Seyed Amin H. Seno
Computer Engineering Department, Ferdowsi University of Mashhad, Iran
Hosseini@um.ac.ir

Sahar A. Alshammari
Computer Engineering Department, Ferdowsi University of Mashhad, Iran
Sahar.alshammari@mail.um.ac.ir

Received on: 30/04/2018
Accepted on: 23/05/2019
Published online on: 25/07/2019

A Cooperation of Fog Computing and Smart Gateways in a Secure and Efficient Architecture for IoT-Based Smart Homes

Abstract- *Nowadays Internet of Things (IoT) is growing to be a serious factor in numerous areas of our daily life style. Internet of Things brings different opportunities of intelligence to important aspects such as health, payments, energy management, industrial sectors, transportation and also many other specialties. It is important to notice that the interaction between these two part the embedded equipment and Cloud based web services is such a common or prevalent scenario of Internet of Things deployment. When it comes to the security point of view, jointly users (consumer) and smart devices need to reassure and establish a secure and confident communication channel and should have a perfect form of digital identity. In many situations, IoT devices needs an already or earlier established infrastructure for their usage and that cannot be managed by the device owner, such as the case in smart homes. Furthermore, the scenario presupposes a security stack that it is appropriate for heterogeneous devices which can be integrated in Internet of Things frameworks or in already presented operating systems. We proposed a Foggy Smart Home Architecture (FSHA). We identify end users by writing an authentication and authorization protocol, and we will reduce the time required for this security operation, so that the proposed method can prevent Non-manipulation, online/offline password guessing attack and user impersonation attack and man-in-the-middle attack. Our method improves performance of smart home and using fog layer can minimize traffic between cloud and gateways.*

Keywords: *fog computing, security, Internet of Things, Smart home, IoT, Authentication*

How to cite this article: S.A Alshammari and S.A. Hosseini Seno, "A Cooperation of Fog Computing and Smart Gateways in a Secure and Efficient Architecture for IoT-Based Smart Homes," *Engineering and Technology Journal*, Vol. 37, Part A, No. 7, pp. 290-301, 2019.

1. Introduction

The technique of CISCO lately relied on the accurate vision of fog computing in order to qualify applications on billions of active devices, previously connected in IoT, to have the ability of running directly at network edge. Consumer can do the processes such as developing, running, managing and deploying of software applications on Cisco IOx framework of networked resources, that holding hardened routers, switches and IP video cameras. However, Cisco IOx works with open source Linux and Cisco IOS network operating system cooperatively in a single networked equipment. The open application environment prompts more developers to connectivity interfaces at edge of network and bring their own applications. Nevertheless, of Cisco's utilizing, firstly let us take a look at Fog computing conception and make a clear comparison of what are distinction between these two illustrations Cloud and Fog.

To start with, fog computing, services and facilities can be appended at the end equipment such as set-top-boxes or access point. The fundament of this modern distributed computing permits applications to run as close as possible to sensed actionable, massive data and processes and thing, coming out of people. Comparable with Fog computing conception, in fact a Cloud computing close to the 'ground' creates automated response that drives the value. One and the other supply data, application services, storage and computation to the Consumer. But, Fog can be discrete from Cloud by its closeness to Consumer, dense geographical distribution and its support for mobility [1].

As a result of momentary development and expansion of IoT, there are various kinds of Internet of Things facilities and different kind of applications that participate to our daily life. Therefor they cover from conventional resources to general household objects that assist to make human being's life preferable. It is of considerable possibility.

Meantime, there are a number of issues or challenges should be considered in path of IoT. In

http://dx.doi.org/10.30684/etj.37.7A.10
2412-0758/University of Technology-Iraq, Baghdad, Iraq

conditions of scalability, an IoT application that needs sizeable and different numbers of resources are often hard to execute because of number of important features such as time limitation, memory, processing and energy qualification. Research has found, the result of calculation that can be obtained from daily temperature alterations around all of the country may need millions of devices and result in unmanageable amount of data. The useable hardware in Internet of Things oftentimes have various operating properties, like error distributions and sampling rates, meantime actuators and sensors elements of IoT are repeatedly very complicated. All of the mentioned features participate in generation and formation of heterogeneous network of Internet of Things in which data of Internet of Things will be intense or deep heterogeneous.

Furthermore, it would be highly cost to transmit large amount of raw data in complicated and heterogeneous network, for that IoT require data fusion and data compression to minimize the data volume. Thus, the order and the standardization of data processing awareness for coming IoT is extremely necessary. In order to prevent malicious software, hackers and virus in communication process from hinder information and data integrity.

Due to the processes of development and expansion of IoT technology, the unsecured information will immediately threat the whole Internet of Things system. These days, IoT is exceedingly applied to social daily life tools and applications for instance intelligent transportation, smart grid, smart security, and smart home. Access cards, bus cards and some other small applications also belong to IoT. Tools and Applications of IoT can provide convenient to the consumers, but should ensure security and personal privacy. Because if it not, this private information may be leaked at any time. So, security of Internet of Things cannot be neglected. The security of entire information of IoT will be immediately affected at the moment when the signal of IoT is stolen or interrupted. As the result of extremely development of Internet of Things, it will supply more extensive wealthy of information, risk of information exposure will be increased. If IoT cannot have a better solution for security problems, it will hugely restrict its development. Consequently, all the mentioned issues of IoT, security vulnerabilities are exclusively significant [2].

When the emerging IoT is considered to be the next generation of the Internet and that is mean hackers will find it a likeable aim for them, in which billions of things are interconnected. Each physical object in the IoT has the ability of interaction without need to human interventions. In contemporary years, different kinds of tools and applications that have various bases and infrastructures have been spreader, for instance logistics, manufacturing, healthcare, industrial surveillance, etc.

There is a fine number of cute-edging techniques (for example wireless communication, intelligent sensors, data analysis technologies, cloud computing, networks, etc.) have been progressed and developed to recognize possibility of Internet of Things with various intelligent systems. Yet, technologies for Internet of Things are still in their early and infant step and a considerable number of technical awkwardness supported with Internet of Things requirement to be control. Security is one of most important holdbacks in Internet of Things, which covers communication network security, application security, general system security, and sensing infrastructure security [3].

Because fog computing originated from cloud computing, and it is suggested in context of Internet of Things, as a result of that fog computing inherits the privacy and security problems of cloud. In spite of that several issues and problems can be managed by using known schemes, but of course that other problems facing new challenges, on account of special features of fog computing, like heterogeneity in fog network and fog node, massive scale geo-distributed nodes, requirement of mobility support, low latency and location-awareness [4].

The rest of the paper is arranged as follows: In Section 2, related work and motivation of this paper are presented. Section 3 is described suggested approach. The result and evaluation of the suggested approach is presented in Section 4. Finally, Section 5 concludes the paper.

2. Related Works

In this section, we are going to have a look at the works done in this field.

Sanaz Rahimi Moosavi in [5], providing a secure and an effective authorization and authentication architecture for IoT-based healthcare. There is prime focus in this work for providing authentication and authorization for healthcare professional system in higher secure manner. In proposed scheme the process of authorization and authentication of end-user that he/she is in a remote distance is accomplished by distributed smart e-health gateways to relieve medical sensors from implementing these missions. The produced architecture depends on the certificate-based DTLS handshake protocol as it is major IP

security solution for internet of things. The suggested authorization architecture and authentication is examined by developing a prototype IoT-based healthcare system. The prototype is built of WiSMotes, a TI Smart RF06 board and a Panda board. The CC2538 module integrated into WisMotes act as medical sensor nodes and TI board acts as a smart gateway. The suggested architecture is more secure than a modern centralized delegation-based architecture since it employs a more secure key management method between smart gateway and sensor nodes. Moreover, impact of DoS attacks is minimized due to distributed nature of architecture.

Barreto et al. in [6] recognized an architecture where TPM (Trusted Platform Module) equipped Internet of Things devices are part of an authentication model. So, this system employs digital certificates and it is role based. The role based system is appropriate for an Internet of Things deployment scenario because it supplies segregation between different missions carried out by users (owner) and by administrators (device manufacturer). The suggested identity management framework includes two modules namely Identity Manager and Service Manager. Service Manager acts as an authorization module which defines accessibility of receiver to information supplied by particular service and accessibility of a service to sensor information. Identity Manager is like an authentication module which authenticates receivers, sensors and services.

Chun-Ta Li et al. in [7] their scheme has some security littleness as demonstrate, the IoT-based medical care system has offered with data encryption scheme and an improved secure authentication, therefore user anonymity is supplied and security threats of replay is prohibited and sensed/password data attacks is disclosure. Furthermore, they mitigate authentication process to minimize redundancy in protocol design, and suggested architecture is more efficient in performance compared with previous related schemes. At last, when random oracle model under ECDHP, the suggested scheme is provably secure.

Pardeep Kumar et al. in this paper [8] present for in connected smart home environments an anonymous secure framework (ASF), by utilizing solely lightweight operations. The framework that have been introduced in this paper offers key agreement and efficient authentication, and enables this features of unlinks ability and equipment anonymity. One-time session key progression orderly changes session key for smart devices and reduces risk of utilizing a compromised session key in the ASF. Lastly if we compare with existing schemes we will notice that computation complication of suggested framework is soft, and security has been safely enhanced.

Marica Amadeo in paper [9] with her produced suggestion (a novel CoT platform) that help to solve many challenges in smart home domain by using two groundbreaking concepts and Fog Computing: Information Centric Networking (ICN). Furthermore, the proposal, called ICN-iSapiens, is a three-layered scheme where an intermediate (Fog) layer, including of smart home servers (HSs), is presented between remote cloud and physical world, to upholding real-time services and hides heterogeneity of Internet of Things equipment. Communication at physical layer includes of name based ICN primitives, which smooth network configuration and enable effective and easy interactions between Internet of Things devices and HSs. As proof of concept, an experimental test bed is proposed and some application examples are defined to showcase advanced capabilities of ICN-iSapiens.

Joy Dutta in paper [10] propose a prototype of a smart building utilizing newly surfacing technologies like IoT, cloud and fog for smart city. The demanding for everything smart is enhancing once a day, but major tumbling block is that it is expensive. Therefore, their target is to enhance criterion of living in office and in home with latterly improved working facilities where the entire system will be automatic, trusted, efficient and will be controlled by the user via his/her smartphone or computer but the important thing to be noticed that is the cost will still within the budget of a common man. All these facilities are done by the incorporation of IoT, fog and cloud. The absorption is done by using open source hardwares and softwares to minimize the cost dramatically than the other existing solutions and perform it in an impressive and ingenious way without compromising Quality of Service of any of the functionalities offered by other existing solutions.

Jianhua Li et al. in paper [11] investigate Fog computing as platforms for a smart living illustrate, namely, EHOPES. They proposed the desired Fog elements such as FS, FEN and Foglet from Internet of Things user's perspective. Different parts of FS and FEN in terms of storage, processing and communication are taken into consideration for EHOPES. Two interest cases are suggested to exhibit the impact and the effectiveness of minimizing the latency for the same self-quantities of data on Fog in contrast to Cloud. Even though that this paper concentrates

on Fog platform for smart living, the framework is prepared to be mostly utilized to other IoT applications wherever Fog is employed. As Fog is simply in its infancy stage, bags of work and task are still needed to be done, e.g., workload mobility between Cloud and Fog, Fog routing and switching, Fog security, Fog deployment and QoS, interplay between smart object, Fog node and Cloud as well as Data storage (pull and push).

Yuvraj Sahni et al. in paper [12] suggest Edge Mesh as a new computing paradigm, which concentrate on vesting distributed intelligence in internet of things. Edge Mesh works like that it distributes the entire application into sub-tasks which are distributed among Edge tools. Edge tools simultaneously with routers form a mesh network which is accountable for many computation missions like processing, data storage, sharing, etc. Edge Mesh attempts to integrate and merge better characteristics from Fog computing, Cloud computing, and cooperative computing to supply multi-dimensional features. So, this paper suggests a software framework for Edge Mesh. Software framework is split into three levels depended on End devices, Cloud, and Edge Mesh. A mission management framework for managing and distributing has also been discussed in detail.

Wangbong Lee et al. in paper [13] propose a gateway based on fog computing architecture for WSANs and argue that the key needs for this type of architecture. This scheme typically contains master and slave nodes, and implement management for resources, flows, and virtual gateway functions. For prototyping OpenWrt platform and Traditional WiFi equipment are perfect start point. Micro server platform will be relied on Raspberry Pi. The 2-tiered distributed architecture includes of gateways in control masters and lower tier, comparatively more powerful gateway platform, in upper layer. They will display that this scheme offers manageability and scalability for many networked objects.

Rahmani, Amir M., et al [14] take advantage of the strategic position of such gateways at the edge of the network to provide numerous kind of services such as real-time local data processing, local storage, embedded data mining, etc. which considered as higher-level services, also presenting a Smart e-Health Gateway. Then they suggest exploiting the notion of Fog Computing in Healthcare IoT systems by pointing a Geo-distributed intermediary layer of smart between Cloud and sensor nodes. By taking responsibility for handling some issues and burdens of a remote healthcare and center sensor network, their Fog-assisted system architecture can cope with a lot of challenges and issues in ubiquitous healthcare systems for instance mobility, reliability, energy efficiency, and scalability problems. An effective performing of Smart e-Health Gateways can enable massive deployment of ubiquitous health monitoring systems particularly in clinical environments. In addition, they show a prototype of a Smart e-Health Gateway named UT-GATE where some of discussed higher-level characteristics have been performed. They also perform an IoT-based Early Warning Score (EWS) health monitoring to practically present the efficiency and relevance of their system on addressing a medical case study. Their testament of concept design made an IoT-based health monitoring system with improve the intelligence of the whole system, energy efficiency, mobility, interoperability, reliability, security and performance. The present study aim was to improve the performance of smart home system. Reducing the amounts of data which are sent to cloud server by hiring a fog device in the edge of network and doing some lighter processes by it. Writing an authenticating and authorizing protocol for identifying end-users and minimize the time that is needed for this security operation.

3. Proposed Method

In following section, we first describe proposed architecture, and then define proposed authentication protocol.

I. Foggy Smart Home Architecture (FSHA)

The main problems of IOT systems in smart homes are about security of data and real-time responses. In this paper, we discussed about the solutions which we believe can solve the mentioned problems. For solving these problems, we proposed bellow suggestions and design a Foggy Smart Home Architecture (FSHA) which is shown in Figure 1.

Firstly, we hired gateways which are embedded on each room to ensure the authentication and authorization to any End-Users who want to access to sensor's data or do something to actuators. This can be done by writing an authenticating and authorizing protocol to outsource some load of sensor nodes that give those sensors authority to communicate efficiently and securely beyond independent network domains. By supplying confirmed connection context to sensor nodes, give a chance to the devices to access their data without needs to authorize and authenticate a remote end-user. Consequently, any malicious and malignant activity can be blocked before entering to their area by putting this task on

gateways which have unlimited and powerful resources for computing and communication.

Gateways are connected to the internet to identify End-Users directly. This idea will reduce the amount of communication overhead which is maybe produced for the operation of authenticating and authorizing.

In order to prevent adversaries from performing Online/offline password guessing attack, User impersonation attack, Man-in-the-middle attack, security operations of each room are performed with gateways which turn the centralized manner of IOT systems to distributed one.

Fog device is also used for doing some computational operations which have to calculate immediately and don't need resources as powerful as cloud systems. Fog computing minimize the amount of data which must transmit to the cloud and obviously, alleviate network traffic which are surely created by such data transmissions. Compared with Cloud, Fog is greatly support smart living because of that it has four unique features, which are:

1. Low latency, i.e., Fog shows millisecond to sub second level latency, while it is in minutes- level in Cloud.
2. Proximity, i.e., Fog selects the decentralized model, which is closer to smart objects. Cloud selects the centralized model.
3. Real-time interaction, i.e., Fog computing offers quick even real-time interaction. Cloud is perfect at batch processing.
4. Multi-tenancy, i.e., both Fog and Cloud Support Multi-tenancy, but Fog performs better for applications that require low-latency.
5. A local database is also determined for fog device by which fog device can store data and based on data, it can make appropriate decisions for sensing and actuating operations.

Fog device also can early filter injected false data at the network edge and prevent this type of network attacks as well.

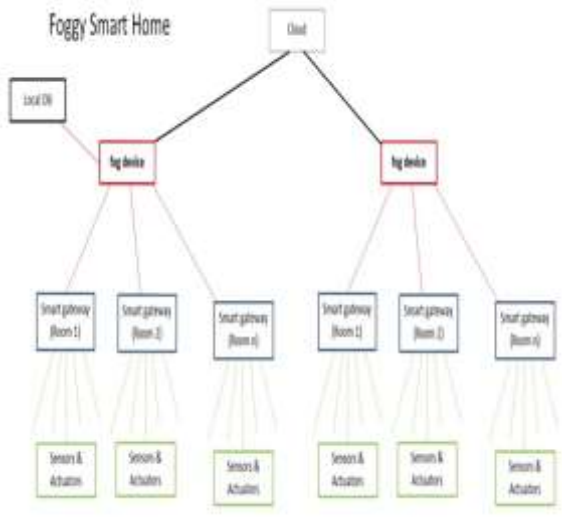

Figure 1: A Foggy Smart Home Architecture (FSHA)

II. Proposed authentication protocol

The security protocol used in the proposed method, which uses the Shamir threshold technique, consists of two phases:

A) Registration phase

In this phase, the user performs the following steps to register as a user authorized in Fog. It is assumed that in this phase, the communication channel between the user and Fog is a secure channel.

1. Generates a user's name and an arbitrary and non-repeat pass.
2. Generates a random number (rand).
3. Using the hash function, it generates a temporary password (t_{pass}) with the use of a password and a random number. t_{pass} = hash (rand || pass)
4. Sends the username and password (user, t_{pass}) to Fog.
5. Fog stores the temporary username and password in order to complete the registration in your local memory.

B) Authentication phase

In this phase, the authentication process is carried out.

1. The user sends his requests for access to sensors intended for smart gateways. These requests include the session number (s_1), the user name (user), the number of SGs involved in the session (countSG), and the required sensor number (p_1, ..., p_k), each pi is in the form of a point (x_i, y_i). For example, if you request access to two sensors in SG1 and three sensors in SG2, a message in the form (s_1, user, 2, p_1, p_2) to SG1 and a request form (s_1, user, 2, p_3, p_4, p_5) sends to SG2.

Each SG sends the request to Fog to authenticate the user. Fog waits to receive all requests with the same session number. When the number of non-repetitive requests received with the same session number (from different SGs) was equal to the amount of the countSG in the requests, it searches for the user name in its user list.

- If the user does not exist then it will deny the request.
- If the user was present but did not have access to all the requested sensors, notifies the user through a message to correct his request
- If the user has access to all requested sensors, then he accepts the request.

1. When the request is accepted, Fog generates a random number (Rand) and forms the fuser (x) function using the points (p_1,..., p_t) and hash' (t_{pass} || rand). In which (p_1,..., p_t) is set without repeating all the requested sensors of the user in this work session.
2. At this point, Fog send to user the Random Number (Rand) along with the session number. Fog also generates a random point from the fuser (x) graph called pr, and sends it to the SG using a shared

key along with the session number and the list of sensor numbers (p_1, ..., p_t).
3. The user forms the fuser function (x) using the sensor number set (p_1,..., p_t) plus pass and random number (Rand). Then calculates the Authentication Value (AV) from the equation below.
AV = hash (s_1, user, f_{user} (0))
And sends a message in the form (s_1, user, AV) for SGs.
4. Each SG after receiving this message from the user, using the list of sensor numbers and random pr, forms the function fuser (x) ' and obtains the value of fuser (0)'. Then calculates the value of AV' from the equation below.
AV'= hash (s_1, user, f_{user} (0)')
If the AV value is not equal to the AV' value, the user request is denied, but if it is equal, its request is accepted. After the authentication process is completed correctly, each SG uses a session key obtained from the equation to create a secure channel for exchanging messages with the user.
K_1 = hash (f_{user} (0))

III. Analysis of security

For this section, there is a preparation done by us that explains the analysis and detailed examination of our proposed system of course with concerning to security and performance. Highly important to notice that authentication that we proposed is theoretically secure. We suppose that a user intends to gain the accessibility to t Things for the purpose of simplification.
Fog sends a random number rand to the user and then he/she computes a secret point hash (rand ‖ pass). Thence, a user can reconstruct function $f_{user}(x)$. Furthermore, the SG receives a random point p_r from fog and reconstructs function named f_{user} (x). Accordingly, the validity can be proven. To satisfy security requirements these below Followings are security analysis.
Non-manipulation: The suggested authentication by us ensures non-manipulation if and if only a passive attacker of gateway. Only the user who knows a random number rand and valid password pass can generate a function f_{user} (x) and computes f_{user} (0). No one of other users can generate AV except corresponding SG which has a random point p_r sent from Fog with secure channel. If the user tried to obtain a One-Time Password generator with Fog, attackers cannot know rand as well, thence attackers cannot reconstruct function f_{user} (x) even so they know user's password and smart home system becomes more efficient and secure.
Online/offline password guessing attack: When the function f_{user} (x) and the authentication value AV are regenerated in every single session for that attackers will fail every time tried to acquire any information of users' password even so they have as many as authentication values AV. unless the (id-pw) list in Fog is disclosed, attackers cannot guess correct password of users.
User impersonation attack: assume attackers reconstruct the function $f_{user}(x)$ by chance. After that they can impersonate a valid user, but it is important to know that is only potential at that session and no more helpful or useful. And also, unless adversaries attain the password of user pass and combination of Things, they cannot reconstruct function f_{user} (x). Thus, the adversaries cannot impersonate valid users.
Man-in-the-middle attack: Attackers have the ability to acquire the information of user like, (session number s1; user name user; combination of Things (p_1,..., p_t); random number rand ; and authentication value AV). After that they (attackers) will try to gain the accessibility to Things using that information. But they do not have the ability to reconstruct fuser (x) and manipulate authentication value AV. For that our suggested authentication architecture is provably secure against Man-in-the-middle attack.

4. Results

I. Simulation environment

In the first step, in order to find a suitable environment that could both simulate an IOT-based environment and develop the security protocol required to secure communication with sensors, the search and evaluation of proposed environments was addressed. And from among these environments, a toolkit called IFogSim [15] was selected. This environment is an extension of the CloudSim simulation environment. The possibility of processing on the network side makes possible through a tool called Fog Device. By default, we considered a high-rise building that has a Fog which server is our authentication, and is connected to a cloud, and the number of nodes has three sensors, three actuators, three smart gateways and one user and gateway related to it. The number of nodes is equal to 11 and the number of servers is equal to 1.

II. Implementation

To implement the proposed authentication method, we used JAMA. Which is a basic linear algebra package for Java. The function we use the Jama class is polynomial Regression.
Polynomial regression is a form of regression analysis in which the relationship between the independent variable x and the dependent variable y is modelled as an nth degree polynomial in x.
JAMA supplies user-level classes that is useful for constructing and manipulating real, dense matrices. Also for supplying enough functionality for the

routine problems, packaged in a way that is natural, ideal and understandable to non-experts. It is intentional to perform as standard matrix class for Java, and will be suggested as such to Forum and then to Sun. A straightforward public-domain reference implementation has been developed by the Math Works and NIST as a straw man for such a class. We are releasing this version in order to gain public comment. There is no guarantee that future versions of JAMA will be compatible with this one [16].

III. Scenarios

In this section, we describe the scenarios that used.
The first two scenarios are to demonstrate the advantage of using Fog in Smart Home. The criteria such as end-to-end latency and Network usage are significant improvements. In this scenario 1, a processing element called Fog is used at the edge of the network, as shown in Figure 2. In scenario 2, Fog is not used and as shown in Figure 3.

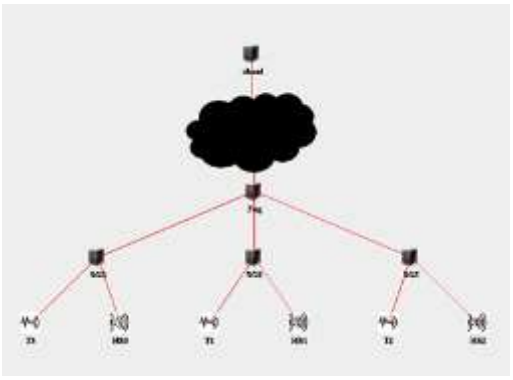

Figure 2: Topology with Fog

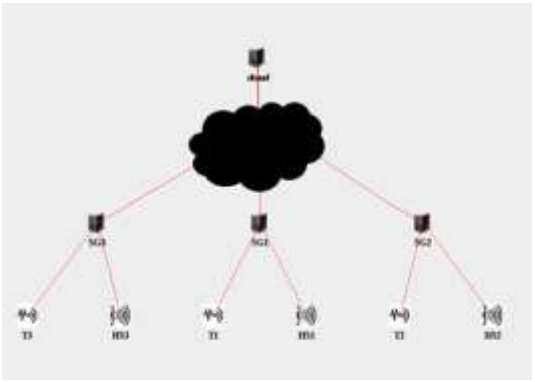

Figure 3: Topology without Fog

In these two scenarios, a processing element called Cloud is used. The specifications are shown in Table 1. It is placed at the level of 0 in the topology. Because this simulator uses a hierarchical structure. At Level 1, there is also a Fog that performs real-time processing and intermediate processing, and sends it to Cloud if it needs to be processed. At Level 2, three smart gateway elements named SG1, SG2 and SG3. These elements are attached to two other elements at the leaf level.

1. The sensors are called T1, T2, and T3 as temperature sensors. Its specifications are shown in Table 2.

Table 1: Element Specifications

Device	Mips	RAM(MB)	UpBW(KB)	DownBw(KB)	Rate/Mips	Level
Cloud	1000	8000	100	1000	0.25	0
Fog	0	1000	1000	10000	0.0	1
SG1, SG2, SG3	1000 500	500	1000	1000	0.0	2

2. Actuators that are heating systems and names are HS1, HS2, HS3. Its specifications are shown in Table 3. In this scenario, the sense-process-actuator process starts with sensors and goes to SGs and processes it into Fog and then is sent back to the actuators via the SGs to apply the final operation. For example, if the temperature is less or greater than that, the sensor will be notified to the Fog. And decisions are made on Fog to order the heat increase to actuators, which are the same as the heating system, and the temperature increases.
In the second scenario, Fog is not used and there are three levels of cloud, gateway and sensors, and the decision-making process is done in the Cloud.

Mips: Million Instructions per Second (Processing Power)
UpBW: upward bandwidth in the hierarchical structure
DownBw: downlink bandwidth
Rate / Mips: The amount of cost calculated for each million instructions used
Level: Node level in tree structure

Table 2: Sensor Specifications

Device	Sensor Type	Type	Distribution
Sensor T	TEMP	Sensor	1(Normal)

Distribution: represents the send rate of the tuple to the next node.

Table 3: Actuator Specifications

Device	Name	Type	ActuatorType
Actuator HS	HS1	actuator	HeatingSystem

The lines between entities are Edge, which has source and destination properties and delays. For example, the latency of elements inside the network is at a low of 3, 5, while the latency of the links through the Internet (such as Cloud-Fog or Cloud-SG) is 50. These delays are shown in Table 4.

Table 4: Delays of links

Source	Destination	Latency
Sensor	SG	3.0
Actuator	SG	5.0
SG	Fog	5.0
Fog or SG (Via Internet)	Cloud	50.0

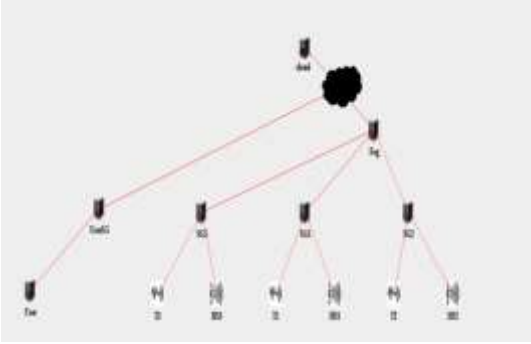

Figure 4: Topology of authentication in Fog

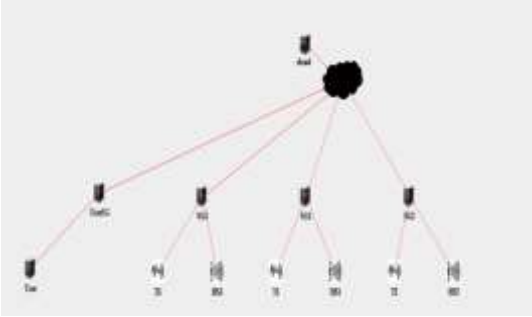

Figure 5: Topology of authentication in SGs

This scenario 3 relates to the process of user authentication in the security system. In the scenario, authentication operations will be performed by Fog and as shown in the Figure 4. In scenario 4, the user authentication process in the security system is performed in the FOG authentication scenario. And in this scenario, authentication operations will be performed by SGs and shown in the Figure 5.

In these scenarios, two new elements are added to the network as in Scenario 3.

- One User, the same user, attempts to connect to the network and use sensor data. This user can be a landlord, building manager, or even an installation manager. The processing power for this user is considered modest and less than SG, and for access, and for access, it must have done the authentication process.
- The next element is called userSG, which is the gateway which the user is connected to the Internet via it and Performs the authentication process. Now in the scenario of 3 users connected to Fog via userSG and the Internet. And authentication carried out for any number of sensors from any SG only and only once per session. In scenario 4 this process for the sensors each of SG needs to be done once.

To implement each of the scenarios in the iFogsim simulator, the scenario steps should be designed in the form of an application. Each APP includes modules and communication edges of these modules. Eachmodule and Edge have separate attributes such as Ram, CPU, etc., which will be presented in Tables 5 and 6.

Each app should be in the form of a Direct Arcylic Graph (DAG), as shown in Figure 6, in which circles, modules, and communication lines are Edges After the app design, the modules must be mapped to the devices, that is, every module will run on which node. For example, the decision maker module runs on the Fog. So, it should be mapped to the Fog node. Table 7 and 8 show module mapping to node in scenario 1 and 2 respectively.

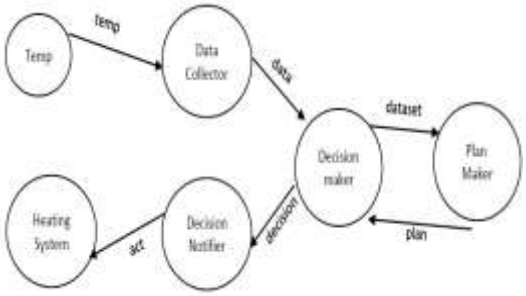

Figure 6: App for scenario 1

Table 5: Specifications of Module for Scenario 1 & 2

Modul Name	DataController	DecisionMaker	PlanMaker	DecisionNotifier
RAM(MB)	100	900	7000	100

Table 6: Specifications of Edges for Scenario 1 & 2

Edge Name	Temp	Data	Dataset	Plan	decision	Act

Mips	10	100	1000	100	10	5
TupleLength	500	500	1000	500	500	500

Table 7: Mapping the module to node in scenario 1

Device	SG	SG	Fog	Cloud
Module	DataController	DecisionNotifier	DecisionMaker	PlanMaker

Table 8: Mapping the module to node in scenario 2

Device	SG	SG	Cloud	Cloud
Module	DataController	DecisionNotifier	DecisionMaker	PlanMaker

Finally, a Loop for the app should be designed to measure different policies such as: Delay and consumption of network and ... The app for the scenario 2 is similar to Figure 6, the difference is how the modules are mapped to devices. The app for scenario 3 is as shown in Figure 7. Each module and Edge have distinct attributes such as Ram, CPU, etc., which will be presented in Tables 9 and 10. Table 11 and 12 show module mapping to node in scenario 3 and 4 respectively.

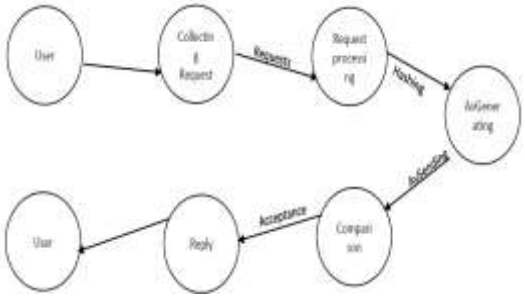

Figure 7: App for scenario 3

Table 9: Specifications of Modules for Scenario 3 & 4

Modul Name	CollectingRequests	RequestProcessing	AvGenerating	Comparison	Reply
RAM (MB)	50	200	50	50	20

Table 10: Specifications of Edges for Scenario 3 & 4

Edge Name	Request	Hashing	AvSending	acceptance
Mips	50	50	50	20
TupleLength	500	500	1000	500

Table 11: Mapping the module to node in scenario 3

Device	User	User	User	Fog	Fog
Module	CollectingRequests	AvGenerating	Reply	RequestProcessing	Comparison

Table 12: Mapping the module to node in scenario 4

Device	User	User	User	SG	SG
Module	CollectingRequests	AvGenerating	Reply	RequestProcessing	Comparison

In the end, we create control loops through AppLoop class. These loops are used to measure end-to-end delay, energy consumption, cost, network usage. In general, an object from this class contains a list of application modules in the loop from module one to the last module. In the next section, these measurements are illustrated by the graph.

IV. Performance evaluation

In this section, we compared our work in terms of parameters such as network usage and delay and energy consumption and cost with the presence of Fog and without the presence of Fog. In all the results, it is seen that in comparison with the parameters, our method in the presence of Fog is better than the time there is no Fog. To evaluate proposed protocol, we compared the proposed method (P.M) with the method presented in the base paper with the SEA [5].

Energy consumption means the total energy consumed by the system. This energy is used by any of the network components such as fog, sensor,

gateway, and so on. This energy consumption can be calculated by Formula (1) in [18]. The amount of network utilization, cost is also obtained from Formula (2), (3) and (4) in [18].

$$energy = CEC + (CT - LUUT) * HLU \quad (1)$$

CEC is current energy consumption. CT refers to current time. Also, LUUT returns the value of last utilization update time, and HLU refers to last utilization of the host. The energy consumed from the beginning of the simulation is zero. After running the simulation to obtain the energy consumed, the simulation time, which is the difference between the current system time and the last utilization updates time, is multiplied in the last utilization of host and eventually added to the amount of current energy consumed. This value is based on mega joule.

$$Networkusage = \frac{(TL * TS)}{MST} \quad (2)$$

In Formula (2), the values of TL and TS represent the total latency and total size of the tuple, respectively. Maximum Simulation Time Shown with MST.

$$Cost = CC + (CT - LUUT) * RPM * LU * TM \quad (3)$$

Where CC is current cost, CT denotes the current time, LUUT is the last utilization update time, RPM represents the rate per MIPS, LU is last utilization, and TM is total cost (allocation of memory, bandwidth, and processor). At the beginning of the simulation, all cost is set and the initial cost is zero. After running the simulation of the updated values and the total simulation cost is obtained according to the Formula (3). The current time value of the system decreases from the value of the utilization update time, then this amount is multiplied by the rate of getting per million instructions per second per host, and the result adds to the current cost of the simulator. This value is based on the non-negative number.

We use Formula (4) to calculate the delay.

Delay = ST – PCT (4)

Where ST is simulation time, PCT denotes packet creation time. This means that end to end delay of a received packet is calculated as the difference between the time of the packet arrival at the server to process and the time the packet created and sent by sender node.

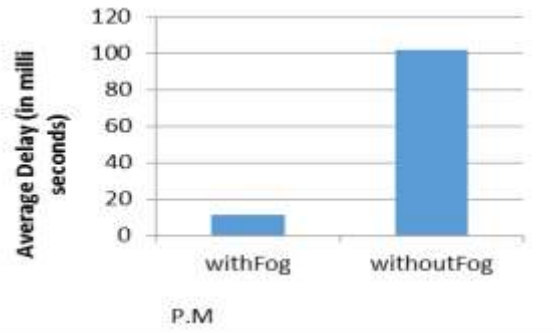

Figure 8: Comparison the network usage in with fog and without Fog

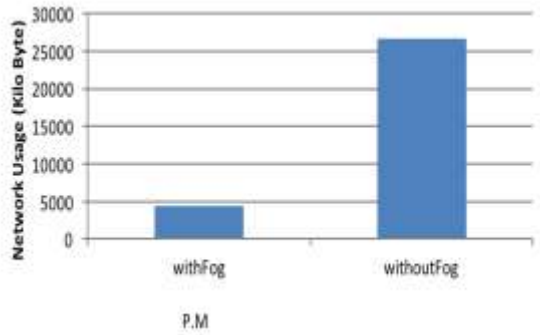

Figure 9: Comparison the delay in with fog and without Fog

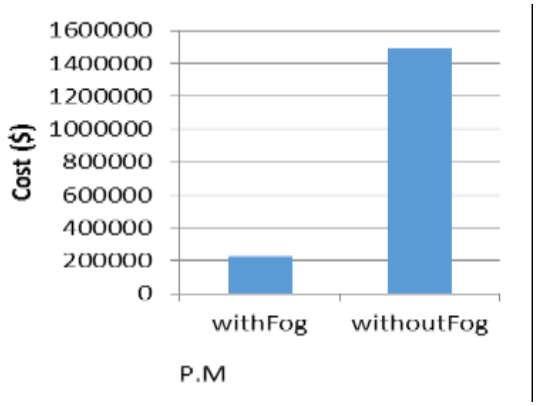

Figure 10: Comparison the cost in with fog and without Fog

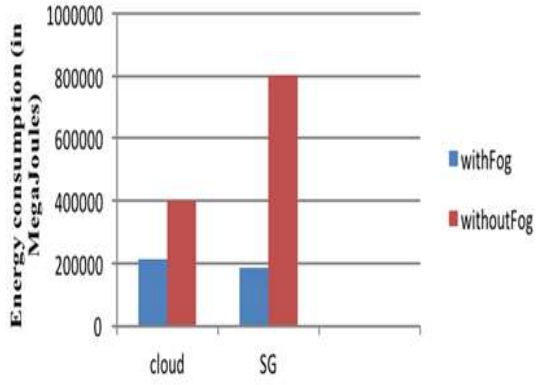

Figure 11: Comparison the energy consumption in with fog and without Fog

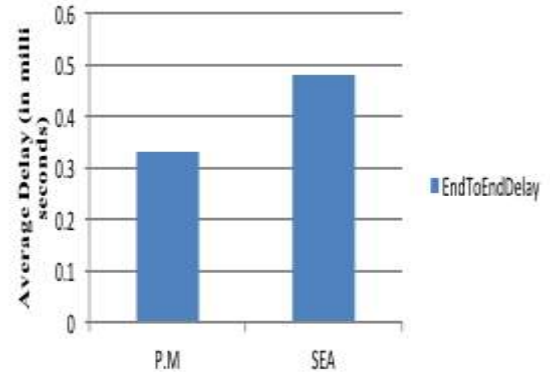

Figure 12: Comparison of the suggested authentication method with SEA in delay End to End

Figure 12 showed the average processing delay of sensing-actuation control loop. In situation of cloud-only placement strategy, as shown in Figure 12, cloud data centers turned to a bottleneck in execution of the modules, which caused a notably significant increase in latency in SEA. However, Edge-ward placement in proposed method succeeds in maintaining low latency, as it places the modules critical to the control loop close to the network edge.

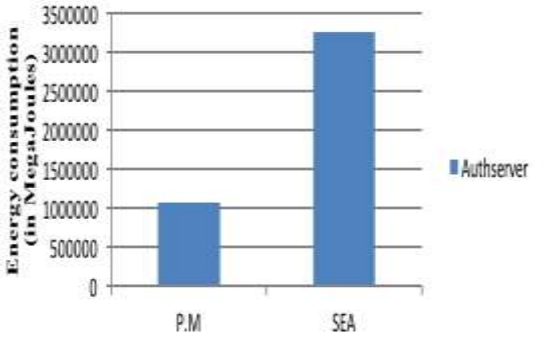

Figure 13: Comparison of the proposed authentication method with SEA in energy consumption server

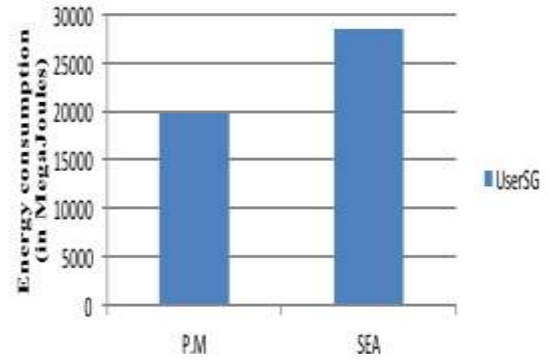

Figure 14: Comparison of the proposed authentication method with SEA in energy consumption user

Figure 13 portrays comparison of energy consumption for the user who is requesting authentication. Figure 14 portrays comparison of energy consumption of the server that performs the authentication process. As shown in Figure 13 and 14, that our suggested method minimizes the energy consumption by using fog devices in Edge-ward placement strategy while increases energy consumption in the SEA is rising. In paper [5], the authentication process is performed by the smart gateway. While we do this process with the Fog and the access list, and additionally, in the base paper, if the user wants to access sensors for other rooms, for each gateway, he/she must be performed once the whole authentication process. For example, for 5 sensors of different gateways, the whole process must be performed 5 times. But in our approach, with one authentication time, can access any number of arbitrary sensors. Because with one authentication time, the message is sent less and the CPU is less involved, the bandwidth is less involved and the memory is less involved. For example, if the user is a building manager, for every home in the base paper, one must be authenticated. But in our approach, we have access to all homes with one authentication time.

5. Conclusion

In this paper, we proposed an authentication scheme using Shamir's threshold technique [17] and design a Foggy Smart Home Architecture (FSHA). The objectives of this work are: 1- Improving the performance of smart home system. 2- Reducing the amounts of data which are sent to cloud server by hiring a fog device in the edge of network and doing some lighter processes by it. 3- Writing an authenticating and authorizing protocol for identifying end-users and minimum the time that is needed for this security operation. 4-Performing security operations so that the proposed method can online/offline password guessing attack and user impersonation attack and man-in-the-middle attack.

On the authority of our analysis and our proposed method actually convinces and satisfies most properties for the procedure of user authentication and effectively copes with Internet of Things environments with considerable and greater efficiency by using secret sharing architecture. For that, it could be usefully and helpfully applied to Internet of Things based.

Smart homes where different kind Things are on service. In our approach, with one authentication time, can access any number of arbitrary sensors. Because with one authentication time, the message is sent less and the CPU is less involved, the bandwidth is less involved and the memory is less involved. Our proposed method minimizes energy consumption and also reduces delay by taking the advantage of using fog devices in Edge-ward placement strategy.

References

[1] I. Stojmenovic and S. Wen, "The fog computing paradigm: Scenarios and security issues," Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on, IEEE, (2014).

[2] Q. Jing, A.V. Vasilakos, J. Wan, J. Lu and D. Qiu, "Security of the internet of things: Perspectives and challenges," Wireless Networks 20, 8, 2481-2501, 2014.

[3] S. Li, T. Tryfonas and H. Li, "The internet of things: a security point of view," Internet Research 26, 2, 337-359, 2016.

[4] S. Yi, Z. Qin and Q. Li, "Security and privacy issues of fog computing: A survey International Conference on Wireless Algorithms, Systems, and Applications, Springer," 2015.

[5] S.R. Moosavi, T.N. Gia, A.-M. Rahmani, E. Nigussie, S. Virtanen, J. Isoaho and H. Tenhunen, "SEA: a secure and efficient authentication and authorization architecture for IoT-based healthcare using smart gateways," Procedia Computer Science 52, 452-459, 2015.

[6] L. Barreto, A. Celesti, M. Villari, M. Fazio and A. Puliafito, "An authentication model for IoT clouds," Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015, ACM, 2015.

[7] C.-T Li, T.-Y. Wu, C.-L. Chen, C.-C. Lee and C.-M. Chen, "An efficient user authentication and user anonymity scheme with provably security for iot-based medical care system," Sensors 17, 7, 1482, 2017.

[8] P. Kumar, A. Braeken, A. Gurtov, J. Iinatti and P. H. Ha, "Anonymous Secure Framework in Connected Smart Home Environments," IEEE Transactions on Information Forensics and Security 12, 4, 968-979, 2017.

[9] M. Amadeo, A. Molinaro, S. Y. Paratore, A. Altomare, A. Giordano and C. Mastroianni, "A Cloud of Things framework for smart home services based on Information Centric Networking," Networking, Sensing and Control (ICNSC), 2017 IEEE 14th International Conference on, IEEE., 2017.

[10] J. Dutta, and S. Roy, "IoT-fog-cloud based architecture for smart city: Prototype of a smart building," Cloud Computing, Data Science & Engineering-Confluence, 2017 7th International Conference on, IEEE, 2017.

[11] J. Li, J. Jin, D. Yuan, M. Palaniswami and K. Moessner, "EHOPES: Data-centered Fog platform for smart living," Telecommunication Networks and Applications Conference (ITNAC), 2015 International, IEEE, 2015.

[12] Y. Sahni, J. Cao, S. Zhang and L. Yang, "Edge Mesh: A new paradigm to enable distributed intelligence in Internet of Things," IEEE Access 5, 16441-16458, 2017.

[13] W. Lee, K. Nam, H.-G. Roh and S.-H. Kim, "A gateway based fog computing architecture for wireless sensors and actuator networks," Advanced Communication Technology (ICACT), 2016 18th International Conference on, IEEE, 2016.

[14] A. M. Rahmani, T. N. Gia, B. Negash, A. Anzanpour, I. Azimi, M. Jiang and P. Liljeberg, "Exploiting smart e-health gateways at the edge of healthcare internet-of-things: a fog computing approach," Future Generation Computer Systems, 2017.

[15] H. Gupta, A. Vahid Dastjerdi, S. K. Ghosh and R. Buyya, "iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments," Software: Practice and Experience 47, 9, 275-1296, 2017.

[16] http://math.nist.gov/javanumerics/jama/

[17] Y. Park, and Y. Park, "A Selective Group Authentication Scheme for IoT-Based Medical Information System," Journal of medical systems 41, 4, 48, 2017.

[18] Kabirzadeh, Sabihe, Dadmehr Rahbari, and Mohsen Nickray. "A Hyper Heuristic Algorithm for Scheduling of Fog Networks." In Proceedings of the 21st Conference of Open Innovations Association FRUCT, p. 20. FRUCT Oy, 2017.