City Research Online

City, University of London Institutional Repository

Citation: Junejo, A. K., Komninos, N. & McCann, J. A. (2021). A Secure Integrated Framework for Fog-Assisted Internet of Things Systems. IEEE Internet of Things Journal, 8(8), pp. 6840-6852. doi: 10.1109/jiot.2020.3035474

This is the accepted version of the paper.

This version of the publication may differ from the final published version.

Permanent repository link: https://openaccess.city.ac.uk/id/eprint/25202/

Link to published version: https://doi.org/10.1109/jiot.2020.3035474

Copyright: City Research Online aims to make research outputs of City, University of London available to a wider audience. Copyright and Moral Rights remain with the author(s) and/or copyright holders. URLs from City Research Online may be freely distributed and linked to.

Reuse: Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

City Research Online: http://openaccess.city.ac.uk/ publications@city.ac.uk

A Secure Integrated Framework for Fog-Assisted Internet of Things Systems

Aisha Kanwal Junejo, Nikos Komninos, *Member, IEEE*, and Julie A. McCann, *Member, IEEE*

Abstract—Fog-Assisted Internet of Things (Fog-IoT) systems are deployed in remote and unprotected environments, making them vulnerable to security, privacy, and trust challenges. Existing studies propose security schemes and trust models for these systems. However, mitigation of insider attacks, namely blackhole, sinkhole, sybil, collusion, self-promotion, and privilege escalation, has always been a challenge and mostly carried out by the legitimate nodes. Compared to other studies, this paper proposes a framework featuring attribute-based access control and trust-based behavioural monitoring to address the challenges mentioned above. The proposed framework consists of two components, the security component (SC) and the trust management component (TMC). SC ensures data confidentiality, integrity, authentication, and authorization. TMC evaluates Fog-IoT entities' performance using a trust model based on a set of QoS and network communication features. Subsequently, trust is embedded as an attribute within SC's access control policies, ensuring that only trusted entities are granted access to fog resources. Several attacking scenarios, namely DoS, DDoS, probing, and data theft are designed to elaborate on how the change in trust triggers the change in access rights and, therefore, validates the proposed integrated framework's design principles. The framework is evaluated on a Raspberry Pi 3 Model B to benchmark its performance in terms of time and memory complexity. Our results show that both SC and TMC are lightweight and suitable for resource-constrained devices.

Index Terms—Fog Computing, Internet of Things (IoT), Security, Trust and Credibility

1 Introduction

Next generation fog-assisted internet of things (Fog-IoT) systems extend the functionality, features and capabilities of fog computing and the Internet of Things (IoT). Such systems are suitable for IoT systems as services with reduced latency, network jitter, and bandwidth are located near the network's edge [1]. However, Fog-IoT systems face numerous security and trust challenges. Resource limited nature of IoT devices and lack of protection mechanisms make these systems an easy target for the attackers. The machine to machine (M2M) communication between sensors, actuators, fog nodes, and cloud services can be compromised via several attacks such as interception (i.e., eavesdropping), interruption (i.e., DoS attacks), modification (i.e., packet payload manipulation in transit) and fabrication (i.e., a man-in-the-middle attack). These attack mechanisms violate security goals namely, ***confidentiality***, ***integrity***, ***availability***, ***authentication***, and ***access control***.

Fog-based systems also face trust issues because, unlike cloud servers, fog nodes are located in remote and unprotected environments, making them vulnerable to tampering and node capture attacks. Fog computing features, namely location awareness, mobility, and decentralized architecture, also introduce trust challenges. With no centralized network, the fog nodes can join and leave the system frequently, mainly due to the inability to provide quality of service (QoS), load balancing, device compromise, or user/operator error. Such cases also raise concerns about secure handling of sensitive data, device logs and network access information. Additionally, the malicious nodes with otherwise legitimate identities can also impact the trust of Fog-IoT entities by reporting fabricated sensory data and routing information.

- *A. K. Junejo and Julie A. McCann are with the Adaptive Emergent System Engineering Group, Department of Computing, Faculty of Engineering, Imperial College London, London, SW7 2AZ, e-mail: ajunejo@imperial.ac.uk, j.mccann@imperial.ac.uk.*
- *Nikos Komninos is with the Department of Computer Science, School of Mathematics, Computer Science, and Engineering, City University of London, EC1V 0HB, London, nikos.komninos.1@city.ac.uk e-mail: nikos.komninos@city.ac.uk.*

Manuscript received April, 2020.

1.1 Motivation

Having discussed the security and trust challenges, it is clear that the establishment of trustworthy systems requires robust security and trust mechanisms. Data encryption, identity, and access management can prevent data modification and fake entities from joining a Fog-IoT system, but they cannot guarantee that entities have not been compromised. Without trust mechanisms, the interactions between fog nodes and IoT devices are subject to risk and uncertainty and, therefore, necessitate that they must have a certain level of trust in one another.

Following this, a few recent studies proposed for fog based IoT systems are discussed. Miranda et al. [2] propose a security framework for the detection and prevention of intrusions in software-defined networks. Hao et al. [3] propose a collaborative game theory-based security detection framework for IoT to model the confrontation between attacker and defender to identify security events. Moreover, collaborative information sharing is achieved by consensus protocol. Nabil et al. [4] propose a trustworthiness scheme for mitigating routing attacks, namely Rank and Blackhole, against RPL. The studies cited above [2]- [4] have various limitations, 1) security and trust challenges are addressed

separately and no study follows an integrated approach, 2) the security frameworks do not achieve the fundamental security goals, namely data confidentiality, integrity, authentication, and authorization, and 3) most of the trust models proposed for Fog-IoT systems do not consider a hostile environment. A few consider a hostile environment but do not propose appropriate mitigation strategies.

Compared to other studies, this paper proposes a framework featuring attribute-based access control and trust-based behavioural monitoring to address the challenges mentioned above. To the best of authors' knowledge, this is the first work which takes an integrated approach for Fog-IoT systems. The proposed framework consists of two key components, 1) the security component (SC) and 2) the trust management component (TMC). The SC ensures all entities of Fog-IoT, i.e. fog nodes and IoT devices, have unique identities and only authorized parties can access the fog resources. TMC evaluates their trustworthiness by computing trust based on a set of QoS and network communication features. Trust computed by TMC is subsequently integrated into SC. Precisely, trust is used as an attribute in access control policies to guarantee that only trusted entities can request fog services and collaborate with other entities in the system.

As part of the SC, a lightweight attribute based encryption (ABE) scheme is proposed to achieve the security goals i.e., ***data confidentiality***, ***integrity***, ***authentication***, and ***authorization***. TMC guarantees dependable behavior of the Fog-IoT entities by robust behavioral monitoring and trust computation. Additionally, a credibility model is designed as a subcomponent of TMC to countermeasure several attacks including but not limited to DoS (Denial of Service) , DDoS (Distributed Denial of Service), Reconnaissance, and data theft attacks, which aim to affect the accuracy of the trust computation process. Precisely, the integration enables the enforcement of access rights based on the performance of the Fog-IoT entities.

1.2 Contribution

The contribution of this paper is three-fold.

First, a secure integrated framework which addresses the security and trust challenges of Fog-IoT systems is proposed.

Second, trust is embedded as an attribute in the access control policies to elaborate on the integration of SC and TMC.

Third, the proposed framework's design principles are validated by modelling several attacking scenarios to elaborate on how the change in trust triggers the change in access rights. For instance, a fog node's access rights can escalate from "write and execute" to "All" if its trust has increased over time and vice versa. However, an increase/decrease in trust can be subject to malicious behaviour of compromised Fog-IoT entities; this is where the proposed credibility model plays its role and controls the rate of change of trust beyond a predefined threshold value. As a result, the access rights also do not change dramatically and therefore enable a dependable Fog-IoT system.

The rest of this paper is organized as follows. The related work is presented in section 2. The proposed secure integrated framework is elaborated in section 3. The experimental results are discussed in section 4. The conclusion and future work are presented in section 5.

2 Related Work

As the research in fog computing is in its infancy so there are very few frameworks. Nevertheless, some related frameworks, security schemes, and trust models are discussed below.

1. Frameworks: Soukup et al. [5] propose a hardware agnostic security framework for fog based IoT networks. It consists of collectors, detectors, and management tools to detect and mitigate the vulnerabilities in several IoT protocols, namely Z-Wave, Long Range Wide Area Network (Lo-RaWAN), and BLE. The study in [2] proposes a security framework for the detection and prevention of intrusions in software defined networks. The framework consists of an authentication scheme to prevent unauthorized entities from joining the network, complemented by a monitoring system to mitigate the intrusions reported by the sensor nodes. The study in [6] adopts a graph modelling approach to determine the relationship between vulnerabilities in an industrial IoT system where the security issues are formulated as graph-theoretic problems. The study also proposes some risk mitigation strategies to detect and remove the attack paths with high risk and low hop-length.

Muhammad et al. [7] propose a surveillance framework for IoT systems using probabilistic image encryption. The study uses a video stigmatization method to extract the information frames and further encrypts the images to prevent data modification attacks. Fadi et al. [8] propose a key agreement framework based on the mobile-sink strategy and extends user authentication to the cloud-based applications. It is based on bilinear pairing and elliptic curve cryptography (ECC). Alhanahnah et al. [9] propose a context-aware multifaceted trust framework for evaluating the trustworthiness of cloud services. Overall, trust in a cloud service is computed by considering both service characteristics and user experience. Service level agreement (SLA) based trust is computed by employing the analytic hierarchy process (AHP), whereas the non-SLA based trust is computed by fuzzy simple additive weighting (FSAW).

2. Security Schemes: Jian et al. [10] propose a three-party authentication and key agreement protocol for fog based IoT health systems. The protocol is based on bilinear pairing. The cloud authenticates fog nodes and IoT devices and generates a shared secret key for them. Yeh et al. [11] propose a variant of ciphertext-policy attribute based encryption (CP-ABE) scheme for eHealth systems. It employs fine-grained access control in cloud-based personal health care applications. Mick et. al [12] propose a lightweight authentication and secure routing scheme for smart cities. Mahmood et al. [13] propose an authentication scheme for smart gird. The major limitation of this scheme is the scalability in case of large-scale IoT systems. Some studies also propose proxy re-encryption schemes whereby fog nodes re-encrypt the ciphertexts and/or keys for end devices. Wang et al. [14] propose an ID-based proxy re-encryption scheme to secure the communication between end devices and cloud that is leakage resilient in an auxiliary input model. It follows a hybrid encryption approach wherein the data files are

encrypted using symmetric keys, while the symmetric keys are encrypted with the master public keys.

3. Trust Models: Lu et al. [15] propose a trust model for evaluating the trustworthiness of cloud services by combining objective and subjective evidence. The study in [16] employs fog computing to evaluate trustworthiness in sensor-cloud systems. It proposes a hierarchical trust computation mechanism to compute trust in the sensor network and between sensor service providers and cloud service providers. Ivan et al. [17] propose a technique for securing the Internet of Vehicles (IoV) systems. The security technique uses prioritization rules, digital certificates, and trust and reputation policies for detecting hijacked vehicles. Haddadou et al. [18] propose a job market signaling scheme for incentive and trust management in vehicular ad hoc networks. The scheme countermeasures the malicious nodes which aim to degrade the network by spreading false and forged data. Alemneh et al. [19] propose a TMS where both service requester and service provider can evaluate each other's reliability prior to any interaction. The trust model is based on subjective logic and mitigates malicious nodes' behaviour to manipulate the trust results. Esposito et al. [20] address the trust issues of IoT systems by proposing a game theoretic based model. A signaling game is modeled to avoid false reputation scores sent by compromised IoT devices. Liang et al. [21] propose a trust model for social sensor networks based on multisource feedback and fog computing. The sensor nodes give feedback about other nodes after an interaction, and fog nodes also monitor them and generate the feedback. The trust for each sensor device is computed by aggregating feedback from multiple sources.

3. Discussion: From the literature review, it is clear that none of the existing frameworks and/or studies adopts an integrated approach to simultaneously address the security and trust issues. The existing security schemes have several limitations. The generation, verification, and distribution of certificates in public key encryption (PKE) schemes incur heavy computation and communication overhead, which is not desirable for resource limited IoT devices. In large-scale networks, key generation and management in symmetric schemes are not trivial. Each node needs $n-1$ keys to communicate with other nodes. The existing ABE schemes based on bilinear pairing requiring large security parameters are computationally expensive and need a lot of processing and memory resources. The trust models proposed for fog based IoT systems do not consider a hostile environment wherein the malicious nodes can join the network and carry out cyber attacks, namely blackhole, sinkhole, Sybil, collusion, self-promotion, bad-mouthing, ballot-stuffing, and providing opportunistic service. The compromised entities can affect the accuracy of trust model by increasing/decreasing the trust and, therefore risk the credibility and trustworthiness of Fog-IoT systems. The challenges mentioned above underline that both security and trust challenges must be simultaneously addressed to protect open and distributed systems.

3 Proposed Integrated Framework

In this section, we present our proposed integrated framework for Fog-IoT systems.

TABLE 1: Notations and their Meanings

Notation	Description
$\mathbb{U}$	Attribute Universe
$\mathbb{A}$	Device Attribute Set
$\mathbb{P}$	Access Policy
MPK/MSK	Master public and secret key pair
k_u	Device secret key
CT	Ciphertext
M	Message
n	time window for trust evaluation
t	time instance for QoS evidence gathering
$\mathbf{T}_t(f_i)$	instant trust of ith fog node at time t
$\mathbf{T}_{fo\rightarrow fog}$	Fog node objective trust computed by FO
$\mathbf{T}_{iot\rightarrow fog}$	IoT Device trust based
$\mathbf{T}_{fog}$	Fog node trust
α	weight for $\mathbf{T}_{iot\rightarrow fog}$
$c(i,f_j)$	parameters sent by ith CPS device for jth fog node
σ	standard deviation of $\mathbf{T}_{iot\rightarrow fog}$ over a time window n

3.1 System Model

As shown in Fig. 1, a general Fog-IoT system consists of three layers, namely, ***IoT Devices***, ***Fog Nodes***, and ***Cloud***. The ***IoT Devices*** layer consists of smart devices, namely smart meters, smart refrigerators, smart garage doors, health monitoring devices, weather monitoring systems, smart lights, and smart thermostats. The ***Fog Nodes*** layer consists of network equipment, such as routers, bridges, gateways, switches, and base stations, augmented with the computational capability and local servers (e.g., industrial controllers, embedded servers, mobile phones, and video surveillance cameras). The fog layer has two types of entities: fog nodes and ***Fog Orchestrator*** (FO) nodes. The FO nodes are dedicated to service orchestration, trust management, and robust identity and access management. FO nodes also manage the proposed integrated framework. The ***Cloud*** layer is a consolidated computing and storage platform that provides various applications for the acquisition, processing, presentation, and management of the Fog-IoT systems.

3.2 Threat Model

A threat model is considered whereby Fog-IoT entities are divided into three categories, honest, semi-honest, and dishonest. The cloud is honest but curious. It extends the services of fog nodes and follows the protocol specification in general but, at the same time, gathers information about services, users and location, etc. The fog nodes, FO, and IoT devices are considered semi-honest as they can be compromised and vulnerable to security and trust attacks. Following this, the security threats are discussed. Unencrypted messages can be intercepted and modified to violate confidentiality and integrity. Likewise, a few compromised fog nodes can be exploited for carrying out botnet attacks, namely DoS, DDoS, probing, and data theft. The DoS and DDoS attacks violate the availability and prevent the legitimate nodes from providing services, and in that case, IoT devices consider them as untrustworthy. Additionally, malicious fog nodes can let multiple compromised IoT devices register with the same identity (Sybil attack) and escalate/deescalate their access privileges.

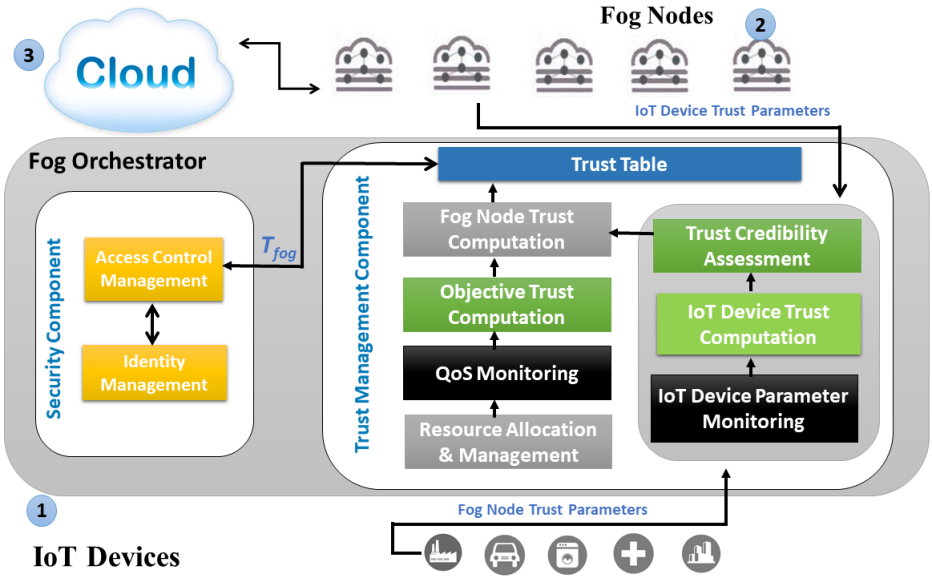


Fig. 1: A Secure Integrated Framework for Fog-IoT Systems

It is essential to compute the trust of each entity in the Fog-IoT system. However, the compromised nodes can corrupt the trust model by reporting false parameters, which, therefore, necessitates to evaluate trust credibility. In a self-promotion attack, malicious bot nodes can report false parameters to increase their trust. Similarly, they report false parameters about peers in a bad-mouthing attack. IoT devices can follow the same attacking strategy to decrease the trust of fog nodes. In opportunistic and on-off attacks, fog nodes can provide good/bad quality of service based on their resources. Like DoS, in collusion attacks, an adversary can create a botnet to modify the trust of target nodes. In a Sybil attack, a fog node can create several fake IDs to report fabricated parameters. The ballot stuffing attack occurs when a device submits more parameters than permitted in a given time period. The immediate effect of the above attacks is the imprecise trust computation, which contradicts nodes' real performance and requires effective detection and mitigation techniques for trust computation.

The components of the proposed framework are discussed in the following paragraphs. Table 1 lists the notation used throughout the paper.

3.3 Security Component

The security goals, namely ***data confidentiality***, ***integrity***, ***authentication***, and ***authorization*** can broadly be categorized as an identity and access management problem. So the first component of the proposed framework is designed by considering these aspects. The SC component consists of two subcomponents, namely Identity Management and Access Control Management.

3.3.1 Identity Management

In Fog-IoT systems, it is essential that all devices, sensors, monitors and fog nodes must have trusted and unique identities. Trusted identities would guarantee that all IoT devices and fog nodes are authenticated. In the proposed secure integrated framework, any entity which needs to be the part of a Fog-IoT system must first register with FO based on a set of attributes. The attributes are further employed in an ABE scheme to generate the secret keys which are subsequently used to encrypt the information.

3.3.2 Access Control Management

In Fog-IoT systems, access control management is required to determine which entity (a device or a user) can access what resources, such as read or write data, execute programs, and control actuators. In Fog-IoT systems, both FO and fog nodes can define access policies for IoT devices and other nodes. Any device which fulfills the access policy can request and/or provision resources. In the proposed framework, the objectives of secure identity management, authentication and access management are achieved by employing lightweight cryptographic techniques. To be more specific, as a part of the SC, an existing ABE scheme [22] based on elliptic curve cryptography (ECC) is implemented to enforce robust authentication, and access control in Fog-IoT systems. From here onwards, [22] is referred to as ECC-CPABE scheme. ECC-CPABE has four algorithms namely, *Setup*, *KeyGen*, *Encrypt* and *Decrypt*.

1. $Setup(\lambda, \mathbb{U}) \rightarrow MPK/MSK$ - This algorithm is executed by the FO. It takes as input a security parameter λ and an attribute universe $\mathbb{U}$ and generates the master public and secret MPK/MSK key pair.
2. $KeyGen(MPK, MSK, \mathbb{A}) \rightarrow k_u$ - The secret key k_u of a Fog-IoT entity is generated by FO based on an attribute set $\mathbb{A}$. This algorithm takes as input master public/secret key pair MPK/MSK and the attribute set $\mathbb{A}$ and outputs the secret key k_u which is subsequently used to decrypt the messages. The generation of secret keys by FO guarantees that all devices are authenticated and legitimate.
3. $Encrypt(MPK, \mathbb{P}, M) \rightarrow CT$ - This algorithm takes as input MPK, $\mathbb{P}$ and a plaintext message M and outputs a ciphertext CT. The CT is generated based on an access policy. The ***data confidentiality*** of the communication is achieved by encrypting the messages exchanged between

Fog-IoT entities.

4. $Decrypt(CT, \mathbb{P}, k_u) \rightarrow M$ - This algorithm takes as input a ciphertext CT, access policy and a secret key k_u. An entity can only decrypt the CT if its k_u is generated based on the attributes which are required in $\mathbb{P}$ i.e. $\mathbb{P} \subseteq \mathbb{A}$. The attributes in $\mathbb{A}$ and $\mathbb{P}$ enables the enforcement of robust ***authentication*** and ***authorization*** in Fog-IoT systems. In addition to above algorithms, any hash function could be used in a digital signature to ensure the ***integrity*** of the messages.

TABLE 2: Attributes of Fog-IoT entities and Trust Integration

IDs	Access Control Rights	Trust	Attribute
IoT01	None	<0.3	000
IoT02	Read	>= 0.3 - 0.4	001
IoT03	Write	> 0.4 - 0.5	010
FN01	Delete	> 0.5 - 0.6	011
FN02	Read and Execute	> 0.6 - 0.7	100
FN03	Modify (Permissions)	> 0.7 - 0.8	101
FN04	Special Permisssion	> 0.8 - 0.9	110
FN05	All	> 0.9 - 1	111

3.3.3 Integration of Trust in ABE Access Control Policies

In this section, the embedment of trust as an attribute in the access control policy is discussed. Table 2 describes a few attributes which can be used for identity and access management in SC. The columns **IDs** and **Access Control Rights** list the identities and the access rights of fog nodes and IoT devices. Likewise, the columns **Trust** and **Attribute** describe the trust and its corresponding representation required for the access rights granted to a Fog-IoT entity. It can be observed that change in trust is triggering a change in access rights; for instance, an increase/decrease in trust is resulting in privilege escalation/deescalation respectively.

Let $\mathbb{U}$, $\mathbb{A}$ and $\mathbb{P}$ be the attribute universe, device attribute set and access policy (AND-Gate) respectively. Both $\mathbb{A}$ and $\mathbb{P}$ are represented with an n-bit string. For brevity of the expression, both attribute set and its string representation are denoted by $\mathbb{A}$ and same applies to $\mathbb{P}$. For example, if $\mathbb{U} = \{A_1, A_2, A_3, A_4, A_5, A_6\}$ and an IoT device has $\mathbb{A} = \{A_1, A_2, A_3\}$ then it is represented as $\mathbb{A} = 111000$. With regard to embedding trust in $\mathbb{A}$ and $\mathbb{P}$, its binary representation (i.e. column **Attribute** in Table 2) corresponding to an access right is added to the device attribute set and access policy. Let us suppose A_6 attribute now represents the access rights of an IoT device which can "Write" (i.e. 010) then its n-bit string would be denoted as $\mathbb{A} = 11100010$. If $\mathbb{P}$ is defined over A_1, A_2, A_3, A_6 and $A_6 = 001$ which is "Read" access (i.e. 001) then it is denoted as $\mathbb{P} = 11100001$. It is maintained that an IoT device with attribute set $\mathbb{A}$ fulfills the access policy $\mathbb{P}$, if and only if $\mathbb{P} \subseteq \mathbb{A}$. The device access rights are granted based on its trust value that must be greater than or equal to the access policy; otherwise, the access is denied. The simplest example of access denial could be the inability to decrypt the ciphertext.

3.4 Trust Management Component

In this section, we discuss the second component of the proposed framework. TMC is adopted from the authors' previously published work in [23]. The TMC component consists of several subcomponents, namely *Trust Table*, *Fog Node Trust Computation*, *IoT Device Trust Computation*, *QoS Monitoring*, *IoT Device Parameter Monitoring* and *Resource Allocation, and Management*. The different sub-components of TMC are interdependent and enable the accurate and precise computation of the trust results. Through TMC, FO computes the trust between various Fog-IoT entities.

Resource allocation and management subcomponent is responsible for entertaining the service requests sent by different IoT devices and other Fog-IoT components. Moreover, for trust computation, the QoS and network communication parameters are monitored by *QoS monitoring* and *IoT device parameter monitoring* subcomponents. Subsequently, the trust for fog nodes and IoT devices are computed by *Fog Node Trust Computation* and *IoT Device Trust Computation* subcomponents.

3.4.1 Trust Table

After computing the trust for fog nodes and IoT devices trust, FO stores them in the *Trust Table*. Any entity can query FO to get the trust of other fog nodes and IoT devices. That would, in turn, help in deciding whether to collaborate with the specific entity or not.

3.4.2 Resource Allocation and Management

The resource allocation and management subcomponent takes a set of requirements as input and subsequently assigns service requests to available fog nodes. It also orchestrates the distributed services in ***Fog Nodes*** layer. FO selects highly trusted fog nodes based on their trust values.

TABLE 3: Fog Node Trust Parameters

Trust Parameters for Fog Nodes and IoT Devices
Throughput
Bandwidth
Energy Consumption
Transaction Duration
No: of inbound connections per source IP address
No: of inbound connections per destination IP address

3.4.3 QoS Monitoring

FO can employ different methods to monitor the run-time QoS (Table 3) parameters of the fog nodes. The service quality parameters can then be retrieved by either pull or push technique depending upon the Fog-IoT use case. Moreover, it is essential that FO must guarantee that only trusted fog nodes are deployed and they are maintaining an acceptable service quality abiding by the service level agreement (SLA).

3.4.4 IoT Device Parameter Monitoring

An IoT device parameter monitoring subcomponent is installed in each IoT device and sensor node to monitor network communication parameters. This subcomponent enables the IoT devices to quantify how much of their resources, namely energy, response time, and bandwidth, are being utilized for communicating with a particular fog node. The parameters are further compared with a set of predefined values to ensure that IoT devices and/or fog nodes are reachable and functioning correctly.

3.4.5 Fog Node Trust Computation

For trust computation of fog nodes, two sources of evidence, namely QoS parameters and network communication features, similar to [23], are used. Precisely, the trust of a fog node $\mathbf{T}_{fog}$ is computed by aggregating the trust computed from QoS evidence monitored by FO and the network communication parameters sent by IoT devices. The trust computed by QoS parameters is referred to as the "Objective" trust denoted by $\mathbf{T}_{fo\rightarrow fog}$ whereas the trust computed by the parameters sent by the IoT devices is referred to as "IoT Device" trust denoted by $\mathbf{T}_{iot\rightarrow fog}$. In both these cases, instant trust degree $\mathbf{T}_t$ i.e., trust at a time instance t is predicted by employing a random forest regression model [24]. In prediction problems, the regression models are trained over a substantial number of samples to improve the accuracy. It then compares the quantified parameters with the already stored SLA values and based on which computes the trust of fog nodes.

- Objective Trust Computation

FO monitors the CPU frequency, memory size, hard disk capacity, average response time, and task success ratio of each fog node. $\mathbf{T}_{fo\rightarrow fog}$ is computed after a specific time window n (i.e., one hour, day, or month). n can be divided into smaller time instances t. FO monitors the fog nodes at each time instance t based on the QoS parameters and predicts the instant trust degree $\mathbf{T}_t(f_i)$ by employing a random forest regression model. Subsequently, the objective trust $\mathbf{T}_{fo\rightarrow fog}$ over a time window n is computed using equation (1):

$$\mathbf{T}_{fo\rightarrow fog} = \bar{\mathbf{T}} \times w(f_i) = \sum_{t=1}^{n} (\mathbf{T}_t(f_i) \times w_t(f_i)), \quad (1)$$

where f_i is the ith fog node, $w_t(f_i) \in [0,1]$ is the weight assigned to each instant trust degree $\mathbf{T}_t(f_i)$ and is calculated by equation (2).

$$w_t(f_i) = \frac{(1-(t+1)^{-1})}{\sum_{t=1}^{n}(1-(t+1)^{-1})} \quad (2)$$

$w(f_i) = \{w_1(f_i), w_2(f_i), ..., w_n(f_i)\}$, and $\sum_{t=1}^{n} w_t(f_i) = 1$.

w is a time-based attenuation function which assigns more weight to $\mathbf{T}_t(f_i)$ computed at a recent time instance. As the fog nodes parameters change over time so the recently monitored parameters and subsequently $\mathbf{T}_t(f_i)$ is given more weight.

- IoT Device Trust Computation

Like $\mathbf{T}_{fo\rightarrow fog}$, the instant trust degree $\mathbf{T}_t(d_i)$ of IoT device d_i at a time instance t is predicted based on a set of network communication features. $\mathbf{T}_t(d_i)$ is predicted by a random forest regression model. Subsequently, $\mathbf{T}_t$ from all devices is averaged to calculate $\mathbf{T}_{iot\rightarrow fog}$ using Equation (3)

$$\mathbf{T}_{iot\rightarrow fog} = \frac{\sum_{i=1}^{p} \mathbf{T}_t(d_i)}{p} \quad (3)$$

where $\mathbf{T}_t(d_i)$ is the trust predicted from the ith set of network communication features sent by the IoT devices provisioning services from a fog node f_j and p is the total number of parameter sets. Note that $\mathbf{T}_{iot\rightarrow fog}$ is calculated by obtaining a simple average over all instant degree trust $\mathbf{T}_t(d_i)$ for a fog node. No weight is assigned to $\mathbf{T}_{iot\rightarrow fog}$.

Finally, the fog node trust is calculated using equation 4:

$$\mathbf{T}_{fog} = \alpha \times \mathbf{T}_{iot\rightarrow fog} + (1-\alpha) \times \mathbf{T}_{fo\rightarrow fog} \quad (4)$$

where $\alpha \in (0,1)$ is the weight of $\mathbf{T}_{iot\rightarrow fog}$, and, $(1-\alpha)$ is the weight of $\mathbf{T}_{fo\rightarrow fog}$. If α is assigned a value of 1, the weight of $\mathbf{T}_{fo\rightarrow fog}$ becomes 0 and equation 4 will only consider $\mathbf{T}_{iot\rightarrow fog}$ trust. In other words, the evidence of IoT devices is employed in trust computation of fog nodes, which might not be a correct evaluation of fog nodes' performance. It is recommended that α should be chosen after careful analysis of fog nodes and IoT devices. The credibility of $\mathbf{T}_{iot\rightarrow fog}$ trust is evaluated based on the model discussed in subsequent subsection.

3.4.6 Trust Credibility Assessment

As the Fog-IoT systems are located in open and unprotected environments, they are vulnerable to several security attacks as discussed in section 3.2. A trust credibility assessment model is designed to mitigate the attacks and to guarantee accurate and precise trust computation. The credibility model adjusts the trust of Fog-IoT entities in three cases whereby the IoT devices, fog nodes, and FO could be compromised. Trust credibility evaluation is applied in all computations i.e. $\mathbf{T}_{fog}$, $\mathbf{T}_{iot\rightarrow fog}$ and $\mathbf{T}_{fo\rightarrow fog}$, and "large" differences are adjusted. It is underlined that when new IoT devices and fog nodes are taking part in the network their few initial trust values are expected to be 0.5. Trust increases/decreases as the network operates. Trust credibility evaluation model analyses the change in $\mathbf{T}$ during consecutive time instances $[t_0, t]$ and subsequently recomputes the trust in recent time instance t using Eq. (5):

$$\mathbf{T}_t = \mathbf{T}_{t_0} \pm \sigma \mathbf{T}_t, \quad (5)$$

where σ is the standard deviation in $\mathbf{T}$ over a time window n. The standard deviation σ informs about the spread of the possible values of trust. σ is computed by Eq. (6):

$$\sigma = \sqrt{\frac{\sum(\mathbf{T}-\mu)^2}{n}} \quad (6)$$

where μ is mean of trust $\mathbf{T}$ at a time instance t. The standard deviation should be considered for every newly calculated trust value. If the trust $\mathbf{T}$ in recent time instance t is less than the previous time t_0 and the difference is greater than σ, then $\mathbf{T}$ in t is increased, otherwise it is decreased.

3.4.7 IoT Device Trust Computation

This subcomponent computes the trust of IoT devices based on the parameters monitored by the fog nodes. The fog nodes also report parameters to FO, which compute the trust for each IoT device using Equation 3.

4 Experimental Results and Discussion

Two experiments are designed to evaluate the effectiveness of the proposed integrated framework. In the ***first*** experiment, the ECC-CPABE scheme in SC component is evaluated. In the ***second*** experiment, the TMC is evaluated by computing the trust of fog nodes and IoT devices. Additionally, in the ***second*** experiment, integration of two components is elaborated by embedding trust of an entity in an access policy and subsequently describing its corresponding access control rights.

System Configuration - The proposed framework is evaluated on Raspberry Pi 3 model B+. It has Quad Core 1.2GHz, 64 bit CPU, 1 GB of RAM, a wireless LAN and Bluetooth Low Energy (BLE) on board, 100 Base Ethernet, 40-pin extended GPIO, 4 USB ports, HDMI and micro SD port. Charm crypto [25], Octave [26], and a few machine learning libraries were installed for implementing SC and TMC, respectively. Charm crypto library [25] is used for the implementation of ABE schemes. GNU Octave is a high-level language similar to Matlab and used for executing the trust model. Random forest regression model is implemented using the "sklearn" python library.

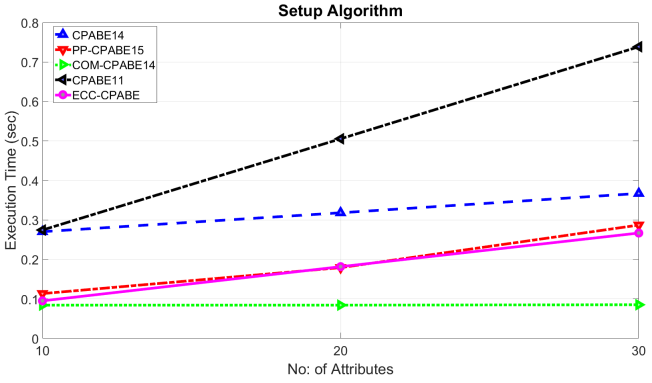


Fig. 2: Processing time (sec) for Setup Algorithm

4.1 First Experiment - SC Evaluation

As mentioned in section 3.3, the identity and access management (SC component) in Fog-IoT systems is implemented by employing an existing ABE scheme [22]. The algorithmic efficiency of ECC-CPABE [22] for lightweight IoT devices is measured in terms of time and memory complexity. This scheme is also compared with other ABE schemes, PP-CPABE15 [27], CPABE14 [28], COM-CPABE14 [29], and CPABE11 [30]. All of the above mentioned schemes are based on a selective security model in which the adversary submits the access policy to the challenger before seeing the secret key.

4.1.1 Implementation and Evaluation

Three of the schemes [27], [29], [30] have been tested on the super-singular SS512 curve, whereas [28] and ECC-CPABE [22] are tested on the non super-singular asymmetric bilinear curve (i.e. MNT159) provided by Charm. Both SS512 and MNT159 curves provide 80-bit security.

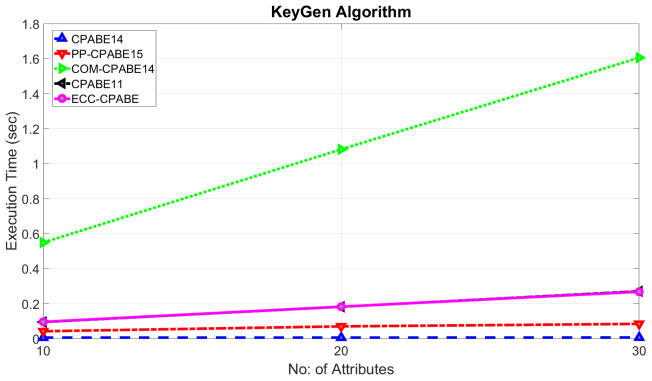


Fig. 3: Processing time (sec) for KeyGen Algorithm

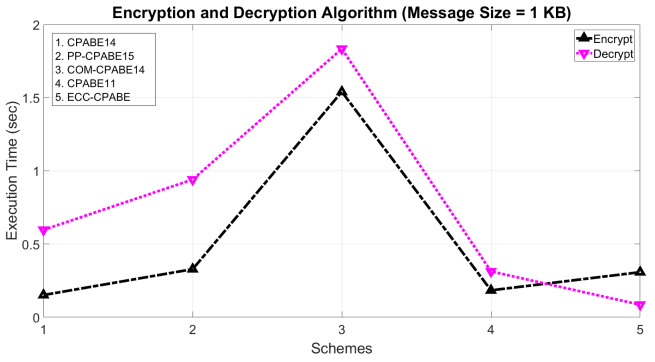


Fig. 4: Processing time (sec) for Encrypt and Decrypt (1 KB) Algorithms

4.1.2 Timing Results

The execution times of all algorithms are recorded to compare the efficiency of different schemes. Timing results are shown in Figures 2 - 5. For $Setup$ and $KeyGen$, three different sizes of attribute universe $\mathbf{U}$ and $\mathbf{A}$ are considered. Precisely, an attribute universe $\mathbf{U}$ of 10, 20 and 30 attributes is considered for measuring the timing of $Setup$ Algorithm. Likewise, for secret key generation, a user attribute set $\mathbf{A}$ of 5, 15, and 25 attributes is considered. Moreover, two types of benchmarks are set for measuring the times for encryption and decryption, 1) 1 KiloByte (1 KB), and 2) 1 MegaByte (1 MB). Due to the limited resource nature of IoT devices, low size messages are chosen for encryption and decryption.

Furthermore, in the encryption algorithm, an access policy $\mathbf{P}$ of constant size i.e., five attributes is considered in all cases. The time required by the $Setup$ algorithm over attribute universe of 10, 20, and 30 attributes in all schemes is shown in Fig. 2. It can be observed that COM-CPABE14 is the fastest, followed by PP-CPABE15 and ECC-CPABE because there are no pairing operations in its $Setup$ Algorithm. CPABE11 is the slowest of all schemes. ECC-CPABE takes 0.09, 0.18 and 0.26 seconds over $\mathbf{U}$ of 10, 20, and 30 attributes. Overall, the $Setup$ algorithm in all schemes is fast. Fig. 3 lists the execution time of the $KeyGen$ Algorithm in all schemes. It can be observed that this time CPABE14 is the fastest compared to all other schemes. It only takes 0.005 seconds for secret key generation over 10, 20 and, 30 attributes. ECC-CPABE takes 0.1, 0.2, and 0.27 seconds for 10, 20, and 30 attributes. Moreover, it can be

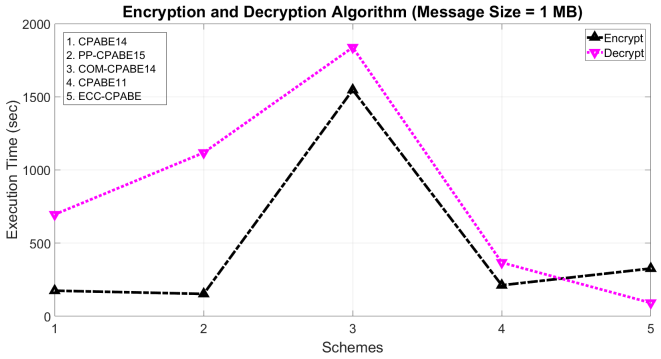


Fig. 5: Processing time (sec) for Encrypt and Decrypt (1 MB) Algorithms

observed that both ECC-CPABE and CPABE11 take equal time for secret key generation. COM-CPABE14 is the slowest of all schemes.

The execution timings of encrypting and decrypting message size of 1 KB are shown in Fig. 4. CPABE takes 0.15 seconds, CPABE11 takes 0.18 seconds, ECC-CPABE takes 0.3 seconds, PP-CPABE takes 0.33 seconds, and COM-CPABE14 takes 1.5 seconds. COM-CPABE14 is the slowest of all schemes. Additionally in decryption, ECC-CPABE is the fastest of all schemes as it only takes 0.08 seconds to decrypt a message of 1 KB, followed by CPABE11, which takes 0.31 seconds. CPABE14 and PP-CPABE15 take 0.6 and 0.94 seconds, respectively. COM-CPABE14 takes the longest time i.e., 1.84 seconds compared to other schemes. Fig. 5 shows the timing results when a message of 1 MB is encrypted and decrypted. Comparing figures 4 and 5, it is clear that all the schemes follow the same timing trend for encryption/ decryption of 1 KB and 1 MB messages. The timings for 1 MB message are just a multiple of 1 KB message.

TABLE 4: Benign and Botnet Attacking Scenarios

Benign Scenarios	Smart home with a weather station, smart fridge, smart lights, smart garage, and smart thermostat	
	Protocols - ARP, IPv6-ICMP, TCP and UDP	
Botnet Scenarios	**Categories**	**Subcategories**
	DoS, DDoS	UDP, TCP, and HTTP
	Probing	Port scanning, OS fingerprinting
	Information Theft	Data theft, Keylogging

4.2 Second Experiment - TMC Evaluation

This experiment is designed to achieve three objectives, 1) computing trust for Fog-IoT nodes based on the proposed TMC, 2) demonstrating the effectiveness of the credibility model to compute precise and accurate trust and 3) integration of SC and TMC components.

4.2.1 Experimental Setup

For evaluating TMC, the BoT-IoT dataset generated by Nickolaos et al. [31] is used. The testbed for dataset consists of several VMs, including both legitimate Ubuntu and attacking Kali Linux machines. IoT sensors are simulated on Ubuntu VMs using Node-red, which were connected with the public IoT hub, AWS. Java scripts are written on the Node-red for subscribing and publishing IoT services to the IoT gateway of the AWS via the Message Queuing Telemetry Transport (MQTT) protocol.

The dataset is quite large (approximately 18 GB), consisting of 73360900 rows. It has 47 features, including protocol, source and destination IP addresses, port numbers, packet counts, bytes, data rate, and traffic categories, etc. Smart home with five services, namely, Smart Refrigerator, Smart Garage door, Weather Monitoring System, Smart Lights, and Smart thermostat, is simulated to model a benign scenario. Several botnet scenarios, namely, DoS, DDoS, probing, and information theft are modelled. DoS and DDoS are modelled for UDP, TCP, and HTTP traffic. For more details, interested readers are encouraged to read [31]. The benign and botnet scenarios are listed in Table 4.

4.2.2 Trust Parameters

As discussed in section 3.4, the trust of fog nodes is computed by aggregating $\mathbf{T}_{fo \rightarrow fog}$ and $\mathbf{T}_{iot \rightarrow fog}$ based on the parameters listed in Table 3. $\mathbf{T}_{fo \rightarrow fog}$ is calculated from the features in the dataset. For instance, the 'dur' and 'rate' features are used for transaction duration and throughput. Bandwidth is calculated from 'rate' using the Nyquist formula, and likewise, energy is calculated for each transaction. Additionally, two features 'number of inbound connections' for each source and destination IP address are also considered as these were useful in the classification of botnet attacks, notably DoS and DDoS. As IoT services are running on the Ubuntu VMs, so the parameters required for $\mathbf{T}_{iot \rightarrow fog}$ cannot be measured. So, $\mathbf{T}_{iot \rightarrow fog}$ parameters are generated based on the distribution of the respective features in the dataset, and are subsequently used as features in the random forest regression model.

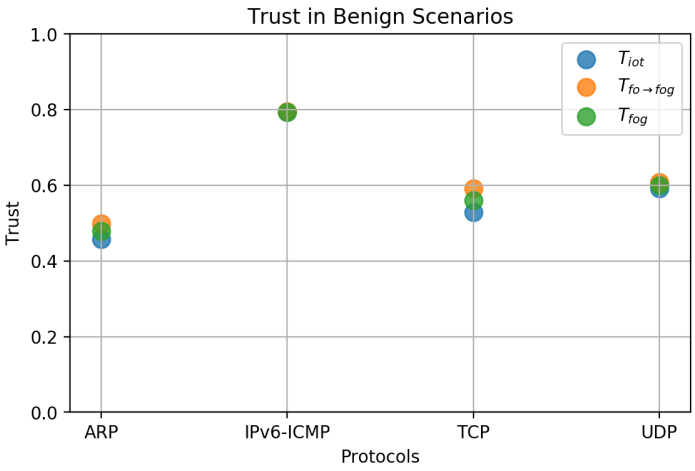
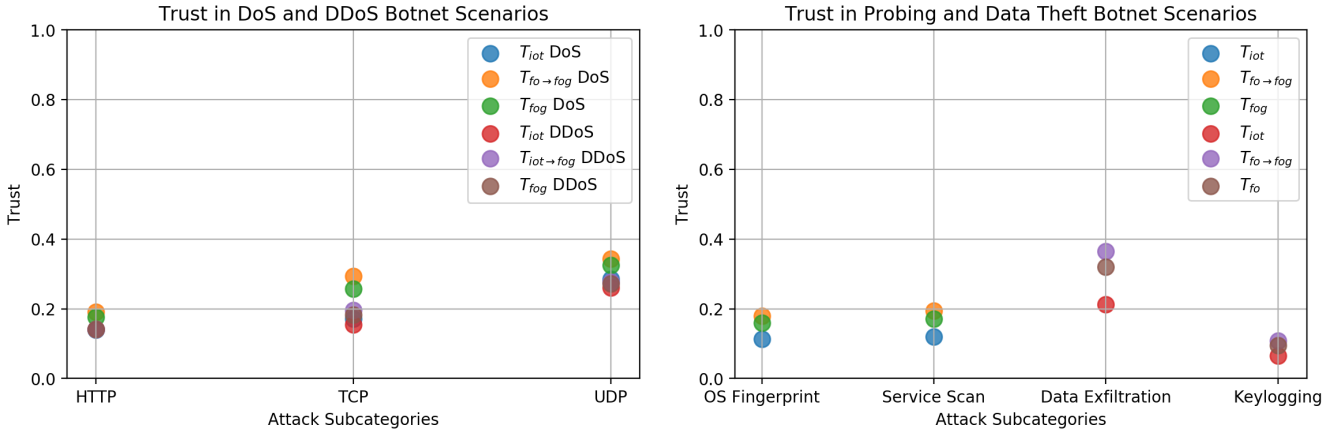


Fig. 6: Fog Nodes Trust in Benign Scenario

4.2.3 Trust Label Generation

After calculating the parameters, the next task is to generate trust labels. The average is calculated for every subgroup of benign and botnet scenarios listed in Table 4. Subsequently,



(a) (b)

Fig. 7: Fog Nodes Trust in Botnet Scenarios

every sample in the dataset is compared with the average of the benign scenarios and their particular botnet scenarios to find the normality and based on which the trust label (i.e., instant trust degree $\mathbf{T}_i$) is predicted by employing random forest regression. Precisely, objective trust $\mathbf{T}_{fo\to fog}$ gets a high value if the throughput and transaction are high, and bandwidth and energy consumption is low. The number of inbound connections per source and destination addresses increases dramatically from 10-20 to 100-200 in the case of DoS and DDOS attacks and helps in the assignment of trust labels. Similarly, the IoT Device trust $\mathbf{T}_{fog\to iot}$ is assigned a higher value if the parameters reported by IoT devices are in a given range quantified by averaging the parameter values.
Train-Test Split: Next, the dataset is partitioned, and 80% of the samples are used for training whereas 20% are used for testing. After intensive training of the regression model, it is tested with the remaining 20% of the data samples. The prediction accuracy of the random forest model is turned out to be 98% resulting in 2% mean square error (MSE) in predicted trust label and actual trust label.

It is underlined that the selection of a suitable regression algorithm was done by evaluating several other models, namely, multiple linear regression (MLR) and support vector machine regression (SVM). All algorithms were evaluated on MSE and the one with least MSE is selected. The MSE was 10% and 19% for MLR and SVM respectively. As random forest regression gave the highest prediction accuracy and the least MSE (2%), so it was selected.

4.2.4 Fog-IoT Entities Trust Results

The trust of fog nodes is computed based on the equations 1, 3 and 4 presented in section 3.4.5. For fog nodes, there are three results, namely, IoT device trust $\mathbf{T}_{iot\to fog}$, objective trust $\mathbf{T}_{fo\to fog}$, and fog node trust $\mathbf{T}_{fog}$. As the equations are based on a notion of time, so it is assumed that each of the botnet scenarios is taking place in a specific time period. Precisely, when analyzing the results, the notion of time should be considered. The notations $\mathbf{T}^i_{iot\to fog}$, $\mathbf{T}^i_{fo\to fog}$, $\mathbf{T}^i_{fog}$ and t_i denote IoT device, objective and fog node trust calculated at ith time instance t respectively. All trust results

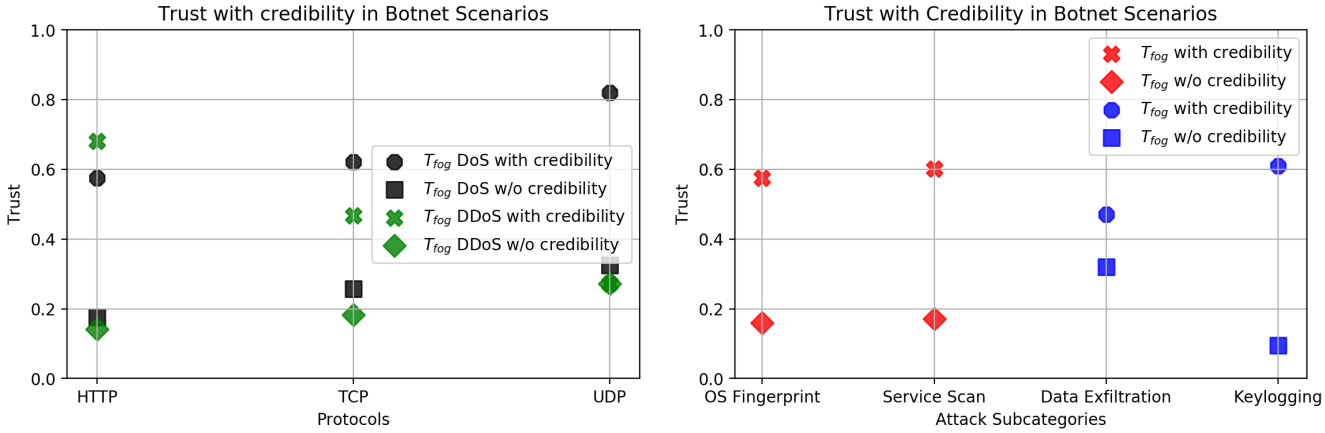
TABLE 5: Fog Node Trust

Scenarios	$\mathbf{T}_{iot\to fog}$	$\mathbf{T}_{fo\to fog}$	$\mathbf{T}_{fog}$	ACR
Benign ($\alpha = 0.5$)				
ARP	0.45	0.49	0.47	Write
IPv6-ICMP	0.52	0.57	0.56	Delete
UDP	0.59	0.62	0.59	Delete
TCP	0.78	0.79	0.77	Modify
Botnet ($\alpha = 0.3$)				
DoS (HTTP)	0.14	0.19	0.17	None
DoS (TCP)	0.17	0.28	0.25	None
DoS (UDP)	0.28	0.34	0.32	Read
DDoS (HTTP)	0.14	0.14	0.14	None
DDoS (TCP)	0.15	0.19	0.18	None
DDoS (UDP)	0.26	0.27	0.27	None
OS Fingerprint	0.11	0.17	0.15	None
Service scan	0.13	0.19	0.17	None
Data Exfiltration	0.21	0.36	0.32	Read
Keylogging	0.06	0.10	0.09	None

and the access control rights (ACR) in benign and botnet scenarios are listed in Table 5.

1. IoT Device Trust $\mathbf{T}_{iot\to fog}$ - Fig. 6 illustrates $\mathbf{T}_{iot\to fog}$ calculated using Equation (3) presented in section 3.4.5. It is the average of $\mathbf{T}_t(d_i)$ calculated using throughput, bandwidth, energy consumption, and transaction duration and predicted using the random forest regression model. As the dataset does not contain a feature mentioning the number of IoT devices provisioning services from each source IP address (fog nodes in this case), so, it is calculated based on the number of transactions for each source IP address. In a benign scenario, the trust of an IoT device using the ARP protocol is 0.45, IPv6-ICMP is 0.52, UDP is 0.59 and TCP is 0.78. Comparing the benign results in Fig. 6 with the botnet in Fig. 7 (a) and (b), it can be observed that $\mathbf{T}_{iot\to fog}$ decreases from 0.78 to 0.06. In botnet scenarios, $\mathbf{T}_{iot\to fog}$ lies between 0.06 and 0.28 indicating a distrust.

2. Objective Trust $\mathbf{T}_{fo\to fog}$ - Fig. 6 lists $\mathbf{T}_{fo\to fog}$ values calculated using Equation (1) presented in section 3.4.5. $\mathbf{T}_{fo\to fog}$ values in benign scenarios for different protocols are 0.49, 0.57, 0.62 and 0.79 respectively. Similar to $\mathbf{T}_{iot\to fog}$, $\mathbf{T}_{fog\to fog}$ also decreases sharply in botnet scenarios. For instance, in DoS (HTTP), it is 0.19, and overall it is between 0.10 to 0.36, which again indicates a low trust and possibly



(a) (b)

Fig. 8: Fog Nodes Trust in Botnet Scenarios

a compromised node.

3. Fog Node Trust $\mathbf{T}_{fog}$ - Having computed $\mathbf{T}_{iot \to fog}$ and $\mathbf{T}_{fog \to fog}$, the FO aggregates them to compute $\mathbf{T}_{fog}$ using Equation (4) presented in section 3.4.5. The weights assigned to α in Equation (4) are carefully chosen to avoid any artificial tuning. As the data is already labelled, so it was easier for us to do so. In benign scenarios, both $\mathbf{T}_{fo \to fog}$ and $\mathbf{T}_{iot \to fog}$ are assigned equal weights of 0.5. However, in botnet scenarios, $\alpha = 0.3$ is selected for $\mathbf{T}_{iot \to fog}$, meaning that $\mathbf{T}_{fo \to fog}$ will get more weightage when IoT devices are malicious and reporting false parameters. Again as the data is labelled so, a constant value is assigned to all botnet scenarios.

Fig. 6 illustrates $\mathbf{T}_{fog}$ trust, it is 0.47, 0.56, 0.59 and 0.77 in different protocols. As $\mathbf{T}_{fog}$ is greater than the threshold 0.5, the fog nodes are considered trustworthy. 0.5 threshold is selected to guarantee that the devices are trustworthy with at least 50% probability. Similarly, $\mathbf{T}_{fog}$ in DoS in HTTP, TCP and UDP subcategories is 0.17, 0.25 and 0.32 respectively. For DDoS, it is 0.14, 0.18 and 0.27 in different subcategories. For probing attacks, it is 0.15 for OS fingerprint and 0.17 for service scan. Lastly, in case of data theft attacks, data exfiltration and keylogging, $\mathbf{T}_{fog}$ is 0.32 and 0.09.

4. Trust and Access Control Rights - Table 5 lists the access control rights (see Table 2) granted based on $\mathbf{T}_{fog}$. For brevity of expression, the access rights are granted based on $\mathbf{T}_{fog}$ alone. It can be observed that in benign scenarios, ACRs are *write*, *delete* and *modify*, however in case of botnet it is either *none* or *read-only*. In normal circumstances, the trust would not change very frequently and, therefore, access rights. However, monitoring the rate of change of trust can help detect and prevent malicious activities.

4.2.5 Credibility Evaluation

This experiment is designed to evaluate the credibility model's effectiveness in maintaining the accuracy of trust computation model even in the presence of malicious/compromised fog nodes and IoT devices. Trust results from previous section underline that the trust decreases in all botnet attacking scenarios. The attacks are labelled in the dataset and therefore the trust computation are comprehensive and in agreement with the parameters. However, as the malicious adversaries attack the fog nodes, they cannot be labelled as untrusted. Moreover, some malicious nodes can purposefully send false parameters to decrease the trust of fog nodes in open and distributed systems. Trust values help detect the network intrusions, but at the same time, FO needs to mitigate their impact by evaluating their credibility and comparing it with its calculations.

TABLE 6: Fog Node Trust Credibility Evaluation

Scenarios	Without Credibility		With Credibility	
	$\mathbf{T}_{fog}$	ACR	$\mathbf{T}_{fog}$	ACR
DoS (HTTP)	0.17	None	0.57	Delete
DoS (TCP)	0.25	None	0.62	Read & Execute
DoS (UDP)	0.32	Read	0.82	Special
DDoS (HTTP)	0.14	None	0.68	Read & Execute
DDoS (TCP)	0.18	None	0.46	Write
DDoS (UDP)	0.27	None	0.27	None
OS Fingerprint	0.15	None	0.55	Delete
Service scan	0.17	None	0.61	Read & Execute
Data Exfiltration	0.32	Read	0.47	Write
Keylogging	0.09	None	0.60	Delete

Next, the credibility of $\mathbf{T}_{fog}$ in DoS, DDoS, probing and data theft attacks is evaluated. Equation 5 is based on the notion of time, so it is assumed that all benign and botnet scenarios have occurred in specific time periods. However, for finding an appropriate value of σ, the standard deviation in a benign scenario by considering all protocols is calculated using Eq. (6). Similarly, σ in four botnet scenarios is also computed. Subsequently, σ in botnet scenarios is added up and subtracted from the benign scenario's value. Following the above computational procedure, $\sigma = 0.07$ is found and subsequently used in the credibility evaluation model.

Fig. 8 and Table 6 list $\mathbf{T}_{fog}$ calculated with and without credibility evaluation and the corresponding ACRs. As can be observed from Fig. 8(a), $\mathbf{T}_{fog}$ in DoS without credibility in HTTP is 0.17, TCP is 0.25, and UDP is 0.32. However, after credibility evaluation and comparing the results with the benign values in previous time t_0, $\mathbf{T}_{fog}$ is adjusted to 0.57, 0.62, and 0.82. The ACR is also adjusted from *none*, *read* to *delete* and *special permissions*. Similarly, in DDoS

categories, $\mathbf{T}_{fog}$ without credibility evaluation is 0.14, 0.18, and 0.27 with no access right. With credibility evaluation, $\mathbf{T}_{fog}$ in DDoS(HTTP) increased to 0.68, DDOS(TCP) to 0.46, and DDOS(UDP) to 0.27. ACRs have also change to *read & execute*, *write* and *none* respectively. $\mathbf{T}_{fog}$ results for probing (OS fingerprint, service scan) and data theft (data exflitration, keylogging) with or without credibility evaluation are shown in fig. 8(b). It can be observed that trust has increased due to a big difference between the value in a benign scenario in t_0 time instance and the σ.

Comparing the results in Table 6, it can be analysed that in all botnet scenarios, $\mathbf{T}_{fog}$ without credibility calculations remained between 0.09 and 0.32. It was lowest in the case of the keylogging attack in the data theft subcategory and highest in the data exfiltration, again in the same category. In both cases, it is increased to 0.6 and 0.47 respectively and the ACRs changed from *read* and *none* to *delete* and *write* respectively. The above results underpin the significance of the credibility model in calculating accurate and precise trust computations.

4.2.6 Comparative Analysis

The proposed scheme is compared with [32] in which a trust model is proposed for cloud based IoT systems. A major limitation of this work is not putting a check on false/fabricated data and let it be incorporated in trust computation. The use of false parameters in the trust model leads to inaccurate $\mathbf{T}_{iot \rightarrow fog}$ which does not fall between -1 and 1, as reported in Eq. (1) and (2) in section 6 of [32]. In contrast to this, our proposed TMC follows a two-fold approach to guarantee an accurate trust computation. First, it detects data anomalies and discards any data which is out of the range of trust parameters. Second, the credibility model countermeasures any discrepancies introduced in the calculated trust values. The comparison is based on two cases, 1) Case-1 wherein trust parameters greater than V_{max} (upper limit of range), and 2) Case-2 wherein trust parameter less than V_{min} (lower limit of range) are considered.

Fig 9(a) and (b) show $\mathbf{T}_{iot \rightarrow fog}$ calculated by both trust models. In Case-1, the parameter values greater than V_{max} are considered. Fig. 9(a) shows $\mathbf{T}_{iot \rightarrow fog}$ results from both models. It can be observed that $\mathbf{T}_{iot \rightarrow fog}$ computed by our model lies between 0.1 to 0.5, whereas it is between 0.48 to 0.99 from [32]. $\mathbf{T}_{iot \rightarrow fog}$ is high despite wrong values resulting into inaccurate trust. In Case-2, the parameter values less than V_{min} are considered. $\mathbf{T}_{iot \rightarrow fog}$ calculated in Case-2 are shown in fig. 9(b). Like Case-1, the trust model of [32] again fails to evaluate the credibility of parameters and subsequently resulting into false trust estimation equal to 1 for all samples.

4.2.7 TMC Overhead

Fig. 10 reports the overheads of the different trust results namely, objective trust $\mathbf{T}_{fo \rightarrow fog}$, IoT Device trust $\mathbf{T}_{iot \rightarrow fog}$, trust credibility evaluation and fog node trust $\mathbf{T}_{fog}$. It is noted that the random forest regression executed as part of the objective trust $\mathbf{T}_{fo \rightarrow fog}$ computation took only 2.23 seconds. Fog node $\mathbf{T}_{fog}$ trust calculation took 0.72 seconds. However, the IoT device trust computation $\mathbf{T}_{iot \rightarrow fog}$ took 3.98 seconds. Time taken by $\mathbf{T}_{iot \rightarrow fog}$ is the sum of time required for data normalization, random forest regression training, and testing. Additionally, the trust credibility evaluation took just 0.27 seconds. Lastly, the entire trust computation is done in 7.2 seconds. The results demonstrate that the proposed TMC is lightweight and incurs small computation overhead even on a Raspberry Pi 3.

5 Conclusion

In this paper, a secure integrated framework for Fog-IoT systems is proposed. The proposed framework is designed after a thorough investigation of these systems. Various dimensions (i.e., security, trustworthiness, and service orchestration) of Fog-IoT systems are studied to find the vulnerabilities and threats faced by such complex and inherently heterogeneous systems. After identifying the security and trust challenges, efforts were made to find a solution. However, soon it had been clear that Fog-IoT systems require an integrated approach that addresses these issues simultaneously; as the limitations and/or absence of one solution can be exploited by malicious attackers to disrupt these systems and impact their availability.

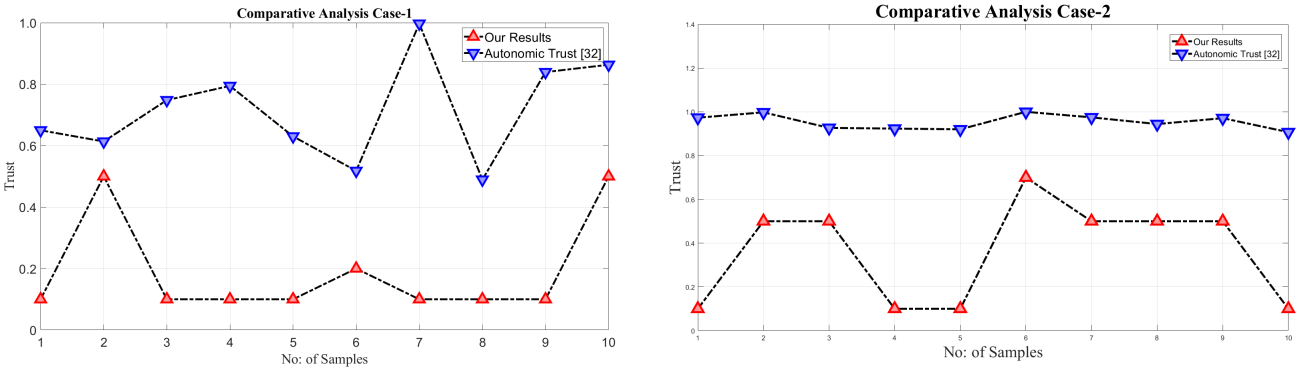
The SC component ensures security by achieving data confidentiality, integrity, authentication, and access control through a lightweight ABE scheme based on elliptic curves. The identity management and access control management subcomponents guarantee that fog nodes and IoT devices are authenticated and authorized. The TMC guarantees the dependability of Fog-IoT entities by computing their trust, based on QoS parameters and other performance indicators. The BotIoT dataset used in TMC evaluation has labelled samples for eight attacking scenarios, including DoS, DDoS, probing, and data theft, so trust is calculated for every scenario and compared with the benign results. Additionally, the credibility of trust is evaluated to highlight the detection and prevention of network intrusions. Trust is further used in defining access control rights for each fog node and IoT device. Moreover, the performance of the proposed framework is experimentally evaluated by implementing SC and TMC on a Raspberry Pi 3 model B. The results demonstrate that a resource constrained device i.e., Raspberry Pi, can execute the encryption scheme and regression models without incurring significant overheads.

From experimental evaluations, it is concluded that lightweight solutions like those proposed in this paper can solve the numerous security and trust problems faced by Fog-IoT systems. Overall, it is believed that this research is timely, and the proposed secure integrated framework is a step in the right direction. As future work, the researchers aim to include other attacking scenarios namely, collusion, Sybil, bad-mouthing, and ballot-stuffing, in trust and credibility evaluation. Moreover, we aim to extend the proposed secure integrated framework to address the privacy challenges faced by the Fog-IoT systems.

References

[1] O. C. A. W. Group, "Openfog reference architecture for fog computing," OpenFog Consortium, Tech. Rep., Feb. 2017. [Online]. Available: www.OpenFogConsortium.org

[2] C. Miranda, G. Kaddoum, E. Bou-Harb, S. Garg, and K. Kaur, "A collaborative security framework for software-defined wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2602–2615, 2020.



(a) (b)

Fig. 9: Comparative Analysis

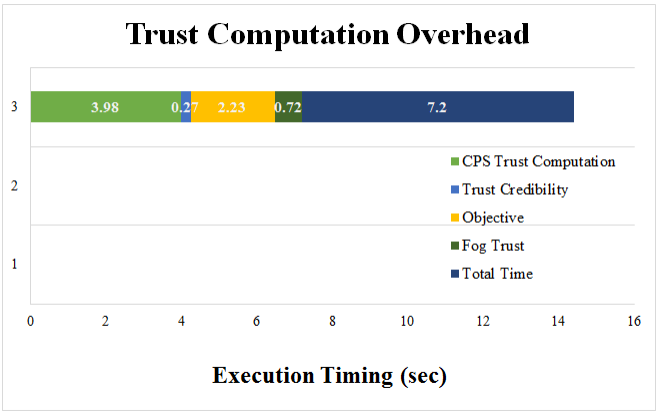


Fig. 10: Trust Computation Overhead

[3] H. Wu and W. Wang, "A game theory based collaborative security detection method for internet of things systems," *IEEE Transactions On Information Forensics And Security*, vol. 13, no. 6, 2018.
[4] N. Djedjig, D. Tandjaoui, F. Medjek, and I. Romdhani, "Trust-aware and cooperative routing protocol for iot security," *Journal of Information Security and Applications*, vol. 52, p. 102467, 2020.
[5] D. Soukup, O. Hujňák, S. Štefunko, R. Krejčí, and E. Grešák, "Security framework for iot and fog computing networks," in *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 2019, pp. 87–92.
[6] G. George and S. M. Thampi, "A graph-based security framework for securing industrial iot networks from vulnerability exploitations," *IEEE Access*, 2018.
[7] K. Muhammad, R. Hamza, J. Ahmad, J. Lloret, H. Wang, and S. Wook Baik, "Secure surveillance framework for iot systems using probabilistic image encryption," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 8, 2018.
[8] F. Al-Turjman, Y. K. Ever, E. Enver, H. X. Nguyen, and D. B. David, "Seamless key agreement framework for mobile-sink in iot based cloud-centric secured public safety sensor networks," *IEEE Access*, 2017.
[9] M. Alhanahnah, P. Bertok, Z. Tari, and S. Alouneh, "Context-aware multifaceted trust framework for evaluating trustworthiness of cloud providers," *Future Generation Computer Systems*, vol. 79, no. 2, pp. 488–499, Feb 2018.
[10] X. Jia, D. He, and N. Kumar, "Authenticated key agreement scheme for fog-driven iot healthcare system," *Wireless Networks*, vol. 25, no. 8, pp. 4737–4750, 2019.
[11] L.-Y. Yeh, P.-Y. Chiang, Y.-L. Tsai, and J.-L. Huang, "Cloud-based fine-grained health information access control framework for lightweight iot devices with dynamic auditing and attribute revocation," *IEEE Transactions on Cloud Computing*, vol. 6, no. 2, 2018.
[12] T. Mick, R. Tourani, and S. Misra, "Laser: Lightweight authentication and secured routing for ndn iot in smart cities," *IEEE Internet of Things Journal*, vol. 5, no. 2, 2018.
[13] K. Mahmood, S. A. Chaudhry, H. Naqvi, S. Kumari, X. Li, and A. K. Sangaiah, "An elliptic curve cryptography based lightweight authentication scheme for smart grid communication," *Future Generation Computer Systems*, vol. 81, no. 1, 2018.
[14] Z. Wang, "Leakage resilient id-based proxy re-encryption scheme for access control in fog computing," *Future Generation Computer Systems*, vol. 87, pp. 679 – 685, 2018.
[15] L. Lu and Y. Yuan, "A novel topsis evaluation scheme for cloud service trustworthiness combining objective and subjective aspects," *Journal of Systems and Software*, vol. 143, pp. 71–86, September 2018.
[16] T. Wang, G. Zhang, M. Z. A. Bhuiyan, A. Liu, W. Jia, and M. Xie, "A novel trust mechanism based on fog computing in sensor–cloud system," *Future Generation Computer Systems*, pp. 15 619–15 629, June 2018.
[17] I. García-Magariño, S. Sendra, R. Lacuesta, and J. Lloret, "Security in vehicles with iot by prioritization rules, vehicle certificates, and trust management," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 5927–5934, 2019.
[18] N. Haddadou, A. Rachedi, and Y. Ghamri-Doudane, "A job market signaling scheme for incentive and trust management in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 8, pp. 3657–3674, 2015.
[19] E. Alemneh, S.-M. Senouci, P. Brunet, and T. Tegegne, "A two-way trust management system for fog computing," *Future Generation Computer Systems*, vol. 106, pp. 206 – 220, 2020.
[20] C. Esposito, O. Tamburis, X. Su, and C. Choi, "Robust decentralised trust management for the internet of things by using game theory," *Information Processing and Management*, vol. 57, no. 6, p. 102308, 2020.
[21] J. Liang, M. Zhang, and V. C. M. Leung, "A reliable trust computing mechanism based on multisource feedback and fog computing in social sensor cloud," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 5481–5490, 2020.
[22] V. Odelu and A. K. Das, "Design of a new cp-abe with constant-size secret keys for lightweight devices using elliptic curve cryptography," *Security and Communication Networks*, vol. 9, no. 17, pp. 4048–4059, Nov 2016.
[23] A. K. Junejo, N. Komninos, M. Sathiyanarayanan, and B. S. Chowdhry, "Trustee: A trust management system for fog-enabled cyber physical systems," *IEEE Transactions on Emerging Topics in Computing*, pp. 1–1, 2019.
[24] A. Criminisi, J. Shotton, and E. Konukoglu, "Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning," vol. 7, no. 2-3, p. 81–227, 2011.
[25] J. A. Akinyele, C. Garman, I. Miers, M. W. Pagano, M. Rushanan, M. Green, and A. D. Rubin, "Charm: a framework for rapidly prototyping cryptosystems," *Journal of Cryptographic Engineering*, vol. 3, no. 2, pp. 111–128, 2013.
[26] GNU, "Gnu octave scientific programming language, https://www.gnu.org/software/octave/," 2019.
[27] Z. Zhou, D. Huang, and Z. Wang, "Efficient privacy-preserving

ciphertext-policy attribute based-encryption and broadcast encryption," *IEEE Transactions on Computers*, 2015.
[28] F. Guo, Y. Mu, W. Susilo, D. S. Wong, and V. Varadharajan, "Cp-abe with constant-size keys for lightweight devices," *IEEE Transactions on Information Forensics and Security*, 2014.
[29] Y. Shota, A. Nuttapong, H. Goichiro, and K. Noboru, "A framework and compact constructions for non-monotonic attribute-based encryption," in *Public-Key Cryptography - PKC 2014*, vol. 1, 2014.
[30] C. Chen, Z. Zhang, and D. Feng, *Efficient Ciphertext Policy Attribute-Based Encryption with Constant-Size Ciphertext and Constant Computation-Cost*. Springer Berlin Heidelberg, 2011, pp. 84–101.
[31] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset," *Future Generation Computer Systems*, vol. 100, pp. 779 – 796, 2019.
[32] S. Namal, G. Hasindu, G. Myoung Lee, and T.-W. Um, "Autonomic trust management in cloud-based and highly dynamic iot applications," in *ITU Kaleidoscope: Trust in the Information Society (K-2015)*. Barcelona, Spain: IEEE, dec 2015.

A. K. Junejo received her Ph.D. degree in Computer Science from City, University of London, UK in 2019. She is currently a Research Associate in Imperial College London, UK. Before her Ph.D. studies, she had been working as a visiting lecturer and software engineer in Mehran University of Engineering and Technology, Pakistan. She is currently researching the security of sensor based systems as part of the S4 project funded by EPSRC. Her research interest include cyber physical systems, wireless sensor networks, security and privacy of cloud computing, fog computing, applied cryptography, and trust management.

N. Komninos received his Ph.D. in 2003 from Lancaster University (UK) in Information Security. He is currently a Senior Lecturer (Associate Professor) in Cyber Security in the Department of Computer Science at City, University of London. Prior to his current post, he has held teaching and research positions at the University of Cyprus, Carnegie Mellon University in Athens (Athens Information Technology), University of Piraeus, University of Aegean, and University of Lancaster. Since 2000, he has participated, as a researcher or principal investigator, in a large number of European and National R&D projects in the area of information security, systems and network security. He has authored and co-authored more than ninety journal publications, book chapters and conference proceedings publications in his areas of interest.

Julie A. McCann (M'16) is a Professor of computer systems with Imperial College London, London, U.K. Her research centers on highly decentralized and self-organizing scalable algorithms for spatial computing systems, e.g., wireless sensing networks. She leads both the Adaptive Embedded Systems Engineering Research Group and the Intel Collaborative Research Institute for Sustainable Cities, and is currently performing research with NEC and others on substantive smart city projects. She has received significant funding though bodies such as the U.K.'s EPSRC, TSB, and NERC, as well as various international funds, and is an elected peer for the EPSRC. Ms. McCann has actively served on, and chaired, many conference committees and is currently Associative Editor for the ACM Transactions on Autonomous and Adaptive Systems. She is a Fellow of the BCS.