID(F_n)$ ▷ *FMN* will update the list & send the *Id* of F_n in the network

end

end

return;

End

Algorithm 1 outlines the sequence of steps for soliciting network access through an FMN. Let's explore the process through a scenario: when a fog node, denoted as Fn, aspires to join the network. Here's a breakdown of the steps:

In line 1, Fn initiates the procedure by forwarding a network access request to the FMN. Proceeding to lines 2-4, the FMN undertakes an assessment of Fn's ID and examines whether Fn is flagged within the malicious fog list. Should Fn's ID appear on this list, the FMN dismisses the request, denying access.

Moving to lines 5-7, upon confirming Fn's legitimacy, the FMN engages in assigning an ID and allocates a specific task to the fog node Fn. This encapsulates the scenario wherein a fog node endeavors to join the network, and Algorithm 1 facilitates the entire process.

3.9 Weighted Trust Management

FMN's monitoring extends to cloud-fog traffic. Weighted trust management has two segments: FMN assesses trustworthiness of fog nodes, and FMN updates trustworthiness of nodes reporting malicious and legitimate nodes.

Experimental Setup

Table 1: System setup and Simulation settings

Parameter	Value
Operating system	Win 10
Processor	Intel(R) Core (TM) i7-7500U CPU @ 2.70GHz 2.90 GHz
RAM	8.00 GB
System Type	64-bit operating system, x64-based processor
Simulation environment	iFogSim, Java jdk-8u241 and Eclipse-IDE
Number of fog nodes	20
Number of IoT devices	20
γ	2.0
θ	2.0
$\propto$	0.5
β	0.2

We evaluate the proposed framework designed to facilitate secure collaboration between Fog nodes (Fog-2-Fog collaboration). Establishing a reliable system to guarantee secure fog service requests is the main goal. We used a Java program to simulate our network topology in order to test the efficacy of our access control and trust evaluation technique. For simulation purposes, we employed iFogSim [18], a specialized simulator tailored for Fog and IoT environments. iFogSim serves as a reliable platform for managing IoT services within a Fog infrastructure.

4. Results and Discussion

This section conducts a comprehensive analysis of experimental and numerical results, confirming the efficacy of our access control management system based on the weighted trust model. The evaluation

focuses on operational efficiency and algorithmic efficacy through two extensive experiments, comparing varying parameter conditions.

4.1 Experiment and Trust Evaluation

In the initial experiment, with parameters set to x = 0.5 and y = 0.2, all fog nodes had initial trust and honesty values of 1. A threshold of 2.0 was maintained. The simulation confirmed the positive impact of our approach on network efficiency and confidence, as seen in Figs. 3 and 6, indicating the algorithm's accuracy in identifying malicious nodes. The experiment comprised two rounds: in the first, each fog node managed 25 service requests to establish collaboration and accurate trust values. The second round involved 10 fog nodes handling 250 requests, systematically analyzing interactions to detect malicious behavior.

Fig. 3 illustrates distinctive patterns among fog nodes. F2, F4, F5, F6, F8, and F10 consistently maintained high trust values, while F1, F3, F7, and F9 exhibited malicious behavior and were eventually removed. Node removal was due to trust value deterioration beyond the predefined threshold. Over time, both trust and honesty metrics increased, regulated by the proposed formula post each transaction. This dynamic process led to trust value surges for favorable behavior and drops for undesirable actions, ultimately expelling nodes with depleted trust values.

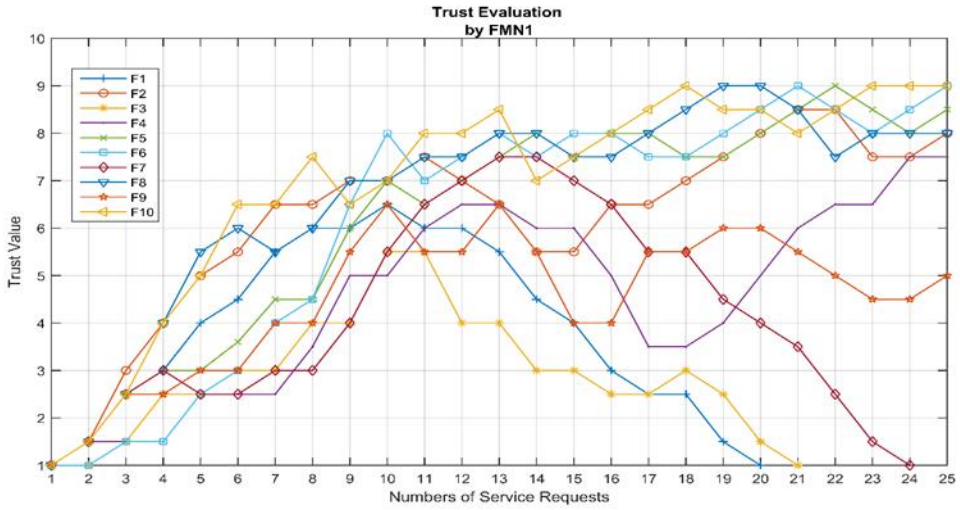

Figure 3: Trust Evaluation of Fog nodes by Fog manager node (FMN1)

4.2 Trust Dynamics and Ambiguity

In a subsequent experiment, we introduced a policy of trustworthiness ambiguity, featuring non-transitive and asymmetric trust values. The assessment of individual fog nodes by FMN1 showcased the intricate dynamics of trust. A visual representation in Figure 4 portrayed multidimensional trust ratings, highlighting varying confidence levels among fog nodes. Non-consistent transitivity became evident in Figure 5, where trust networks were disrupted due to revelations of unreliability. This experiment illuminated the complexities of trust evaluation and management among collaborative fog nodes.

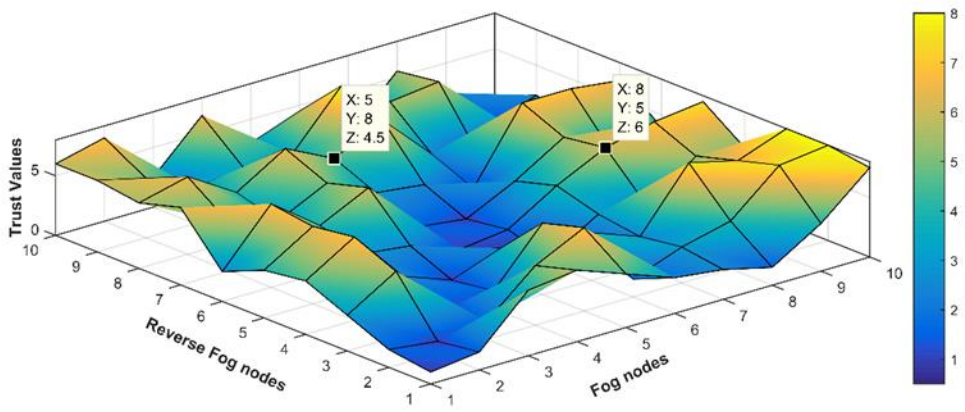

Figure 4: Trust evaluated by FMN1 for the 10 participated fogs against each other proven that Trust values is asymmetric

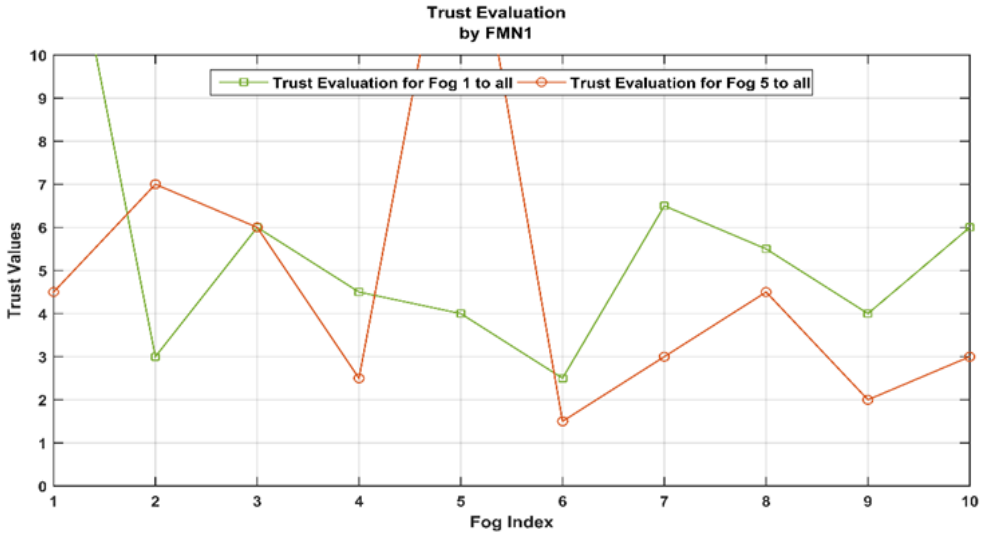

Figure 5: Trust evaluated by FMN1 for fog1 and fog5 proven that Trust values is not transitive.

Figure 6 depicted the trust dynamics within the network, showcasing certain fog nodes' consistent trust maintenance (F11, F13, F15, F16), reaching the highest levels of trust. Conversely, malicious activities of F12, F14, F18, and F19 led to their removal.

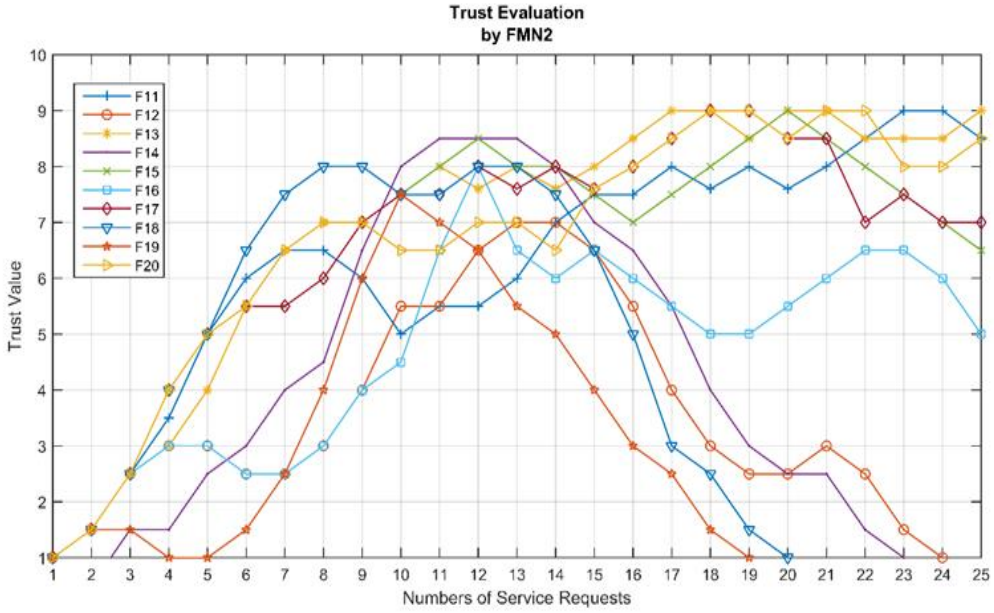

Figure 6. Overall Trust Evaluation of Fog nodes by FMN2

This dynamic was further demonstrated in Figure 7, a three-dimensional representation of acquired trust values among collaborative fog nodes. Various confidence levels were evident, such as the confidence value between F14 and F17 at 6.5, and between F17 and F14 at 7.5. However, the trust evaluation's transitivity was not consistent. This was exemplified in Figure 5, where F1 trusts F15, and Fb trusts F19, but F11's discovery disrupted this trust network due to F19's unreliability.

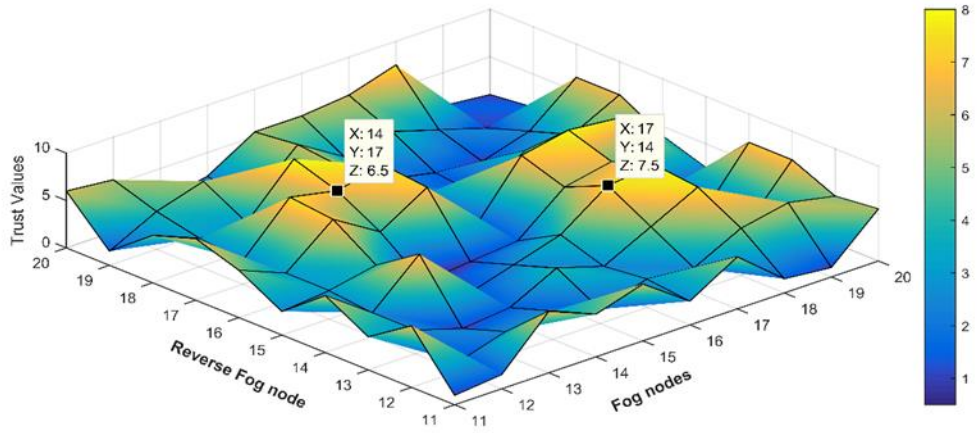

Figure 7: Trust evaluated by FMN2 for the 10 participated fogs against each other proven that their trust values are asymmetric.

Figure 8 presented a case where trust evaluation was not transitive. FMN2 evaluated trust for fog11 and fog15, showcasing the complexities of transitivity.

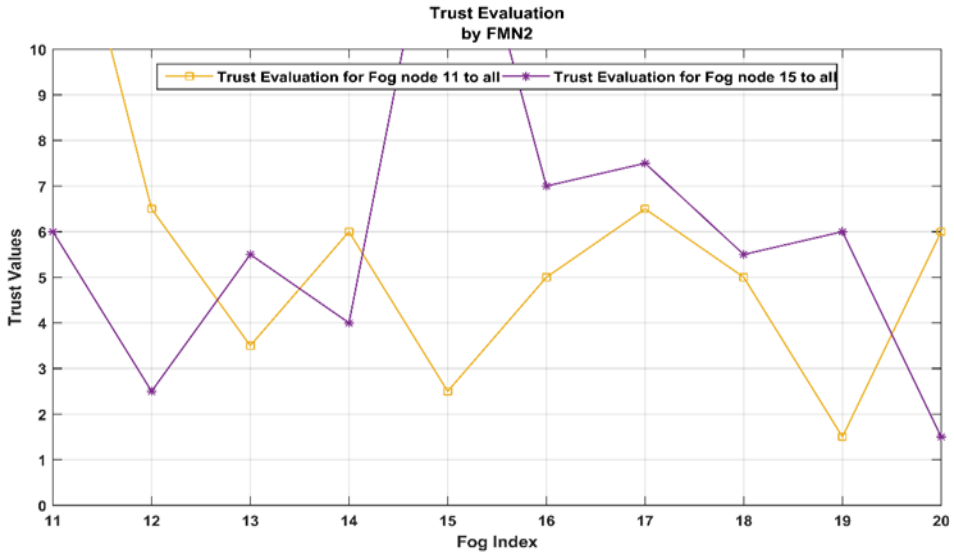

Figure 8: Trust evaluated FMN2 for fog11 and fog15 proven that Trust evaluation s not transitive.

4.3 Honesty Updating

After assessing trust, FMN recalibrated honesty attributes for interactors, balancing rewards and penalties for feedback authenticity. Figs. 9 and 10 depicted the transition of specific fog nodes to dishonesty due to erroneous feedback, while others gradually achieved high honesty through consistent sincere feedback. Notably, honesty ratings decreased for nodes offering misleading feedback.

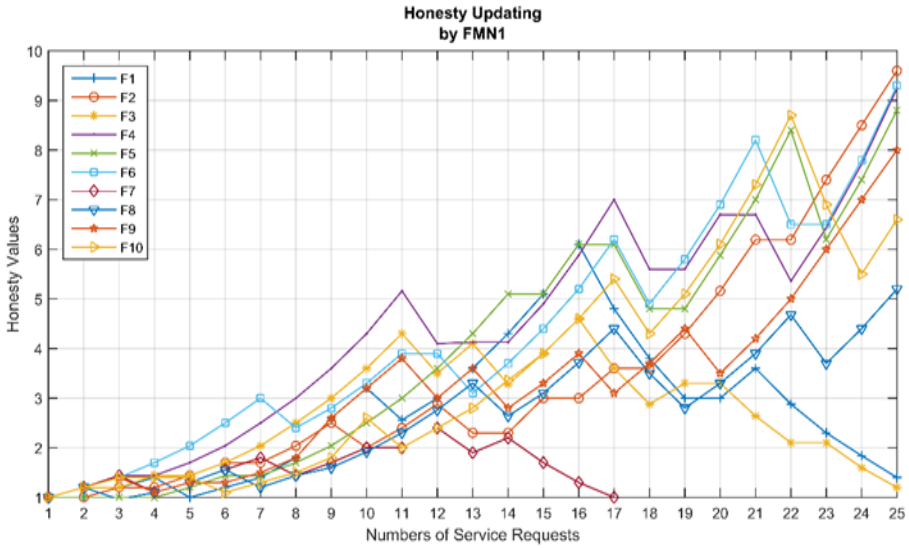

Figure 9: Honesty updating of Fog nodes by FMN1

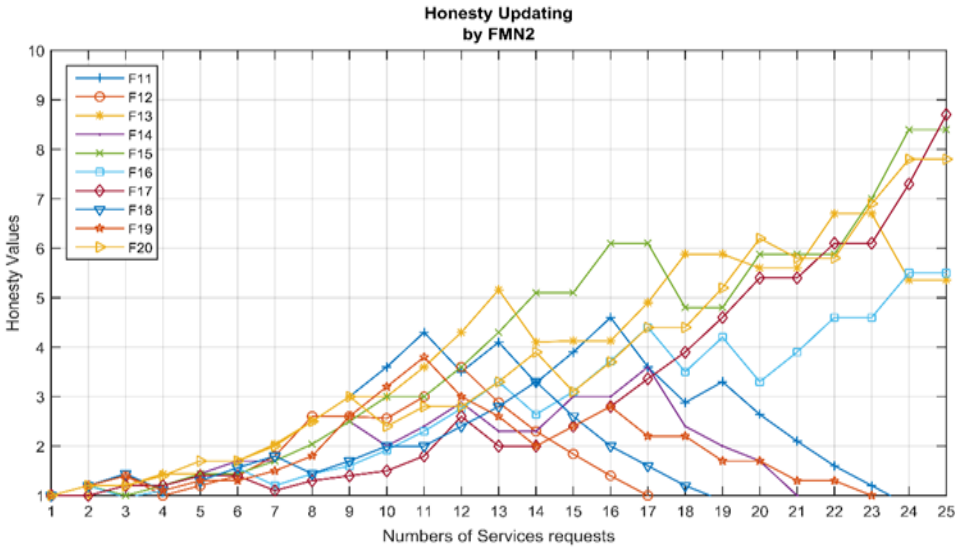

Figure 10: Honesty updating of Fog n**odes** by FMN2

Tables 2 and 3 below demonstrate the fog nodes' final honesty and trustworthiness as well as the eliminated malicious fog nodes.

Table 2: Fog nodes present in network with their ID, Honesty, and Trust level

Node ID	Node Trust	Node Honesty
F1	-	-
F2	8.0	9.6
F3	-	-
F4	7.5	9.2
F5	8.5	8.8
F6	9.0	9.2
F7	-	-
F8	8.0	5.2
F9	5.0	8.0
F10	9.0	6.0
F11	8.5	-
F12	-	-
F13	9.0	5.3
F14	-	-

F15	6.5	8.0
F16	5.0	5.5
F17	7.0	8.7
F18	-	-
F19	-	-
F20	8.0	7.8

The identifiers of malevolent fog nodes are stored within a registry of malicious fog nodes. This repository is utilized to thwart any attempts by these nodes to rejoin the network and make requests to the fog manager node. When such a request is initiated, the Fog Node Manager (FMN) cross-references the node's identifier with the entries in the malicious fog node list, subsequently rejecting the request if there's a match.

Table 3: Malicious Fog Node List (ML_F)

Malicious Node ID	Values
F1	0.0
F3	0.0
F7	0.0
F11	0.0
F12	0.0
F14	0.0
F18	0.0
F19	0.0

4.4 Access Control Management

Access control mechanisms efficiently rejected malicious fog nodes listed in the registry, causing a slight increase in access time due to enhanced security measures. Employing the malicious fog node list prevented reintegration attempts, maintaining network integrity.

Table 4: Example of the various values are reported by iFogSim

Fog ID	STATUS	FMN ID	Time
F1	FAILURE	-	-
F2	ACCEPT	1	2.5
F3	FAILURE	-	-
F4	ACCEPT	1	3.5
F5	ACCEPT	1	3
F6	ACCEPT	2	3.4
F7	FAILURE	-	-
F8	ACCEPT	2	3.18
F9	ACCEPT	2	2.6
F10	ACCEPT	2	2.3
F11	FAILURE	-	-
F12	FAILURE	-	-
F13	ACCEPT	2	2.4
F14	ACCEPT	-	-
F15	ACCEPT	1	3.4
F16	ACCEPT	1	2.52
F17	ACCEPT	2	3.01
F18	FAILURE	-	-
F19	FAILURE	-	-
F20	ACCEPT	1	2.82

5. Conclusion

Our research embodies a groundbreaking advancement in the realm of IoT-enabled networks. With the proliferation of advanced features such as processing prowess, mobility, and heightened discovery capabilities in IoT devices, the need for seamless connectivity to neighboring counterparts, gateways, and network components becomes paramount. Yet, this connectivity opens the door to potential vulnerabilities, where unscrupulous fog nodes might exploit unauthorized access, imperiling the integrity of the IoT network's functioning. Moreover, the looming threat of malevolent nodes manipulating data and undermining system performance accentuates the urgency of robust security measures.

In light of these challenges, our study presents a comprehensive security architecture founded upon an SDN-based fog computing infrastructure. This architectural innovation not only bolsters the efficiency of the fog layer but also harnesses the untapped potential of dormant resources within proximate smart devices. By introducing innovative access control and trust evaluation algorithms, we have paved the way for a paradigm shift in fog network security. Our architecture's real-world applicability is showcased through the tangible outcomes of simulations conducted within the iFogSim environment. This research's significance lies in its ability to address the intricate security concerns that arise in the burgeoning landscape of IoT devices, offering a promising avenue for fortifying network integrity while enabling the unhindered expansion and movement of IoT users.

6. References

[1] S. Tomovic, K. Yoshigoe, I. Maljevic, and I. Radusinovic, “Software-Defined Fog Network Architecture for

IoT," Wireless Personal Communications, vol. 92, no. 1, pp. 181–196, Jan. 2017, doi: 10.1007/S11277-016-3845-0/FIGURES/6.

[2] D. Shehada, A. Gawanmeh, C. Y. Yeun, and M. Jamal Zemerly, "Fog-based distributed trust and reputation management system for internet of things," Journal of King Saud University - Computer and Information Sciences, Nov. 2021, doi: 10.1016/J.JKSUCI.2021.10.006.

[3] F. Hosseinpour, A. S. Siddiqui, J. Plosila, and H. Tenhunen, "A Security Framework for Fog Networks Based on Role-Based Access Control and Trust Models," Lecture Notes in Business Information Processing, vol. 310, pp. 168–180, 2018, doi: 10.1007/978-3-319-94845-4_15.

[4] S. Kumar, P. Tiwari, and M. Zymbler, "Internet of Things is a revolutionary approach for future technology enhancement: a review," Journal of Big Data, vol. 6, no. 1, pp. 1–21, Dec. 2019, doi: 10.1186/S40537-019-0268-2/FIGURES/9.

[5] A. Diro, H. T. Reda, and N. Chilamkurti, "Differential flow space allocation scheme in SDN based fog computing for IoT applications," Journal of Ambient Intelligence and Humanized Computing, pp. 1–11, Jan. 2018, doi: 10.1007/S12652-017-0677-Z/FIGURES/11.

[6] Khakimov, I. Gudkova, A. A. Ateya, E. Markova, A. Muthanna, and A. Koucheryavy, "IoT-Fog based system structure with SDN enabled," ACM International Conference Proceeding Series, Jun. 2018, doi: 10.1145/3231053.3231129.

[7] M. Al-khafajiy et al., "COMITMENT: A Fog Computing Trust Management Approach," Journal of Parallel and Distributed Computing, vol. 137, pp. 1–16, Mar. 2020, doi: 10.1016/J.JPDC.2019.10.006.

[8] A. Hakiri, B. Sellami, P. Patil, P. Berthou, and A. Gokhale, "Managing Wireless Fog Networks using Software-Defined Networking," in 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), Oct. 2017, pp. 1149–1156. doi: 10.1109/AICCSA.2017.9.

[9] D. Shehada, C. Y. Yeun, M. Jamal Zemerly, M. Al-Qutayri, Y. Al-Hammadi, and J. Hu, "A new adaptive trust and reputation model for Mobile Agent Systems," Journal of Network and Computer Applications, vol. 124, pp. 33–43, Dec. 2018, doi: 10.1016/J.JNCA.2018.09.011.

[10] K. M. Sadique, R. Rahmani, and P. Johannesson, "Trust in Internet of Things: An architecture for the future IoT network," 2018 International Conference on Innovation in Engineering and Technology, ICIET 2018, Mar. 2019, doi: 10.1109/CIET.2018.8660784.

[11] S. Salonikias, I. Mavridis, and D. Gritzalis, "Access Control Issues in Utilizing Fog Computing for Transport Infrastructure," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 9578, pp. 15–26, 2016, doi: 10.1007/978-3-319-33331-1_2.

[12] G. Rathee, R. Sandhu, H. Saini, M. Sivaram, and V. Dhasarathan, "A trust computed framework for IoT devices and fog computing environment," Wireless Networks, vol. 26, no. 4, pp. 2339–2351, May 2020, doi: 10.1007/S11276-019-02106-3.

[13] K. N. Pallavi and V. Ravi Kumar, "Authentication-based Access Control and Data Exchanging Mechanism of IoT Devices in Fog Computing Environment," Wireless Personal Communications, vol. 116, no. 4, pp. 3039–3060, Feb. 2021, doi: 10.1007/S11277-020-07834-W/TABLES/2.

[14] P. Zhang, Z. Chen, J. K. Liu, K. Liang, and H. Liu, "An efficient access control scheme with outsourcing capability and attribute update for fog computing," Future Generation Computer Systems, vol. 78, pp. 753–762, Jan. 2018, doi: 10.1016/J.FUTURE.2016.12.015.

[15] Z. Yu, M. H. Au, Q. Xu, R. Yang, and J. Han, "Towards leakage-resilient fine-grained access control in fog computing," Future Generation Computer Systems, vol. 78, pp. 763–777, Jan. 2018, doi: 10.1016/J.FUTURE.2017.01.025.

[16] W. ben Daoud, M. S. Obaidat, A. Meddeb-Makhlouf, F. Zarai, and K. F. Hsiao, "TACRM: trust access control and resource management mechanism in fog computing," Human-centric Computing and Information Sciences, vol. 9, no. 1, pp. 1–18, Dec. 2019, doi: 10.1186/S13673-019-0188-3/FIGURES/11.

[17] S. Alharbi, T. Halabi, M. Bellaiche, P. Montréal, and S. al Harbi, "Fog computing security assessment for

device authentication in the internet of things," ieeexplore.ieee.org, Accessed: May 24, 2022. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9407860/

[18] K. S. Awaisi, A. Abbas, S. U. Khan, R. Mahmud, and R. Buyya, "Simulating Fog Computing Applications Using iFogSim Toolkit," Mobile Edge Computing, pp. 565–590, 2021, doi: 10.1007/978-3-030-69893-5_22.